Sharing and trading in a telerobotics system

Ong, Kai Wei

2006

Ong, K. W. (2006). Sharing and trading in a telerobotics system. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/6236>

<https://doi.org/10.32657/10356/6236>

Nanyang Technological University

Downloaded on 26 Apr 2023 21:53:59 SGT

SHARING AND TRADING IN A TELEROBOTICS SYSTEM

ONG KAI WEI

SCHOOL OF MECHANICAL AND AEROSPACE ENGINEERING

Nanyang Technological University

2006

SHARING AND TRADING IN A TELEROBOTICS SYSTEM

ONG KAI WEI

SCHOOL OF MECHANICAL AND AEROSPACE ENGINEERING

A thesis submitted to the Nanyang Technological University in
fulfilment of the requirement for the degree of
Doctor of Philosophy in Engineering

2006

Acknowledgements

First of all, I would like to express my sincere appreciation and gratitude to both of my PhD supervisors, Associate Professor Gerald Seet and Associate Professor Sim Siang Kok for giving me the opportunity to explore and research into this interesting work. Their patience and rigor has been invaluable over these past three years. The accomplishment of this thesis would not have been possible without their dedication, valuable knowledge and guidance.

I would like to thank Mrs. Agnes Tan, Mr. Lim Eng Cheng, Mr. You Kim San and Miss Toh Yen Mei from Robotics Research Centre (RRC), Nanyang Technological University, Singapore for providing technical and administrative support through these three years period of exceptional quality.

I would like to extend my appreciation to all fellow research students who have helped me through the period of research and experiments.

Last but not least, I would like to thank my family for their support and encouragement.

Table of Contents	Page
Acknowledgements	i
Table of Contents	ii
Abstract	v
Glossary of Acronyms and Abbreviations	vi
List of Figures	viii
List of Tables	xi
Chapter 1 Introduction	1
1.1 Problem Statement	3
1.1.1 Concept of Seamless Human-Robot Interaction	5
1.1.2 Background of Sharing and Trading	7
1.1.3 Application of Sharing and Trading in Human-Robot Interaction	10
1.1.4 The Need of a Framework of Sharing and Trading	11
1.2 Objective	14
1.3 Research Approach	15
1.4 Thesis Outline	17
1.5 An Overview of the Research Contributions	18
Chapter 2 Literature Review	20
2.1 Human-Robot Systems	20
2.1.1 Evolution of Human-Robot Roles and Relationships in Robotics	23
2.1.2 Summary	27
2.2 Methods of Human Control	28
2.2.1 Control Framework	29
2.2.2 Control Modes	39
2.2.3 Control Mode Transitions	41
2.2.4 Summary	43
2.3 Robot Autonomy	44
2.3.1 Robot Autonomy versus Human Control Involvement	44
2.3.2 Robot Control Architectures	47
2.3.3 Coordination Mechanisms	49
2.3.4 Incorporation of Human Control Inputs into a Robotics System	51
2.3.5 Summary	54
2.4 Human-Robot Communication	54
2.4.1 Methods of Communication	55
2.4.2 Communication Format	55
2.4.3 Purposes of Communication	56
2.4.4 Summary	58
2.5 Chapter Summary	59
Chapter 3 A Framework of Sharing and Trading	63
3.1 Task Allocation	65
3.1.1 Task Sharing and Trading	67
3.1.2 The Role of Task Sharing and Trading in Task Reallocation	71

3.1.3	Summary	74
3.2	Defining Sharing and Trading in a Human-Robot System	75
3.2.1	Input Task (T_I)	76
3.2.2	Task of Human (T_H)	78
3.2.3	Task of Robot (T_R)	79
3.2.4	Task Sharing and Trading ($T_{S\&T}$) between Human and Robot ...	79
3.2.5	Task Reallocation (T_{RE})	84
3.3	Discussion on Framework Formulation	84
3.3.1	Why should Human and Robot Share and Trade?	85
3.3.2	When should human and robot share and trade?	85
3.3.3	How does Human and Robot Know When to Share and Trade? ..	86
3.3.4	How does Human and Robot Share and Trade?	87
3.3.5	What Triggers the Change from Sharing to Trading (or Trading to Sharing)?	89
3.3.6	Who is in Charge of the Sharing and Trading Process?	90
3.4	Chapter Summary	90
Chapter 4	Concept of Sharing and Trading in Telerobotics	94
4.1	Development of a Shared and Traded Telerobotics Framework	94
4.1.1	Application Requirements and Analysis Phase	95
4.1.2	Human and Robot Integration Phase	96
4.2	Requirements	98
4.2.1	Shared Representation	99
4.2.2	Monitoring	100
4.2.3	Learning	102
4.2.4	Conflict Resolution	103
4.3	Robot Capabilities	104
4.4	Chapter Summary	106
Chapter 5	Implementation of a Sharing and Trading Telerobotics System	108
5.1	Design and Development of the Subsystems for the Telerobotics System	109
5.1.1	Robot Hardware Subsystem	112
5.1.2	Navigation Subsystem	116
5.1.3	Localisation Subsystem	120
5.1.4	Planning Subsystem	123
5.1.5	Learning	128
5.1.6	Interfaces Subsystem	133
5.1.7	Summary	133
5.2	Design and Development of the Task Interaction Modes Components	133
5.2.1	Human Control Components	134
5.2.2	Components for Enabling Robot Autonomy	135
5.2.3	Feedback Components	137
5.2.4	Coordination of Task Interaction Modes	137
5.3	Implementation of the Software Modules for the Telerobotics System	139
5.3.1	Software Framework Requirements	140
5.3.2	Software Implementation	141
5.4	Chapter Summary	145

Chapter 6	Experimental Evaluation	147
6.1	Experimental Evaluation on Robot Assists Human	148
6.1.1	Experimental Objectives	149
6.1.2	Robot Assists Human (RAH) Scheme	151
6.1.3	Experimental Design and Procedure	155
6.1.4	Experimental Results	164
6.1.5	Discussion	168
6.2	Experimental Evaluation on Human Assists Robot	169
6.2.1	Waypoints Navigation	171
6.2.2	Landmarks Based Navigation	180
6.2.3	Vision Based Tasks	187
6.2.4	Human-Assisted Localisation	191
6.3	Discussion on Seamless Human-Robot Interaction	193
6.4	Chapter Summary	195
Chapter 7	Discussion and Conclusion	198
7.1	Discussion	198
7.1.1	Achieving Seamless Human-Robot Interaction	198
7.1.2	The Need for a Framework of Sharing and Trading for Seamless Human-Robot Interaction	199
7.1.3	Development of the Framework of Sharing and Trading	200
7.1.4	Shared and Traded Architecture	203
7.1.5	Proof-of-Concept Experimental Evaluation	205
7.2	Contributions	207
7.3	Future Work	209
7.3.1	Unaddressed Issues	209
7.3.2	Experimental Evaluation	210
References		212
Appendix A:	Robot Hardware	
Appendix B:	Implementation Details	
Appendix C:	Questionnaire	
Appendix D:	Experimental Results	
Appendix E:	Computer Vision Libraries	

Abstract

With the functions of physical robots now extended beyond academia into factories, homes and fields, the interactions between humans and robots have become increasingly extensive and ubiquitous. The current state of human interaction with robots in comparison to simple “machines” that operate in structured environment, such as industrial robotics, is quite different. Robots differ from these machines in that they are capable of functioning in evolving situations, reasoning and acting in a relatively complex domain. The deployment of these robots leads to an important research issue in the design and development of a Human-Robot System (HRS), i.e. to look into how human and robot can interact seamlessly to accomplish task objectives.

The main contribution of this thesis is the achievement of seamless Human-Robot Interactions (HRI) based on the concept of *sharing and trading* in an HRS. Seamless HRI implies flexibility in human control through which a human interacts with a robot in different situations, and the adaptability of robot’s autonomy in response to human control. Sharing and trading, is a human-robot cooperation concept that allows both human and robot to contribute according to their degree/level of expertise via exploiting their unique strengths. This work emphasises the importance of modelling a conceptual framework of sharing and trading as a foundation for seamless HRI and the central role of sharing and trading in the design and development of a cooperative HRS. The study of how seamless HRI is achieved, via sharing and trading, can provide a greater understanding of how a human and a robot might cooperate in an HRS.

The development of the framework of sharing and trading is divided into two stages. The first stage is the identification of the key elements and their associated features and attributes for defining what can be shared and traded between human and robot. The elements are human control, robot autonomy and information, identified based on: (1) the command and control of a robot from the perspective of human interacting with the robot; (2) the required degree of robot autonomy from the perspective of a robot interacting with the human; (3) the issues of communication between human and robot. The second stage is the formalisation of the concept of sharing and trading, called task sharing and trading ($T_{S\&T}$). Instead of concentrating on how task can be allocated between human and robot as in traditional task allocation, $T_{S\&T}$ provides a complementary view by looking into the possibilities of letting a robot assists human and human assists robot (RAH-HAR). The basic idea of the RAH-HAR paradigm is to let human and robot work as a team, where both can actively take initiatives to accomplish task objectives by assisting each other in different ways to fit situational needs and capabilities change throughout a task. Consequently, to study how the formulated framework can be applied, it is employed in the implementation of an HRS (i.e. a *teleroobotics system*). The development includes: (1) the implementation of a shared and traded control architecture that supports a spectrum of interaction modes using different coordination mechanisms. The key idea of the development of different interaction modes is based on the different human-robot roles, namely Master-Slave, Supervisor-Subordinate, Partner-Partner, Teacher-Learner and Fully Autonomous mode by the robot. Instead of using only one fixed interaction role strategy, both the human and the robot can engage in different roles to compensate for the unique kinds of limitations possessed by each other; (2) The establishment of a highly flexible communication protocol between the human and the robot facilitates the coordination of their actions both implicitly (for task sharing) and explicitly (for task trading) in accordance to different task situations, task demands and needs.

Finally, using the implemented HRS, proof-of-concept experiments were conducted to assess how seamless HRI is achieved via $T_{S\&T}$. The assessments were based on how a robot assists human in a seamless way and how human assistance can be provided to a robot seamlessly in different task situations respectively. The results obtained show that the RAH-HAR paradigm invoked by the concept of $T_{S\&T}$ for seamless HRI provides a useful approach for human-robot cooperation in compensating each other’s deficiencies.

Glossary of Acronyms and Abbreviations

2D	Two-Dimensional
3D	Three-Dimensional
3T	Three-Layer Architecture
AHRS	Attitude and Heading Reference System
AI	Artificial Intelligence
ANOVA	Analysis of Variance
AP	Access Point
A _s	Autonomy Sharing
A _T	Autonomy Trading
ATRV-Jr	All-Terrain Mobile Vehicle – Junior
AuRA	Autonomous Robot Architecture
CAD	Computer Aided Design
CC	Collaborative Control
CMVision	Colour Machine Vision
COM	Component Object Model
CORBA	Common Object Request Broker Architecture
C _s	Control Sharing
C _T	Control Trading
DARPA	Defense Advanced Research Projects Agency
DAI	Distributed Artificial Intelligent
DES	Discrete Event Systems
FAM	Fuzzy-Association-Memory
FIFO	First-In-First-Out
fps	frame per second
GPS	Global Positioning System
HAR	Human Assists Robot
HRC	Human-Robot Communication
HRI	Human-Robot Interactions
HRS	Human-Robot System
HRT	Human-Robot Team
ICSC	Interactive and Cooperative Sensing and Control
IDL	Interface Definition Language
I _s	Information Sharing
I _T	Information Trading
KBB	Knowledge-Based Behaviour
LMS	Laser Measurement System

MABA-MABA	Men-are-better-at – Machines-are-better-at
MAS	Multi-Agent System
Mbps	Megabits per second
MCL	Monte-Carlo Localisation
MIC	Mixed Initiative Control
NSF	National Science Foundation
OpenCV	Open Source Computer Vision
PC	Personal Computer
PDA	Personal Digital Assistant
PTZ	Pan/Tilt/Zoom
RAH	Robot Assists Human
RBB	Rule-Based Behaviour
RF	Radio Frequency
RRC	Robotics Research Centre
SA	Station Adapter
SBB	Skilled-Based Behaviour
SC	Supervisory Control
SLAM	Simultaneously Localisation and Mapping
SRK	Skill-Rule-Knowledge
TCP/IP	Transmission Control Protocol/Internet Protocol
T_I	Input Task
T_H	Task of Human
TOF	Time-of-Flight
T_R	Task of Robot
T_{RE}	Task Reallocation
$T_{S\&T}$	Task Sharing and Trading
VB6	Visual Basic 6
VFH	Vector Field Histogram
WLAN	Wireless Local Area Network

List of Figures

Figure 1.1: Evolution of interactions between human(s) and machine(s)	1
Figure 1.2: Concept of Seamless Human-Robot Interaction	6
Figure 1.3: The original diagram of the concept of sharing and trading of control between human and computer from Sheridan [2]	7
Figure 1.4: Human-computer sharing and trading of control	8
Figure 1.5: Block diagram of the HRI testbed	16
Figure 2.1a: An exposition of direct master-slave manual control with no computer- aided assistance	30
Figure 2.1b: An exposition of indirect master-slave manual control with computer- aided	31
Figure 2.1c: An exposition of semi-autonomous control - parallel type	31
Figure 2.1d: An exposition of semi-autonomous control - serial type	32
Figure 2.1e: An exposition of semi-autonomous control - parallel and serial combined type	32
Figure 2.1f: An exposition of autonomous control	33
Figure 2.2: Rasmussen's SRK model	42
Figure 2.3: Control modes based on robot autonomy and human control involvement in accordance with varying nested ranges of action of robot	45
Figure 2.4: Robot control system spectrum	47
Figure 2.5: Hybrid robot control architecture	48
Figure 2.6: An implementation example of shared control using motor schemas coordination approach	52
Figure 2.7: An implementation example of traded control using subsumption coordination approach	53
Figure 2.8: An illustration of information sharing and trading between a human and a robot in a fetch-and-carry task	57
Figure 2.9: Summary of Chapter 2	60
Figure 3.1: The Fitts' List	66
Figure 3.2: Concept of task sharing and trading between a human and a robot	68
Figure 3.3: Activities within a Human-Robot System	76
Figure 3.4: Phenomenon of sharing and trading induce by different human-robot roles and relationships described in Table 2.2	82
Figure 3.5: Range of task interaction modes in accordance to the characterisation of $T_{S\&T}$ in different human-robot roles and relationships depicted in Figure 3.4.....	88
Figure 3.6: An application-driven framework for the design and development of a HRS	92
Figure 4.1: A conceptual structure of an HRS	95
Figure 4.2: Examples of a range of task interaction modes in the telerobotics system ..	97
Figure 4.3: An overview of the essential requirements for achieving effective $T_{S\&T}$	99
Figure 5.1: An overall view of the implementation of the sharing and trading telerobotics system	108

Figure 5.2: Schematic of the relationship between different functional subsystems of a telerobotics system	110
Figure 5.3: Telerobotics system architecture	111
Figure 5.4: ATRV-Jr™ system description chart	113
Figure 5.5: Robot reference system, basic motion and navigation behaviours	117
Figure 5.6: Example of a 2D binary occupancy map, a 3D CAD modelled of the environment and map building	119
Figure 5.7: An illustration of reactive path planning	124
Figure 5.8: Learning scheme for the identification of landmarks	130
Figure 5.9: Sample vision attention focus frames	132
Figure 5.10: An Overview of interaction modes coordination	138
Figure 5.11: An example of hybrid coordination in exclusive shared mode	139
Figure 5.12: Software Framework	142
Figure 6.1: An overall view of the experimental evaluation	148
Figure 6.2: An overview of the human-robot roles and roles transitions for the experimental evaluation on robot assists human	149
Figure 6.3: An overview of the robot assists human scenarios	153
Figure 6.4: Experiment Setup	157
Figure 6.5: Schematic layout of the test environment – RRC	158
Figure 6.6: Snapshots of the various perspectives of the test environments	158
Figure 6.7: Sample video frames from participant 2 (trial 1) via exclusive shared mode	162
Figure 6.8: Sample robot front perception representations of the test environment from the same run recorded in Figure 6.7	163
Figure 6.9: Schematic layout of the multiple waypoints navigation course – RRC	173
Figure 6.10: Exclusive traded mode coordination architecture for waypoints navigation	174
Figure 6.11: Multiple waypoints navigation state flow	175
Figure 6.12: Sample video frames of multiple waypoints navigation taken by the robot's vision system	176
Figure 6.13: Schematic layout of the single waypoint navigation course – RRC	177
Figure 6.14: A snapshot of the dead-lock situation in Figure 6.13	178
Figure 6.15: Sample video frames for single waypoint navigation	179
Figure 6.16: Actual x-y plot of the robot trajectory for single waypoint navigation ...	180
Figure 6.17: Schematic layout of the landmarks based navigation course – RRC ...	181
Figure 6.18: Landmarks learning state flow	182
Figure 6.19: Sample vision attention focus frames for the learning phase taken by the robot's vision attention system	183
Figure 6.20: Sample laser range finder plot for the learning phase in accordance to the sample video attention frame in Figure 6.19	184
Figure 6.21: Compass heading for the learning phase	185
Figure 6.22: Actual x-y plot of the robot trajectory for landmark based navigation ...	186
Figure 6.23: Sentry task state flow	188
Figure 6.24: Snapshots of people following	189
Figure 6.25: Snapshots of the robot following	190
Figure 6.26: Actual x-y plot of the robot trajectories overlayed on the schematic layout of the test course depicted in Figure 6.5	191
Figure 6.27: Snapshots of Human-assisted localisation	192

Figure 6.28: An overall view on the assessment of seamless HRI in Chapter 6	194
---	-----

List of Tables

Table 1.1: A research curriculum of Human-Robot Interaction	4
Table 2.1: Different types of Human-Robot Systems	21
Table 2.2: A summary of the five human-robot roles and relationships	26
Table 2.3: Four human or robot roles in a HRT suggested by Bruemmer et al. [47]	26
Table 2.4: Different forms of control derived from the five human-robot roles and relationships summarised in Table 2.2	28
Table 2.5: Degrees of autonomy based on varying nested ranges of action	45
Table 2.6: Ten-level formulation of robot autonomy	46
Table 2.7: Arbitration coordination mechanisms	49
Table 2.8: Command fusion coordination mechanisms	50
Table 2.9: A summary of human control, robot autonomy and information	61
Table 3.1: Different types of cooperation strategies between a human and a robot based on how the human and the robot might assist each other as shown in Figure 3.2	69
Table 3.2: Woods' Un-Fitts List	70
Table 3.3: Examples of the kinds of questions that the human and the robot need to be Able to know so as to assist each other	70
Table 3.4: Examples of the types of questions and allocation responses to be addressed in adaptive task allocation	72
Table 3.5: A flexible prior task allocation based on "who does what" mandatory allocation and "when and how" provisional allocation decisions	74
Table 3.6: Basic questions to be addressed for the formulation of a framework of sharing and trading	84
Table 5.1: Description of the four processes in Figure 5.8	131
Table 5.2: A summary of the implemented behavioural units in this research	136
Table 5.3: Robot server modules	144
Table 6.1: Experiment design based on a replicated Latin square	159
Table 6.2: Results of the experiments	164
Table 6.3: ANOVA table for task completion time	164
Table 6.4: A summary of the four experiments conducted for human assists robot	170

Chapter 1

Introduction

The evolution in the field of robotics has greatly affected the way people work in every sector of the economy. Many different kinds of tasks that were once performed manually by human workers have been replaced by machines (or robots). A simple illustration of this trend between the interactions of human(s) and machine(s) is presented in Figure 1.1.

Figure 1.1: Evolution of interactions between human(s) and machine(s) (presented in the paper by the author in Ong et al. [1])

As illustrated in Figure 1.1, the evolution starts from hard manual labour to work performed autonomously by a robot colony without any human intervention. However, three important phenomena are reflected. The first is the invasion of machines. Machines are employed to improve task performance as compared to work that is performed manually. The second is the evolution in which human is decoupled from the robot which is performing its task in a remote environment. This evolution is due mainly to development in the communication technology. The third is that of the human involvement in the control loop. This is because in most applications, the tasks involved are often non-repetitive and unpredictable where special-purpose machines cannot be used. Even in highly automated systems (e.g. in factory automation), humans are still required for monitoring, detection of abnormalities and intervention when necessary. Such tasks require human perception (e.g. judgment), planning and control to ensure reliable operations. Therefore, these kinds of tasks are normally performed via remote control by human operators using video inspection and master-slave mechanical manipulation, called teleoperation (Sheridan [2]).

Automation and teleoperation have long been identified as a key technology for space exploration [2, 3, 4]. Operations in space [3] such as manufacturing and transportation of rocks are automated with human monitoring and intervention through teleoperation when necessary. This integration of automation with teleoperation became the foundation of what is now termed *telerobotics*. Here, telerobotics is viewed as a Human-Robot System (HRS); defined as a “*mixed system in which both human and physical robot interact, each as a cooperative intelligent entity*” (Hancock [5]). Telerobotics [2] combines the concepts of teleoperation and automation, enabling a human operator to supervise the execution of a remote task rather than exercising continuous manual control. This evolution is illustrated in Figure 1.1(e) and 1.1(f). Since 1970s, telerobotics has been an active area of research and development in other domains as well. This includes other terrestrial applications such as manufacturing in nuclear plant [6], search and rescue [7,] military operations [8, 9], and so forth. Increasingly it is also being considered for applications in the agricultural [10], security [11] and construction industries [12] where, although not overtly hazardous, socio-economic arguments can be made in favour of the adoption of telerobotics [2].

1.1. Problem Statement

The current state of human interaction with robots in a telerobotics system (or an HRS¹), in comparison to teleoperated/automated “machines” that operate in structured environment, such as industrial robotics, is quite different. Robots differ from these machines in that they are capable of functioning in evolving situations, reasoning and acting in a relatively complex domain (Giralt et al. [13]). It is possible to define “robot” in such a way that it includes simple actuating or sensing machines, for example based on the definition from Amigoni et al. [14], which define a robot as “*the information machine of robotics which has interaction between itself and the world, with sensors, or with actuators, or with both sensors and actuators*”. Although such a definition is useful in some contexts, it is too broad for the purposes of this thesis. Here, the term robot is viewed as a physical embodied *semisentient agent* (Lopes & Connell [15]), which has the capability to act reasonably in a semi-structured environment, reasons about its own action, learn, and adapt to some extent on the basis of human feedback or from its own environment.

In spite of promising research from the field of robotics and artificial intelligence (AI), it is apparent that the attempts to develop and employ intelligent autonomous robots have not been successful to address the inevitable limitations to what the robot can perceive and reason apart from human input [15, 16]. The primary reason is due to the high degree of complexity of perception and motion of a robot required in an unstructured and dynamic environment, which made research in robotics and AI extraordinarily difficult [15]. This complexity arises in part from the need for a robot to perceive, act and reason about the uncertainty of the environment in real-time. Although, the goal of building intelligent autonomous robots have not been achieved, it is possible for the current-state-of robots to perform useful tasks (e.g. [7, 11]) and to provide appropriate assistance to human to correct his/her control input errors by supporting perception and cooperative task execution [16, 17]. Systems which facilitates robots to cooperate (i.e. to work closely) with human are practical and are attracting increasing attention from researchers. This leads to an important research issue in the design and development of an HRS, i.e. to look into how a human and a robot should

¹ Different types of HRS and their relation to telerobotics system is discussed in Section 2.1

“interact” so as to cooperatively accomplish task objectives (Murphy & Rogers [16]). This issue is an emerging research area in robotics, namely Human-Robot Interaction (HRI) [16]. HRI can be broadly defined as “*the study of the humans, robots, and the ways they influence each other*” (Fong et al. [17]).

In 2001, DARPA/NSF held a workshop on HRI (Murphy & Rogers [16]). The interdisciplinary nature of the workshop promoted the interactions of roboticists with psychologists, sociologists, cognitive scientists, communication experts and human-computer interaction specialists to address the challenging issues in HRI for meaningful research, synthesis, and technology transfer. The primary objective of this workshop is to define a future vision for robotics, which must work together with humans to achieve common goals. The research curriculum of HRI presented in Table 1.1 was an outcome of the workshop. Hence, the research work presented in this thesis focuses on the first three curriculums, namely, *task allocation*, *modelling of HRI* and *HRS architecture* for the design and development of a cooperative HRS. The main concern for the design and development of a cooperative HRS is the achievement of *seamless HRI*. This concept is discussed in the following section.

Table 1.1: A research curriculum of Human-Robot Interaction (adapted from Murphy & Rogers [16])

<p>1. Task allocation: Concerns of deciding which tasks should be performed by the humans, the robots or by a combination of both during the initial design and development stage of an HRS.</p>
<p>2. Modelling of HRI: Concerns include modelling of human reasoning, intention and action (i.e. to let the robot understands what the human is doing within the context of the task) to more interaction specific concerns such as modelling of human-robot roles and relationships, human-robot authority issues, etc.</p>
<p>3. HRS Architecture: Concerns of the integration of humans and robots at system-level.</p>
<p>4. Human-Robot Interfaces: Concerns of the design and development of user interfaces for HRI.</p>
<p>5. Usability, reliability and safety evaluation standards & metrics: Concerns of the design and development of user interfaces for HRI.</p>
<p>6. Application evaluation: Concerns of the design and development of user interfaces for HRI.</p>

1.1.1 Concept of Seamless Human-Robot Interaction

Suppose in an HRS, a human and a robot are requested to perform assigned tasks². In order to achieve adaptation to changes in the task, work environment and capabilities of the human and/or the robot during task execution, both human and robot need to change their interaction roles to take on different task responsibilities³. The ability to vary interaction roles and responsibilities among the human and the robot presents new challenges. In remote operation applications such as space explorations, military operations, automated security, search and rescue, etc., the human does not have direct visual awareness of the environment to perform the required tasks. In these applications, a tight interaction between the human and the robot is required for effective cooperation. This raises an interaction dilemma. On one hand, the robot operating in the remote environment can be expected to be in a “better position” to advise/inform the human regarding navigation issues (i.e. react locally to the remote environment) and refuses consent to dangerous human commands (e.g. running into obstacles). On the other hand, due to its limited ontologies, the robot requires human assistance on tasks such as object recognition, decision-making, and so forth. Here, limited ontology means that the robot is not able to use knowledge either from its knowledge-base or from the environment to control its unspecified parameters.

To overcome the interaction dilemma above, adopting appropriate roles that exploit the capabilities of both human and robot as well as crafting natural and effective modes of interaction are important in creating a cooperative HRS. This concern is currently identified as one of the research problem in the area of HRI for effective human-robot cooperation. To this end, innovative paradigms have been proposed over the years to redefine the roles of the human and the robot from the traditional master-slave relationship [2, 5] , to a model such as the human as cooperator (e.g. [17, 18, 19]), supervisor (e.g. [2, 20, 21]) or teacher (e.g. [22]) rather than just as the master controller of the robot. On the other hand, the slave robot is modelled in such a way that it becomes an active assistant/partner (e.g. [17, 18, 19]), subordinate (e.g. [2, 20, 21]) or learner (e.g. [22]) of the human, supporting perception and cooperative task execution.

² The role of *task allocation* (i.e. the first HRI research curriculum) in the design and development of HRS is presented in Chapter 3.

All of the above human-robot roles and relationships are important, since each stresses a different aspect of HRI. It will be beneficial if there is a common framework formulated to characterise how human and robot cooperate based on these human-robot roles. This is because an understanding of the nature of interactions of these roles can lead to the identification and classification of different HRI strategies⁴ for the design and development of a cooperative HRS. This is important for effective cooperation, where human and robot can engage in different roles based on their task capabilities to meet new and unexpected task situations. However, such a formulation based on the advantages of different roles is lacking in the current literature of robotics. The evidence to address this issue is the aim of Chapter 2. To facilitate this, a basic research need is to consider how to achieve *seamless* HRI based on different human-robot roles. Figure 1.2 depicts the concept of seamless HRI in an HRS with different human-robot roles.

Figure 1.2: Concept of Seamless Human-Robot Interaction

The term seamless refers to the *flexibility* in human control for the human to interact

³ Roles and responsibilities are related, and normally roles are defined in order to specify the responsibilities of both human and robot.

⁴ Here, HRI strategy refers to task interaction scheme or method that allows human and robot to cooperate to execute the HRS task as effectively as possible

with a robot in different situations and the *adaptability* of the robot autonomy in response to the human control. In this context, flexibility means the ability to perform different aspects of the HRS task easily by the human. On the other hand, adaptability means adjustment to the robot autonomy for performing the task, meaning that a robot should be able to carry out its process no matter what disturbances might occur in the task environment (Lopes & Connell [15]). In this manner, both the human and the robot may work together more effectively to ensure high-level of system performance and the satisfaction of task demands. In this thesis, it is suggested that an approach towards *seamless* HRI is via adjusting appropriate degree of *sharing* and *trading* between the human and the robot. In order to provide the context and perspective for the study, the following section gives a background to sharing and trading.

1.1.2 Background of Sharing and Trading

The concept of sharing and trading was first introduced by Sheridan [2] to describe the roles of computer in human supervisory control of dynamic systems. Example of such systems includes chemical or power plant, aircraft, automobile, etc. He invoked the concept of sharing and trading in the context of a “control” in accordance to how much task-load a human can carry out with the help of the computer to what the human can carry alone. These are further broken down into the sub-categories of *extend*, *relieve*, *replace* and *backup* and are conveyed in Figure 1.3.

Figure 1.3: The original diagram of the concept of sharing and trading of control between human and computer from Sheridan [2]. L is the load or task, H the human, C the computer

Sheridan [2] states (pp. 65):

In sharing - “The computer can extend the human’s capabilities beyond what he can achieve alone or it can partially relieve the human, making his job easier”; and in trading – “the computer can backup the human in cases where he fails, or the computer can replace him completely.”

He affirmed that extending and reliving are examples of *sharing of control*, and backing up and replacing are examples of *trading of control*. Specifically, sharing of control means that the human and the computer control a process or different aspects of the process at the same time (Figure 1.4 (a)). On the other hand, trading of control means that the human and the computer exchange control from time to time to control a process (Figure 1.4 (b)). Although the concept of sharing and trading of control was invoked by Sheridan more than a decade ago, his understanding on sharing and trading still holds true today. This concept is widely used in automation to let computer provides appropriate assistance to human. To describe how a human and a computer shared and traded control, examples are given in the following sub-section.

Figure 1.4: Human-computer sharing and trading of control (adapted from Sheridan [2])

Examples of Sharing and Trading of Control in Automation

The concept of sharing and trading is widely adopted in automation as one of the essential aspect of design for envisaging physical cooperation between the human and the computer (Inagaki [23]). The form of the required automation differs, depending on the type of cooperation. For example, in sharing, the automation is designed to assist the human by reducing his/her workload in every possible process. In the case of trading, the automation is designed so that it can replace the human completely. Here, two types

of automation system, namely automobile and aircraft are selected to describe how human and computer shared and traded control.

In *automobile* research, the concept of sharing and trading is employed to assist the human driver. An example of sharing is a lane-keeping support system that reduces the human burden (i.e. relieving) during driving. The computer system detects white lane marker on the road, and it generates torque to assist the driver's steering action for keeping the host vehicle approximately in the centre lane [24]. Another instance of sharing control is the collision prevention system to prevent accidents caused by delayed driver's response. The computer system assists the human driver by slowing down or stops the automobile via power steering or braking. This feature extends the capability of the human driver by allowing the driver to drive at a higher speed. In the case of trading of control in automobile, an example is an in-vehicle computer system that can temporarily take over control from the human to drive the vehicle during emergencies (i.e. backing-up) or allow auto-driving for prolonged durations (i.e. replacing) [25].

In *aircraft*, the concept of sharing and trading is employed for letting the human pilot and the automation (i.e. the autopilot) cooperate in making safe, smooth and efficient flights. Lateral and vertical navigation are two essential processes [23, 26]. The human pilot usually takes responsibilities for both lateral and vertical navigation during take off. In the climb phase, the pilot may handle lateral navigation and hand the control over to the computer (i.e. trading of control) to direct the vertical navigation of the craft. This means that after the responsibilities of controlling the vertical navigation is handed over to the computer, the human pilot and the computer control the aircraft's navigation simultaneously (i.e. sharing of control). During cruise, the pilot often hand both lateral and vertical navigation over to the computer (i.e. trading of control, the computer replaces the human). In descending or landing, depending on the situation, the pilot may seize back control of either lateral or vertical navigation process from the computer (i.e. trading of control back from the computer to the human).

Given above is the discussion on the concept of sharing and trading as a basis for envisaging how technological system (in this case, computer) might assist human through different type of interaction modes (i.e. *extend*, *relieve*, *back-up* and *replace*) introduced by Sheridan [2]. Although the concept introduced by Sheridan provides a

good basis on how human and technological system work together (Inagaki [23]), it might not be sufficient to address how human and robot can interact seamlessly. This is because in accordance to the current-state-of theory and practice in robotics, the goal of building robots that can operate without any human guidance has not yet been achieved (Giralt et al. [13], Lopes & Connell [15], Murphy & Rogers [16]). Virtually all applications of robot systems require human to be in the control loop to provide appropriate assistance to the robot in order to ensure safe operation or to ensure that the system performance does not degrade. This implies that to achieve seamless HRI, the concept of sharing and trading envisaged must not only facilitates the requirements of letting a robot assists human, it must also allow for human to provide appropriate assistance to the robot in different task situations. The formalisation for sharing and trading to address this research need for seamless HRI is the aim of Chapter 3.

1.1.3 Application of Sharing and Trading in Human-Robot Interaction

Although the area of HRI has recently begun to establish a niche in the research and application community of robotics, there have been good foundations developed in understanding how human and robot might interact via sharing and trading.

Within the discipline of robotics, the concept of sharing and trading is widely used for incorporating the strengths⁵ of human and robot. The aim is to achieve mutual compensation of both the human's and the robot's individual weakness [2, 4, 17, 18, 19, 20, 21]. For instance, sharing of control has often been described in both the literature of telemanipulation [2, 4, 19] and teleoperation of mobile robots [17, 18, 20, 21]. In telemanipulation, an example of sharing is the manipulation of a task where the compliance control is done by the robot automatically while position control is achieved by human's manual control [4, 19]. In mobile robot teleoperation, an example of sharing of control is described as follows: the human directly controls the robot on board pan-tilt-zoom camera to provide a movement direction, i.e. to provide perceptual guidance; and the robot will respond to the human command by scaling its autonomy to drive the mobile platform according in the direction of the gaze [21]. In both cases, trading is

⁵ That is by providing opportunity for the robot to utilise the intelligence, skill, perception and judgment of the human; and for the human to take advantages of robots, such as the ability to perform repetitive and routine tasks, work in hazardous environment, etc.

normally used in conjunction with sharing to let the human and the robot assist each other via the exchange of control and task information when both have problem performing the assigned task [17, 18, 19, 20, 27].

Basic Questions of Sharing and Trading

The basic questions in sharing and trading are as follows (from Sheridan [2]):

In sharing: “*Which tasks should be assigned to human and which to the robot?*”

In trading: “*Which aspects of the tasks to trade, and when should control be handed over and when should it resume control during task execution?*”

As a consequence, researchers from the domains of telemanipulation (e.g. [4, 19]) and mobile robot teleoperation (e.g. [17, 18, 20]) have developed various novel HRS architectures (i.e. the third HRI research curriculum as listed in Table 1) to address these questions. Although their solutions are application specific, the fundamental principles are similar, i.e. to facilitate interactive task allocation and cooperative decision-making between the human and the robot. The purpose of interactive task allocation is to spatially/temporally distribute the task to the human and/or the robot, based on their capabilities and performance during task execution. The purpose of cooperative decision-making is to provide for arbitration or fusion of task commands from the human and the robot.

1.1.4 The Need of a Framework of Sharing and Trading

Even though the concept of sharing and trading has been widely adopted and studied in robotics, a unified general framework (i.e. a comprehensive formalisation) based on this concept to assist in the design and development of an HRS for seamless HRI, is still missing. This is because key *elements* involved in the process of sharing and trading between a human and a robot has not been addressed in a holistic manner. To explain what is meant by key “elements” in this context, it might be useful to first provide a working definition of sharing and trading as a basis for further discussion. To facilitate, the definition of sharing and trading from Sheridan (see Section 1.1.1) are redefined to explain the process of sharing and trading between the human and the robot.

Definition of Sharing and Trading

Webster's dictionary [28] defines "sharing" and "trading" as: "to *join* with another or others in use of some (thing)" and "to *exchange* one (thing) for another" respectively. Here, "to join" means that both human and robot work together through the use of some "thing" to ensure the success of task performance; and "to exchange" means that a human and a robot give and receive an equivalent of "thing" which they own while working together. In the context of sharing and trading, the tasks are the actions the human and the robot undertakes to achieve their goals. The human needs to be able to see those tasks and control them, if necessary, during sharing and trading. On the other hand, the robot needs to be equipped with the capability to scale its own degree of autonomy to meet whatever level of control input from the human. To facilitate, both human and robot must adopt the same "language" or representation so as to prevent miscommunication when they share and trade information. The rationale of defining sharing and trading in this manner is substantiated in Chapter 2, in accordance to the basic requirements of HRI invoked by different types of human-robot roles.

Based on the above definition, it is posited in Chapter 2 that the "things" or key elements that a human and a robot share and trade are *human control*, *robot autonomy* and *information*. Human to robot sharing and trading is based on how human varies his/her level of control involvement with the robot, while robot to human is based on how the robot adjusts its autonomy level dynamically to accept a full spectrum of human control changes. To facilitate smooth coordination of control and autonomy sharing and/or trading, human and robot needs to share information so as to aware of each other behaviours. In addition, they may trade information if they have doubt regarding the perceived information. Following this, the reasons why there is a need to develop a unified general framework for sharing and trading is presented in the following sub-section.

Why a Unified General Framework is needed?

First, a unified general framework is needed to synthesise the key elements involved in the process of sharing and trading to model how a human and a robot interacts (i.e. the second HRI research curriculum, modelling of HRI). The purpose of the framework is to

incorporate different human-robot roles seamlessly in an HRS. Such a framework formulated to exploit the advantages of different human-robot roles for seamless HRI is missing from literature of robotics adopting the concept of sharing and trading, introduced by Sheridan [2], for the design and development of a cooperative HRS. The evidence to substantiate this research need is presented in Chapter 2, through a critical review of published literature of the applications of the concept of sharing and/or trading in robotics for effective HRI.

Second, there is a need to have a framework to address the role of sharing and trading in decomposing and allocating tasks between humans and robots (i.e. the first HRI research curriculum, task allocation) in a structured and systematic manner. This is essential in the initial design stage of HRS for providing a holistic basis of determining which system-level task should be performed by a human, by a robot or by a combination of both in accordance to their capabilities and limitations during task execution. However, in accordance to the critical review in Chapter 2, a formalisation for sharing and trading to address how task can be allocated between humans and robots is lacking in the current literature of robotics and HRI. This is because applications of sharing and trading are normally applied on an ad hoc basis, without a comprehensive formalism to address the problems of task allocation.

Third, without a unified general framework it is difficult to address, or discuss the research issues associated with the concept of sharing and trading in a holistic manner. In accordance to the discussion in Section 1.1.2 and 1.1.3, the primary research issues of sharing and trading are summarised as follows:

- In sharing, there are apparent dependencies between the actions taken by the human/robot and the actions available to another as both human and robot can be operating competitively or cooperatively. In this case, their actions can reinforce or interfere with each other. Hence, the main concern is how to resolve their conflicting actions dynamically during task execution.
- In pure trading, (e.g. Kortenkamp et al. [27]), at any one time either the human or robot has full control, and over time this control responsibility is switched between them in accordance to the task at hand. The next issue involved is: who decides when the control is transferred and how to ensure the transfer of control is exchanged

smoothly.

- The above issues consider sharing and trading separately. This is useful because sharing and trading are differentiated to simplify the types of human-robot cooperation strategies. If both sharing and trading are to be applied in conjunction, how can these concepts assist in the design and development of a cooperative HRS? In particular, what are the consequences and requirements of sharing and trading, how are they related, and what form they take? This poses greater difficulty and more challenging than the issues mentioned above. This is because the HRS architecture must not only facilitate the combination of the human and the robot actions (i.e. for sharing), it must also allow for the arbitration of their actions (i.e. for trading).

Finally, a unified general framework is needed to address the role of sharing and trading in accordance to humans' interacting with current state of robots. This is essential due to the progressive introduction of more intelligence and autonomous robots equipped with powerful and versatile mechanisms for interacting with humans in different robotics applications (Haegele et al. [29], Menzel & D'Aluisio [30]). Technological advances give rise to new HRI requirements to consider how humans and robots might share and trade seamlessly. Hence, a formulation for sharing and trading to address this research need is useful in robotics to assist in the design and development of different types of HRS for different applications.

1.2. Objective

This thesis addresses the concept of seamless HRI and system design issues raised by sharing and trading in an HRS (i.e. a telerobotics system). The goal is to formalise a conceptual framework to describe the concept of sharing and trading that takes into considerations both the unique strengths of the human and the robot. This research work emphasises the importance of modelling a conceptual framework of sharing and trading as a foundation for achieving seamless HRI, and the central role of sharing and trading in the design and development of a cooperative HRS. To achieve, it is important to address the fundamental issues first so as to establish a research basis in this area. Hence, the overall objective of this research work is the study of the fundamental issues that

constitute the development of a framework of sharing and trading to assist in the design and development of an HRS for seamless HRI. The study of the achievement of seamless HRI, via sharing and trading, can lead to a contribution towards the knowledge of how human and robot might cooperate in an HRS.

1.3. Research Approach

The development of the framework of sharing and trading is divided into two stages. The first stage is the identification of the basic elements for defining what can be shared and traded between human and robot in an HRS. The elements are *human control*, *robot autonomy* and *information*, identified based on the requirements for facilitating HRI. They are: (1) how the robot is controlled from the perspective of human interacting with the robot; (2) the required degree of robot autonomy from the perspective of the robot interacting with the human; (3) the issues of communication between the human and the robot. The second stage is the formalisation of the concept of sharing and trading. Here, the concept of sharing and trading that is formulated is in the context of a task, called task sharing and trading ($T_{S\&T}$). By *task* implies the required human's and robot's functions and the goals they are attempting to accomplish. The approach towards the formalisation of $T_{S\&T}$ is based on the characterisation of basic task activities (e.g. desired input tasks, human tasks, robot tasks, etc.) within an HRS. The HRI strategies invoked by $T_{S\&T}$ are based on how human and robot assist each other using different types of human-robot roles. Consequently, to study how the concept of sharing and trading can be applied, it is employed in the modelling and implementation of a telerobotics system.

To provide realistic experimental settings, researchers working in telerobotics need to develop “physical system” to facilitate the study of HRI. According to Fong et al. [17], HRI is strongly related to Human-Computer Interactions (HCI). In HCI, the focus is in the design, implementation and evaluation of interactive computing systems, which includes user interfaces and input devices for human use (Hewett et al. [31]). However, HRI is fundamentally different from HCI because it concerns physical intelligent systems operating in changing, real-world environments, equipped with complex dynamic control systems, and imbued with certain levels of intelligence (e.g. decision-making, reasoning, etc.) and autonomy (Fong et al. [17]). Hence, as compared to HCI, conducting research in HRI will be more challenging because it not only concerns the

use of user interfaces by humans; it also involves the use of interfaces for humans-robots communications. In addition, it also concerns the design and implementation of physical robots that can interact with humans. Therefore, for conducting research in telerobotics, it will be beneficial if there is an easy to use and safe front-end testbed that facilitates humans' interactions with physical robots. For this research, the HRI testbed developed in Robotics Research Centre (RRC) is employed. An overview of the testbed is presented in Figure 1.5. The purpose of this testbed is for studying HRI in a working environment that comprise of both humans and robots working together under human supervision (e.g. the scenario depicted in Figure 1.1(f)). The detail implementation of this testbed is presented in the paper by the author in Ong et al. [32].

Figure 1.5: Block diagram of the HRI testbed

Based on this testbed, sets of tasks have been designed for conducting the experimental study. These tasks are robot-assisted teleoperation, single and multiple waypoints navigation, landmarks navigation, vision-based tasks (such as people and robot following), human-assisted localisation. The aim is to assess how seamless HRI can be achieved via the concept of sharing and trading. The assessments are based on how a robot assists human in a seamless way and how human assistance can be provided

to a robot seamlessly in different task situations respectively.

1.4. Thesis Outline

This thesis is structured as follows. Chapter 1 has identified the need of a conceptual framework for seamless HRI. The main approach for the development of this framework is based on the concept of sharing and trading. In Chapter 2, a critical review of existing work on sharing and trading is provided. The review is based on the survey of major research thrusts and technologies related to sharing and trading in the current literature of robotics. This encompasses a detailed discussion of different HRI paradigms and the essential interaction requirements invoked by each paradigm. The review supports the research problem raised in Chapter 1 that a general framework for seamless HRI based on sharing and trading is missing and hence reaffirms the need to develop such a framework. Based on the concept of task allocation, Chapter 3 describes the concept of sharing and trading. Here, key elements involved in the process of sharing and trading between a human and a robot are identified and characterised. The outcome is a conceptual framework of sharing and trading to assist in the design and development of a cooperative HRS. Consequently, to study how the concept of sharing and trading can be applied, the framework is employed in modelling of a telerobotics system in Chapter 4.

Based on the modelled telerobotics system, Chapter 5 depicts the implementation of this system. The system development described here serves as the experimental platform to assess how seamless HRI is achieved via the concept of sharing and trading established in Chapter 3. Subsequently, the experimental studies on how a robot assists human (RAH) and how human assists robot (HAR) are presented in Chapter 6. The experiment conducted on RAH assessed the cooperation between the human and the robot based on the concept of sharing. On the other hand, the experiment conducted on HAR assessed the cooperation between the human and the robot based on the concept of trading. In evaluating the concept of seamless HRI due to change of different human-robot roles, the experimental evaluation on RAH addresses human-robot roles transition with same task specification, while the experimental evaluation on HAR addresses human-robot roles transition with completely different type of task specification. The findings in Chapter 6 show that the RAH-HAR paradigm invoked by the concept of sharing and trading for seamless HRI provides a useful approach for human-robot

cooperation in compensating each other's deficiencies in different task situations and task demands. Finally, Chapter 7 concludes this thesis by discussing the essential contribution and proposed future work of this research work.

1.5. An Overview of the Research Contributions

The primary contribution of this research is to provide a conceptual framework to implement flexible and adaptive HRI based on how a human and a robot assist each other through sharing and trading. Research contributions include the following:

- A formalisation for sharing and trading based on a paradigm of Robot Assists Human - Human Assists Robot (RAH-HAR) to address contingencies that emerge when a human and a robot work together during task execution. This is achieved via letting both human and robot accomplish task objectives by assisting each other in different ways to fit situational needs and capabilities change throughout a task. This formalisation is presented in Chapter 3. The formalisation is useful for the initial design and development stage of an HRS, because it provides a flexible task allocation scheme in deciding which system-level tasks should be performed by a human, by a robot or by a combination of both during task execution. In other words, the solution invoked facilitates HRI through the considerations of timeliness and pragmatism of a situation for making provisional task allocation decisions between the human and the robot. Such a formalisation for explaining how sharing and trading is applied in the design and development of a cooperative HRS is lacking from the literature of robotics that uses the concept of sharing and trading for seamless HRI.
- The “physical” implementation of a shared and traded control architecture based on the paradigm of RAH-HAR that supports a spectrum of interaction modes using different coordination mechanisms. This is presented in Chapter 4 and 5 respectively. The key idea of the development of different interaction modes is based on the different human-robot roles and relationships strategies as depicted in Figure 1.2. This implies that instead of using only one fixed role and relationships strategy, both human and robot can engage in different roles and relationships to compensate for the unique limitations possessed by each other. This is achieved via the establishment of a highly flexible communication framework that allows both human

and robot to directly exchange goals and intended actions either *implicitly* (for sharing) or *explicitly* (for trading) at any circumstances. As a consequence, the implementation of the HRS architecture shows how sharing and trading are applied in conjunction.

- Provided proof-of-concept experiments to assess how seamless HRI can be achieved through sharing and trading. This is presented in Chapter 6. The experiments conducted for sharing were based on how a robot assists human in a seamless way in accordance to the human demands and needs. On the other hand, the experiments conducted for trading were based on how human assistance can be provided to a robot seamlessly in different task situations. Such experimental evaluation for assessing seamless HRI via sharing and trading has been found to be lacking in the current literature of robotics and HRI.

Chapter 2

Literature Review

In Chapter 1, the need of a general framework of sharing and trading was identified and defined. The purpose is to assist in the design and development of an HRS for seamless HRI. To reiterate, seamless means *flexibility* in human control for the human to interact with a robot in different situations and the *adaptability* of the robot autonomy in response to the human control. The purpose of Chapter 2 is to present a review of the existing work of sharing and trading in robotics in order to affirm the identified research problem. The review is based on the interaction strategies (i.e. how a human and a robot may work together) and requirements derived from the HRI roles and relationships in different types of HRS given in Section 2.1. While the review shows that there are many works applying the concept of sharing and trading as a basis for HRI, there is a lack of an agenda to synthesise the key elements of the process of sharing and trading in a holistic manner, let alone a unified general framework of sharing and trading. This review is important as it therefore suggests that there is a need to develop a framework of sharing and trading in robotics that may form the foundation to study how a human and a robot can interact seamlessly in an HRS, in particular a telerobotics system. Such a framework could then be used to develop a cooperative HRS. Hence, the purpose of this chapter is essential in establishing this need to develop a unified general framework of sharing and trading in robotics.

2.1 Human-Robot Systems

Current HRS takes many forms. This can range from manually controlled systems, such as teleoperation (Sheridan [2]) to autonomous robotics system that employ artificial intelligence, machine perception, and advanced control (Giralt et al. [13]). A simple illustration of this spectrum is presented in Table 2.1. The illustration in Table 2.1 is based on the review of literature of different robotics applications, presented in order of increasing robot autonomy/intelligence.

As shown in Table 2.1, Type 1 represents traditional teleoperation systems. Type 2 represents teleoperation system that employs video technology, computer technology and

force feedback. This facilitates a finer-gain of control (as compared to Type 1) for performing more complex/intrinsic tasks. Type 3 represents telerobotics system, the target HRS for this research. As compared to Type 1 and 2, the robot is not directly teleoperated throughout the whole work cycles, but can operate in continuous manual, semi-autonomous or autonomous modes depending on the situation context.

Table 2.1: Different types of Human-Robot Systems

Type	Human-Robot System	Descriptions	Possible Applications
1	<p>Teleoperation System (not computer-aided)</p>	The human is located remotely from the robot via the use of electric cable. However, the robot is directly controlled by human supervisor's own visual senses (line of sight). The robot extends the human's manipulation capability to a remote location so that he can work safely from the hazardous environment.	Underwater cleaning of reactor vessels, pipe inspection, etc. in nuclear power industry [6].
2	<p>Teleoperation System (computer-aided)</p>	An extension of Type 1, but the human controls the robot through artificial sensing, computer, and displays. The robot extends both the human sensing and manipulation capabilities.	Robotics Surgery (e.g. the Da Vinci™ Surgical System [33]), underwater operation [34], etc.
3	<p>Telerobotics System (An extension of Type 2)</p>	Here, the human and the robot are separated by a barrier (environment, distance, time, etc.) that prevents direct interaction. In addition, the robot is normally equipped with high level of intelligence (such as safe navigation, path planning, etc.) while receiving higher-level instructions from the human instead of under continuous manual control as in Type 1 and 2.	Space exploration [3], search and rescue [7], military operation [8], automated security [11], etc.
4	<p>Intelligent Mobility System</p>	A variant of Type 3, but the human and the robot are located closed together.	Rehabilitation, such as intelligent wheelchair [20, 35] or mobility support system [36].
5	<p>Work Partner</p>	A variant of Type 3. Robot is equipped with powerful and versatile mechanisms to communicate, interact and cooperate with human in a natural and intuitive way.	Robot as work assistants in factories [37] and in space [38], caretaker in home [29], etc.
6	<p>Autonomous Robot</p>	A variant of Type 3 in autonomous mode. Robot replaces the human and performs the desired tasks autonomously.	Intelligent vacuum cleaner [39], tour guide [40], etc.

The concept of telerobotics (i.e. Type 3 HRS) has influenced the development of different HRS (e.g. Type 4-6) for different applications. Type 4 is a form of Type 3 configuration with an important difference: the human located on the robot mobile base (e.g. the wheelchair), has direct visual awareness of the robot environment. The research

in this domain focuses on the development of safe and reliable robotics system that can support handicapped or elderly people in their daily lives. Type 5 is a more sophisticated form of Type 3 configuration, a highly autonomous and intelligent robotic system that has the capability to work cooperatively with humans. For example, based on the concept of telerobotics, Suomela and Halme [37] have developed a cooperative HRS to enable the communication between the human and the robot at a high level of abstraction. Commands and supervision are made by speech and gestures – the most natural communication for humans. Finally, Type 6 represents fully autonomous robotic system that can operate without any human guidance and control. This HRS can be considered as another form of Type 3 HRS where the robot is operating in autonomous mode. As HRS of Types 4-6 needs to operate in the direct vicinity of humans, their malfunction could cause severe harm to people. Therefore, such HRS has to be considered as safety-critical systems. To ensure this, techniques to define safety requirements and to prove the satisfaction of these requirements during HRI must be considered when developing such HRS [20, 29, 37, 40].

The discussion of the six types of HRS above provides an overview of how a human and a robot may interact in different applications. This gives rise to a basic concern, which is what is the nature and manner of the interaction roles that a human and a robot might play in an HRS. Do all the six types of HRS discussed above have been designed based on the same interaction role? If not, what are the interaction roles of the human and the robot? More specifically, what are the human-robot roles and relationships in an HRS? This concern is currently identified as one of the primary research problem in the domain of HRI (Murphy & Rogers [16]), namely *modelling of HRI* (Section 1.1, Table 1.1). The purpose is to provide a basis of how a human and a robot should interact so as to characterise different types of HRI strategies. The five human-robot roles and relationships (Section 1.1.1, Figure 1.2) considered in this thesis are:

- ❑ Master-Slave
- ❑ Supervisor-Subordinate
- ❑ Partner-Partner
- ❑ Teacher-Learner
- ❑ Fully Autonomous

The basis for considering these five human-robot roles and relationships are discussed below.

2.1.1 Evolution of Human-Robot Roles and Relationships in Robotics

Master-Slave

According to Norman [41], how human interacts with any technological system (including robot) directly depends upon the human view of his/her relationship with that system. Historically, human normally recognises him/herself as the *master* of the robot (Hancock [5]). On the other hand, a robot is normally view as the *slave* of the human to service the needs and demands of the human. The history of modern robotics application based on this human-robot role and relationship began in the late 1940's, when the first *master-slave* telemanipulator system was developed in the Argonne National Laboratory for chemical and nuclear material handling (Vertut & Coiffet [42]). With this system, the "slave" robot manipulator in the remote side reproduced exactly the motions imposed on the "master" handle by a human operator to perform the hazardous tasks.

Supervisor-Subordinate

As technological advancement offers new kinds and degrees of robot power that greatly expand the potential to facilitate and augment human work activities, it becomes critical to refine the roles and relationships that both human and robot can interact instead of just simply the "master-slave" relationship. In the late 1960's, many researchers and practitioners recognised this potential and started to consider how to improve the human relationship with the robot. Perhaps, Sheridan [2] was one of the first to extend beyond the master-slave paradigm and formalised a new human-robot role and relationship called *supervisor-subordinate* relationship. This human-robot role and relationship is derived from the close analogy between the human supervisor's interaction with human subordinates in a human organisation and a human interacting with robots. A human supervisor gives directives that are understood and translated into detailed actions by human subordinates. In turn, human subordinates gather detailed information about results and present it in summary to the human supervisor, who must then infer and make decision for further actions. Sheridan stated that the human and the

robot can also engage in such relationship but how “involved” the human supervisor becomes in the interaction process is determined by the autonomy/intelligence of the subordinate robot. To date, the majority of research in robotics using this human-robot role and relationship has focused on telemanipulation for process control [42] and also teleoperation of mobile robots for space exploration [3], search and rescue [7], military operation [8], automated security [11].

Partner-Partner

The master-slave and supervisor-subordinate relationship in HRS is hierarchical, with the human always being superior and the robot always subservient. In the early 1990s, researchers began to look into other human-robot role and relationship that is non-hierarchical, where the nature of interaction between the human and the robot is likened to a *partner-partner* relationship. Perhaps, one of the first to design an HRS (i.e. a telemanipulation system) based on this perspective is from Lee [19]. He stated that to let a robot work together with human in an HRS, the robot should not be viewed as a slave or subordinate of the human, but rather as an active partner of the human. In particular, taking the full advantage of the robot capabilities to let the robot supports the human perception, action and intention. This was purported by Fong [43] that to develop a cooperative HRS, the human and the robot should work as partner to exchange ideas, to ask questions, and to resolve differences just as in human-human interaction. He stated that: “*instead of the human always being completely in charge, the robot should be more equal and can treat the human as a limited source of planning and information*”. To date, the *partner-partner* human-robot role and relationship is widely adopted in the area of rehabilitation to let the robot work as a partner of the human so as to provide appropriate assistance to him/her [35, 36]. One example in rehabilitation is from Bourhis & Agostini [20] that uses the *supervisor-subordinate* paradigm to let the human work cooperatively with a robotics wheelchair. In their work, the cooperation between the human and the robot are based on the idea that both the human and the robot can be supervisor of each other for overriding each other actions.

Teacher-Learner

Teaching a robot through a human teacher has been widely studied since 1970s [44].

For example, human have performed the role of a teacher in the domain for robot manipulators. In this domain, the robot as a learner normally learns its trajectory either through a teaching pendant or actual guidance through a sequence of operations given by a human. With recent advances in the theory and practice of robotics, this approach has been extended to allow the robot learner to learn from the interaction at the human teacher's high level of abstraction (e.g. by demonstration [22]). It lets the robot learns up to the point at which the robot is able to carry out complex task and request appropriate help. Currently, this human-robot role and relationship is widely used in HRI to *enhance* the interaction between the human and the robot. This is because researchers in HRI recognise that effective HRI not only requires technological intelligence of the robot but also a "knowledge" transfer between the human and the robot during operation so as to let the robot learns more difficult or poorly defined tasks (Haegele et al. [29]).

Fully Autonomous

Since the days of the Stanford cart and the SRI's Shakey in the early 1970's, the goal of building fully autonomous system has been what researchers in robotics have aspired to achieve (Arkin [45]). To date, cleaning robots (e.g. intelligent vacuum cleaner) are among the first members of the autonomous robot family to reach the marketplace with practical and economical solutions (Fiorini & Prassler [39]). In such HRS configuration, once the human has specified a goal for the robot to achieve (e.g. "Clean Area A"), the robot operates independently. As the robot performs the task, the primary role of the human is to monitor the robot execution.

Human-Robot Team (HRT)

Each of the five human-robot roles and relationships discussed above is important, since each stresses a different aspect of the interactions between the human and the robot (summarised in Table 2.1). It will be beneficial if the advantages of these five human-robot roles and relationships are considered to assist in the design and development of an HRS. This implies that instead of using only one fixed role and relationship, multiple interaction roles and relationships (Table 2.1) are envisaged to let human and robot to work as a team. This idea is analogous to a human-human team where each team member usually does not engage in a single role when they work together. Within a

human-human team, they normally engage different roles based on their task skills and their changes in interaction roles to meet new and unexpected challenges (Chang [46]).

Table 2.2: A summary of the five human-robot roles and relationships

Human-Robot roles and relationships	Characteristics	Since
Master-Slave	To let the robot mimics the human actions exactly in performing a task.	1940s
Supervisor-Subordinate	To let the robot performs a sequence of tasks planned by the human.	1960s
Partner-Partner	To let the robot supports the human perception, action and intention in performing a task.	1990s
Teacher-Learner	To let the robot learns how to perform a task from the human.	1970s
Fully Autonomous	To let the robot performs a task independently for the human.	1970s

The idea of getting a human and a robot to work as a team is not novel but the use of multiple interaction roles and relationships and roles changing during operation is. The concept of Human-Robot Team (HRT) in the literature basically refers to human and robot adopting a partner-partner role and relationship described earlier (e.g. [19, 38, 43]). One notable exception is the work from Bruemmer et al. [47]. They explored the concept of HRT where each team member has the ability to assume initiative within a task. They state that to achieve this, both the human and the robot must have equal responsibility for performance of the task, but responsibility and authority for particular task elements shifts to the most appropriate member, be it the robot or the human. To facilitate, they suggested four roles for each member of a human-robot team as depicted in Table 2.3. These roles are derived from the supervisor-subordinate human-robot role and relationship discussed earlier.

Table 2.3: Four human or robot roles in a HRT suggested by Bruemmer et al. [47]

Human or Robot roles	Description
Supervisor (Leader)	Directs the other team member to perform a high level task.
Subordinate	Team member is given a high level task with less direct supervision by a supervisor.
Equal	No direct supervision; each team member is wholly responsible for some aspect of the task.
Tool	Performing a task with direct supervision by a supervisor.

The work by Bruemmer et al. [47] is similar to the concept of HRT envisaged here,

because both use multiple interaction roles and the need of role transitions. However, there are two differences. Firstly, the *type* of human-robot roles and relationships envisaged here not only considered human as supervisor and partner (Table 2.2 *versus* Table 2.3), it also considered human as master and teacher of a robot. Secondly, the work presented in this thesis does not claim that the robot has the responsibility and authority to *direct* human in performing a task. In this work, human retains as the overall responsibilities of the outcome of the tasks undertaken by the robot and retains the authority corresponding with that responsibility even though the robot may be in the authority to guide certain aspect of the tasks (e.g. correct human control actions).

In short, the concept of HRT envisaged here requires both the human and the robot to engage in multiple interaction roles (Table 2.2) and role transitions during operation. The purpose is to let them perform different type of tasks and to meet new and unexpected challenges with the human maintained as the final authority over the robot.

2.1.2 Summary

This section has established the interaction roles that a human and a robot may play in an HRS (Table 2.2). This is important because the human relationship with the robot can dictate the boundaries and constraints on the interactions between them. In addition, the concept of HRT is introduced to differentiate the work here with other research work that also considers the use of different human-robot roles and relationships. Given the roles that a human and a robot may play in an HRS and the interaction strategies derived from the respective human-robot roles and relationships, the following requirements are considered:

- ❑ *Methods of Human Control:* This determines how a robot is controlled in an HRS from the perspective of human interacting with the robot [2, 16]. The aim is to provide a review of different control strategies that strive to give *flexibility* in human control as human interacts with a robot. This is discussed in Section 2.2.
- ❑ *Robot Autonomy:* This determines the *adaptability* of the robot autonomy by the robot in response to human control [13, 48]. This is discussed in Section 2.3.
- ❑ *Human-Robot Communication:* This determines how a human and a robot communicate [17, 49, 50, 51]. This consideration is discussed in Section 2.4.

2.2 Methods of Human Control

The use of human's adaptive characteristics as a controller has a long history of providing a cost-effective method of increasing system reliability. The key question, over the last few decades, has been the role of human in the control of a system. Should he be an active, serial element in the control loop or should he be a supervisor monitoring the progress of the system (Curry & Ephrath [52])? As a human is a necessary system element in the control loop, effective human control method is important to determine how a human and a robot interact to increase the system performance. HRI practitioners and researchers normally adopt certain paradigms to guide the development of the system. Their interaction approach can be characterised by the interaction roles and relationships between humans and the robots in an HRS as discussed in Section 2.1.1. In accordance to the five human-robot roles and relationships discussed in Section 2.1.1, the form of control derived from the master-slave relationship is normally called *manual control* [42], i.e. human needs to exercise continuous control without any assistance from the robot. On the other hand, the form of control derived from the fully autonomous mode by the robot is normally called *autonomous control* [13], i.e. no human control is required except for starting and stopping the robot. Between the extremes of master-slave to that of a fully autonomous control, there is a spectrum of control options based on supervisor-subordinate, partner-partner, teacher-learner relationships. The form of control is normally known as *semi-autonomous control* [2, 13, 20, 22, 36, 43]. The term semi-autonomous control normally refers to an autonomous robot which can interact intelligently with a human, who might command, modify, or override the robot's behaviour [2, 13]. An overview of the different forms of control based on the five human-robot roles and relationships is presented in Table 2.4.

Table 2.4: Different forms of control derived from the five human-robot roles and relationships summarised in Table 2.2

Human-Robot Roles and Relationships	Form of Control
Master-Slave (Type 1–3 HRS)	<i>Manual Control</i> – This describes the traditional teleoperation system (Vertut & Coiffet [42]). The master-slave operation is the most basic form of control, where the human must always remain continuously in the control loop. The operating principle is simple; i.e., human (master) has full control of the robot (slave), e.g. all the control decisions will depend on the human. When human stops, control stops.
Supervisor-	<i>Semi-Autonomous Control</i> – Here, the robot does not simply mimic the human's

Subordinate (Type 3 and 4 HRS)	movements as in the Master-Slave. Instead, the worker robot has the capability to plan and execute all the necessary intermediate steps, taking into account all events and situations with minimum human intervention. On the other hand, the human as a supervisor divides a problem into a sequence of tasks, which the robot performs on its own (Sheridan [2]). If a problem occurs, the human supervisor is responsible for finding a solution and devises a new task plan.
Partner-Partner (Type 3-5 HRS)	<i>Semi-Autonomous Control</i> – Here, the robot is viewed as the human's work partner and is able to work interactively with the human. Both the human and robot are able to take advantage of each other and to benefit from each other's advice and expertise (Bourhis & Agostini [20], Wasson & Gunderson [36], Fong [43]). As compared to the Supervisor-Subordinate role and relationship, if a problem occurs, the robot may provide the necessary assistance to find a solution (Fong [43]).
Teacher-Learner (Type 3-6 HRS)	<i>Semi-Autonomous Control</i> – This assigns the human a primary role of teacher or demonstrator and assumes that the learning robot possesses sufficient intelligence to learn from him (Nicolescu & Matarić [22]). Once the robot is able to handle the task, it can replace the human completely or work together with the human depending on the context of the application.
Fully Autonomous (Type 3-6 HRS)	<i>Autonomous Control</i> – Here, the aim is to develop robotics system that has the capabilities to operate without any human intervention once the control is delegated to the robot (Giralt et al. [13], Burgard et al. [40]). This implies that the human can only monitor but not influence the robot operation. The only control intervention is to stop the robot operation when a potentially serious error occurs.

The solution for the concept of semi-autonomous control comes from two main stems (Murphy & Rogers [53]): the *teleoperation* concept (Sheridan [2]) and the *autonomous robot* concept (Giralt et al. [13]). According to Giralt et al. [13], in the teleoperation concept, both the human and the robot interact at the human operator station level. On the other hand, in the autonomous robot concept, the focus is to have on-board, in-built intelligence at machine level so that the robot can adapt its actions autonomously to the task conditions during HRI. Although the semi-autonomous control concept may emerge from the two mentioned stems, the basic objective remains the same. That is, in order to advance beyond simple human control of a robot there is a need to provide the robot basic competence and a degree of autonomy (see Section 2.3). This leads to a reduction in the degree of supervision by the human [2, 13].

2.2.1 Control Framework

Section 2.2 has provided an overview of the different forms of control, namely *manual control*, *semi-autonomous control* and *autonomous control* derived from the five human-robot roles and relationships (Table 2.4). However, to look into how human controls a robot to perform a task in each form of control, there is a need to provide a *control framework* to describe the nature of the interactions between the human and the

robot. This is depicted in Figure 2.1a to 2.1f⁶. In this context, *task* refers to the HRS task, e.g. manipulate an object, welding, monitor an area (surveillance), etc. The human cannot perform the task directly, but must perform the task via two main interaction loops. One loop defines the interaction between the human and the robot via an interface. The second loop defines the interaction between the robot and the task via its sensors and actuators. The “intermediary” that facilitates the interaction between these two loops is the control mode. Here, each control mode is viewed as a “task interaction mode” for human to interact with the robot in performing a task. Figure 2.1a represents traditional master-slave manual control system (Type 1). Figure 2.1b represents indirect (i.e. with computer-aided) master-slave manual control system (Type 2). Figure 2.1f represents autonomous control for fully autonomous robot (Type 6). Figure 2.1c to 2.1e represents semi-autonomous control system (Type 3-5) based on supervisor-subordinate/partner-partner/teacher-learner human-robot roles and relationships. Figure 2.1c depicts the parallel type of semi-autonomous control; Figure 2.1d depicts the serial type; and Figure 2.1e depicts a combination of both parallel and serial types. These three types of semi-autonomous control system are discussed in the following sub-sections.

Figure 2.1a: An exposition of direct master-slave manual control with no computer-aided assistance

⁶ Figures 2.1a, 2.1b, 2.1e & 2.1f are adapted and modified from Sheridan [2], while Figure 2.1c and 2.1d are adapted and modified from Yoerger & Slotine [54] and Yasuyoshi et al. [55].

Figure 2.1b: An exposition of indirect master-slave manual control with computer-aided⁷

Figure 2.1c: An exposition of semi-autonomous control - parallel type

⁷ Computer-aided in the context of providing human with better display (e.g. fusing of different sensors information, virtual reality display, etc.) and multi-modal control inputs (e.g. joystick with force feedback, keyboard, mouse, speech, etc.) for controlling the robot. This also applied to Figure 2.1c to 2.1f.

Figure 2.1d: An exposition of semi-autonomous control - serial type

Figure 2.1e: An exposition of semi-autonomous control - parallel and serial combined type

Figure 2.1f: An exposition of autonomous control

Semi-Autonomous Control - Parallel Type (Shared Control)

In the parallel type (Figure 2.1c), both manual control and autonomous control operate at the same time. The parallel type is normally referred to as *Shared Control*. To reiterate, an approach to incorporate the capabilities of the human and the robot by letting them control same/different aspects of the system simultaneously in situations that best benefit from partnership [2, 4, 18, 19, 21, 36, 56, 57, 58, 59]. It is normally used in situations where the task is too difficult to be achieved by either the human (via manual control) or the robot (via autonomous control) alone. Shared control has been studied in different forms in both the domain of telemanipulation and teleoperation of mobile robot. The examples include position-compliance control [4, 19], vision-based perceptual guidance control [21, 56], safeguarding control [36, 57, 58] and behavioural control [18, 59]. All these approaches have been based upon some form of coordination (i.e. arbitration and/or fusion) strategy⁸ with respect to the human inputs and the robot own assessment of the environmental task. As compared to manual control, shared control

⁸ As this topic is related to how robot coordinates its processes (i.e. a set of perception-action units) with human control inputs, it is discussed in Section 2.3.4

frees the human's attention from directly controlling nominal activities while allowing direct control during more perceptually intensive activities such as manipulation of parts (e.g. [4, 19]) and navigation in cluttered area (e.g. [18, 20]).

Semi-Autonomous Control - Serial Type (Traded Control)

In this serial type (Figure 2.1d), either manual control or autonomous control can be selected as the operating mode at any one time. The serial type is normally referred to as *Traded Control*. To reiterate, a mutually exclusive approach for the human and the robot to exchange control in performing a task based on the following situations:

- *Task initialisation*: In this context, human passes control to the robot to perform a task autonomously once the human has initialised the task (e.g. [2, 27, 56, 60]). Normally, traded control is employed as a means for the human teacher/supervisor to break down a complex task (before actual task execution) into a set of simpler tasks for the robot to perform autonomously. For example, to teach a robot to perform a visual tracking telemanipulation task (Papanikolopoulos & Khosla [56]), the human teacher first teaches the robot by positioning the manipulator with respect to the target and selects a number of features for tracking. Once the robot learns how to track the target, the human teacher then hands over the control to the robot to perform the task autonomously. In performing a long distance navigation task of moving from point A to point B, the human supervisor assists the robot by designed a set of waypoints for the mobile robot to navigate (Lin et al. [60]). Once the waypoints are specified, the human supervisor then hands over the control to the robot to perform the task autonomously.
- *Emergency*: In this context, human takes control over from the robot to resolve evolving problems. For the visual tracking telemanipulation task described above, emergency situations include loss of one feature due to occlusion by an unknown object and singular configuration of the manipulator resulting the robot's failure to track the target [56]. As for the navigation task above, emergency situations include getting struck in the environment and sensors drift causing the robot in failing to reach the target points [60]. For both cases, the human takes control and tries to solve the problems manually. Once the problems are solved, the human then hands the control back to the robot again to perform the desired task autonomously.

- ❑ *Opportunity* for improvement: In this context, the human takes the control over from the robot when he/she recognises the opportunity to improve the task performance even though the robot is performing well (Kortenkamp et al. [27]).
- ❑ *Task interruption*: In this context, the human takes the control over from the robot when he/she does not want the robot to continue with the current task (e.g. [2, 27]).
- ❑ *Task completion*: In this context, the human takes the control over from the robot once the robot has completed the task (e.g. [2, 27, 56, 60]).

The discussion above provides an overview of how traded control gives human the flexibility of delegating task to a robot. However, using this approach alone is not sufficient for tasks that require the robot to perform a task autonomously but at the same time requires the human to provide control input to the robot from time to time to keep the robot focused on its goal. These types of tasks require the use of both sharing and trading of control. This is discussed below.

Semi-Autonomous Control - Combine type (Shared and Traded Control)

In the combined configuration (Figure 2.1e), both serial and parallel types interact to an extent, where the subtasks within each mode may also be shared and traded. A classical control framework of the combined type is the Supervisory Control (SC) by Sheridan [2], based on the Supervisor-Subordinate paradigm (Section 2.1.1). SC has been around since 1967. It was first introduced by Ferrell and Sheridan [61] from the research on earth-based teleoperation of lunar vehicles. Although the control framework is termed SC, Sheridan notes that the human is not restricted only to a supervisory role. Instead to perform different types of tasks, the human may engage in a non-cooperative or cooperative relationship with the robot to share and/or trade control (as discussed above) with the robot. This purports that the human and the robot may engage in different roles when they work together as discussed in Section 2.1.1. However, what specific roles a human and a robot may engage when they shared and traded is not discussed in (Sheridan [2]).

To date, SC is the most widely used paradigm in the domain of telemanipulation (e.g. [4, 27, 55, 56]) and teleoperation of mobile robot (e.g. [18, 20, 21, 60]). The common approach towards the implementation of SC for an HRS is to implement a range of

control modes ranging from manual to autonomous control for a human and a robot to share and trade control (e.g. [4, 20, 21, 27, 55, 56, 60]). In this context, an essential consideration for SC is how to provide varying degree of sharing and trading of control between the human and the robot [19, 43, 47]. Even though SC has been studied since the 1960's, this continues to be one of the open issues. To solve this problem, many researchers have proposed various control approaches to envision a tighter cooperation between the human and the robot, where the interactions are more mixed initiative, with all parties (both human and robot) contributing according to their capabilities. Three control approaches since SC, namely *Interactive and Cooperative Sensing and Control* [19], *Collaborative Control* [43] and *Mixed Initiative Control* [47] are discussed below.

Interactive and Cooperative Sensing and Control (ICSC)

ICSC is an extension of SC based on the Partner-Partner paradigm by Lee [19]. The aim of ICSC is to make use of the full advantage of robot capabilities to support a human in performing dexterous manipulation. Although this work was proposed 12 years ago, it remains one of the important works in the domain of telemanipulation for considering how the capabilities of a human and a robot can be integrated (Dorais & Kortenkamp [62]). The solution provided by Lee of integrating the human and the robot capabilities is an implementation of an HRS architecture.

According to Lee, to provide varying degree of sharing and trading of control between a human and a robot, the concept of sharing and trading invoked by Sheridan [2] for SC (discussed in Section 1.1.1) is not sufficient. He stated that although the concept of sharing and trading of control by Sheridan is useful in providing a good perspective of how to “combine” the capabilities of a human and a robot, it does not present a general and powerful concept of “integrating” the human and the robot. In the context of Lee's work, by *integrating* means to *mix and/or fuse* human control and robot autonomy so as to address contingencies arises during task execution when the human and the robot share and trade. The reason why Sheridan's concept cannot address this is because his concept is meant for explaining the different type of physical cooperation between a human and a robot (Inagaki [23]), and is not meant for describing the *process* of sharing and trading between the human and the robot during task execution. This purports the claimed in Section 1.1.4 that there is a need of synthesising the key elements

involved in the process of sharing and trading.

Collaborative Control (CC)

CC is an extension of SC based on the Partner-Partner paradigm by Fong [43]. According to Fong, CC is proposed as a new control approach for teleoperation of a mobile robot that allows both human and robot to participate in task performance, and facilitates HRI. In CC, a human and a robot collaborate to perform tasks and to achieve common goals. Instead of a supervisor dictating to a subordinate, the human and the robot engage in *dialogue* to exchange ideas, to ask questions, and to resolve differences in a partner-partner like manner.

The aim of Fong's work using CC is for designing better human-robot interfaces for the teleoperation of mobile robot. Particularly, Fong's work investigates how to provide a human with a better understanding of a robot's task environment and state, and also means of allowing the human to issue more abstract commands to the robot. Fong believes that a robot is able to take advantage of human capabilities (perception, decision making, etc.) and to benefit from human advice and expertise. However, to do this, the robot needs to function not as passive tool, but rather as active partner. They need to have more autonomy to interact with human, instead of merely waiting for or blindly executing human commands.

According to Fong, the essential difference between CC and SC is that it can adjust its method of operation based on situational needs so as to enable "*fine-grained sharing and trading of control*" for human to interact with a robot in different task situations. Specifically, in a situation where the robot does not know what to do or is performing poorly, the robot has the option to adjust its autonomy by giving control (e.g., for decision making) to the human in that situation. In the context of Fong's work, sharing of control is in the form *safeguarding* control (discussed earlier in semi-autonomous - parallel type section), in which the robot follows the human's control but ignores those control inputs that would, in the robot's assessment, endanger the robot. On the other hand, trading of control is in the form of *information* trading using dialogue as a framework for coordinating human control and robot autonomy during task execution.

Mixed Initiative Control (MIC)

MIC is an extension of SC based on the concept of HRT (see Section 2.1.1) by Bruemmer et al. [47]. Their work focuses on designing better human-robot interfaces and developing autonomous robotic system architecture for the teleoperation of mobile robot. According to Bruemmer et al., the essential difference between MIC and SC is that a robot can adopt the role as supervisor of human (see Section 2.1.1). The aim of MIC is to allow multiple levels of human control intervention into the functioning of an autonomous robotic system. Specifically, to provide appropriate degree of sharing and trading of control between a human and a robot, human control intervention must be incorporated as an *integral* part of the robot's control architecture (discussed in Section 2.2.3). On the other hand, the robot must have the ability to accept different levels and frequencies of human control intervention by adapting to its degree/level of autonomy. To facilitate, Bruemmer et al. stated that the robot not only require to be aware of the limitations of its autonomous capabilities and query for appropriate assistance from a human (as in Fong work [43]); the robot must also have the ability to recognise when assistance is needed from the human and learn from the interactions with the human. However, these "ideas" are still under research and development by Bruemmer et al. research group. To Bruemmer and his co-researchers, sharing of control is in the form behavioural control (discussed earlier in semi-autonomous - parallel type section) using subsumption logic (discussed in Section 2.3.3) for control coordination. On the other hand, trading is via switching of control modes by the human.

Summary

Common to all the three control approaches, namely ICSC, CC and MIC (i.e. semi-autonomous control - combine type) reviewed above is to provide human the *flexibility* to control a robot in different task situations via varying degree of sharing and trading of control. This is important because it highlights the potential of using sharing and trading as an approach for seamless HRI proposed in Section 1.1.1. However, as defined in Section 1.1.1, to achieve seamless HRI, it not only requires providing human with control flexibility but also the *adaptability* of robot autonomy for the robot to response to the human control. Human control and robot autonomy are two separate issues but are related and must be considered together when designing an HRS. In particular, the issue

is how these can be integrated in the robot's control architecture as stressed by Lee [19] and Bruemmer et al. [47]. This concern purports that to achieve seamless HRI, there is a need to look into the autonomy of a robot and how human can be *integrated* into the robot control system apart from considering different methods of human control as discussed in this section. This concern is further discussed in Section 2.3. In addition, the discussion in this section has also raised several concerns regarding the achievement of seamless HRI via sharing and trading as follows:

First, to provide human the flexibility to share and trade with robot in different situation, there is a need to implement a range of control modes (e.g. ranging from manual to autonomous control) in an HRS. This raises a basic question, i.e. what constitutes the design of a particular control mode. To reiterate, a control mode is viewed as a task interaction mode for human to interact with the robot in performing a task. This concern is important because it provides insight for looking into how a particular control mode is envisaged to let human controls the robot. To facilitate, an attempt to identify the features and characteristics of control mode is presented in Section 2.2.3.

Second, the need of using different control modes raises the issue of control modes transitions. Particularly, this concern is important in providing insight into how human and robot trade control. This is discussed in Section 2.2.3.

Third, the reviewed of Fong work [43] (i.e. CC) has shown another facet of trading, i.e. trading of *information* using dialogue as a mechanism for control trading between a human and a robot. However, this process also involves the phenomenon of sharing, in the form of information sharing. This is because when human exchange information with a robot, he/she must first be aware of the robot task environment and state. To facilitate, the robot must share task information with the human. This concern pertains to the communication between the human and the robot is discussed in Section 2.4.

2.2.2 Control Modes

As discussed in Section 2.2.1, to facilitate varying degree of sharing and trading of control in an HRS, an approach is to develop a control architecture that provides a range of control modes (i.e. from the continuum of manual control to autonomous control

presented in Figure 2.1a to 2.1f) for the human to interact with the robot. The purpose of each control modes can be viewed as a *strategy* to adapt to different operational and interaction needs. Adoption of a certain control strategy is required for adequate interaction and appropriate intervention. A control strategy can range from using abstract goal-oriented commands (i.e. high-level commands) to detail descriptions of the task. One reason for using abstract goal-oriented control strategy is to reduce the communication content in situations when the communication delay is high. Here, task is specified in a sufficiently high-level form (i.e. in terms of goals and constraints) where the robot performs the task on its own without constantly requesting guidance/assistance. Examples of high-level abstract goal-oriented commands are: follow the target, grasp the target, etc. Clearly, to perform the task specified in this manner, the robot must have the built-in task knowledge to execute the designated task. In complex task, detailed descriptions of the task can be specified in a hierarchical manner based on the desired goal, e.g. by describing the robot's direction, movement, traveling distance and so forth, in a stepwise manner.

Basically, most of the proposed control modes in the literature [4, 18, 19, 20, 27, 32], have two important features: *complement* and *redundancy*. The control modes are complementary in order to let both human and robot contribute according to their expertise. The aim is to envisage a tighter cooperation, by letting the human and the robot to deal with mutually complementary parts of an HRS task. On the other hand, the control modes are also redundant so as to provide more options for the human to develop strategies (i.e. via a sequences of control modes) to perform the task. By *redundancy* it means control modes that have similar function for letting the human to perform a task. For example in Ong et al. [32], to navigate from point A to point B, the human can have the option to control the robot manually to point B or he/she can specify a set of waypoints for the robot to navigate to point B autonomously.

According to Callantine [63], control modes have four basic characteristics:

1. *Engagement Conditions* – dictate when the mode will engage and encompass target values that must be set so the mode can attain and/or maintain them, and the modes that are currently in use;
2. *Disengagement Conditions* - govern when the mode disengages. A mode may

disengage when another mode is engaged, or when critical target value information no longer applies;

3. *Operation Modifications* – dictate the allowable modifications *to operation* that a human or a robot can make while the mode is engaged;
4. *Control Properties* – state the specific set of parameters (e.g. speed, direction, etc.) that the mode controls, and the manner in which the mode controls them.

2.2.3 Control Mode Transitions

The characteristics of each control modes give rise to specific relationships between modes. Each control modes may have its own set of sub-modes, therefore the sub-modes of a given control modes can interact with the control modes of another. Hence, an important facet of control modes is mode transition. It determines when a particular control mode/sub-modes should be engaged or disengaged. According to Degani et al. [64], a mode transition can result from three types of input: *human initiated*, *robot initiated*, or *mixed initiated* (i.e. from both human and robot).

An effective control mode transition will involve two important attributes, i.e. *monitoring* and *intervention*. Monitoring can be viewed as a precondition for intervention (Sheridan [2]). For example, once a task is delegated to a robot, the human must monitor the robot's operations to obtain adequate feedback on its task performance so as to ensure that it is done properly. Adequate feedback can be achieved via observation to inspection, such as checking the robot agenda, reasoning, plan, etc. The observation can either be by direct viewing or mediated via a sensing device (see Section 2.4.1). If the robot encounters problems during execution, the human monitoring the situation will step in to update the commands or provide guidance to the robot. In cases where the errors cannot be recovered, the human may trade the control over from the robot, by stopping the operation and repairing the robot actions, e.g. via programming of new behaviours that are necessary to accomplish the task.

To classify the different levels of intervention, the classic *Skill-Rule-Knowledge* (SRK) model proposed by Rasmussen [65] that incorporated Skill-Based Behaviour (SBB), Rule-Based Behaviour (RBB) and Knowledge-Based Behaviour (KBB) for Supervisory Control is adopted. This model is adopted because it is the most widely

used formal model for describing human control intervention behaviours (Sheridan [2]). One recent work based on this model is by Sawaragi et al. [66] that use this model for the design of a human-robot interface for the teleoperation of a mobile robot. This model is also adopted by Bourhis & Agostini [20] to characterise both human and robot control behaviours. This model is depicted in Figure 2.2. Based on this model, control intervention may depend on a repertoire of automated behavioral patterns (SBB), a set of cue-action mappings (RBB), or problem solving operations on a symbolic representation (KBB). The level of control intervention used depends on the task to be executed. In addition, it also depends on the type of information provided. For instance, raw sensory data from the robot that can not be perceived directly by the human needs to be reconstructed for human control intervention. In this case, to take appropriate actions, the human has to decide at the level of RBB or KBB, not at that of SBB.

Figure 2.2: Rasmussen's SRK model (adapted from 65)

According to Bourhis & Agostini [20], when a problem arises, the human or the robot may simply use its sensory-motor actions (i.e. the SBB) to react to the situation, or in known situation, standard operation/reaction procedure may be applied (i.e. the RBB). On the other hand, if the situation is unknown to the human, he/she can use all his knowledge to evaluate the situation and make a decision from various goals [2] (i.e. the KBB). This can also be used to describe the robot intervention behaviour. A good example is the application of remote operations where the robot situated at the remote environment is in a better position to give indication to the human if he/she executes the

wrong commands [17, 18, 20, 32]. Another instance is the robot may trade the control over and execute autonomously in situation such as loss of communication (Ong et al. [32]). Depending on the context of the situation, the intervention frequency can range from low to high.

A problem in mode transition is the robot may not be able to keep up with the state of the world or of the task when human intervenes and takes control from the robot (i.e. during trading of control). This can make it difficult and dangerous for the robot to resume its operation once the human delegates the task back to the robot. This is because the robot's models of the world and of the task are inconsistent with the real state of the world (Kortenkamp et al. [67]). In addition, it is also difficult to know when control should be handed over to the robot and when it should be taken back (Sheridan [2]). To overcome, human and robot must share some knowledge of the robot activities during task execution (Jackson et al. [48]). The human must understand the behaviours and the intention of the robot, if he/she wants to intervene to modify or change the mode (Bruemmer et al. [18]). On the other hand, the robot must have the knowledge to interpret the human commands so as to respond to the changes of control mode (see Section 2.3). In addition, it must constantly update its knowledge base so as to keep up with the real state of the world [67]. This implies that it is important for both human and robot to develop a model of the interaction process based upon readily available interaction cues from each other so as to prevent mode confusion. Mode confusion (Bredereke & Lankenau [68]) arises when the mental model of the human does not match the robot's model of the world during HRI.

2.2.4 Summary

An exposition of the essential features for considering how a particular control mode can be designed and control modes transitions are presented in Section 2.2.2 and 2.2.3 respectively. An insight discussion on these two topics (which encompasses different types of control strategy, control intervention, etc.) showed that the designed of an effective control mode for giving human the flexibility to control a robot requires the robot to have the required autonomy to work with the human. This is further discussed in the following section, Section 2.3.

2.3 Robot Autonomy

To respond to the range of control modes and facilitate mode transitions, the robot must have the required autonomy to interact with the human. Here, the term robot autonomy is defined as “*the ability of an agent (in this case, a robot) to act efficiently without any human’s intervention*” (adapted from Braynov & Hexmoor [69]). By stating that a robot is autonomous, it does not mean that the robot is thoroughly self-governing and capable of completing self-planning and self-control. However, it can operate with some known (to the human) level of capabilities in the absence of human supervision or management for a defined period of time (Jackson et al. [48]).

Robot autonomy encompasses two basic attributes (Giralt et al. [13]): *operating autonomy* and *decisional autonomy*. Operating autonomy refers to the basic operational capability (i.e. the technological considerations) of a physical robot. For instance, to be “operational”, a typical mobile robot must be equipped with the following basic components: Adequate sensors for navigation (e.g. range sensors for obstacles avoidance, detection, and location sensors to determine its own location), communication transceivers to interface with the human interface via a communication link, embedded computation and program storage for local control systems (e.g. to interpret commands from the human control interfaces and translate these into signals for actuation). Decisional autonomy refers to the level of intelligence imbued in a robot. This includes an internal representation of the world and of the task, and the capabilities to act reasonably in an unstructured/semi-structured environment. This encompasses the ability to reason about its own action, learn, and adapt to some extent on the basis of human feedback or from its own environment over a given period of time.

2.3.1 Robot Autonomy versus Human Control Involvement

Figure 2.3 presents another view of describing the control modes in Figure 2.1. The basic idea is to set up a discrete scale of control mode, which enables the human to interact with the robot with different degree of human control involvement and degree of robot autonomy. The horizontal axis represents the degree of robot autonomy, while the vertical axis corresponds to the degree of human control involvement.

Figure 2.3: Control modes based on robot autonomy and human control involvement in accordance with varying nested ranges of action of robot

As shown in Figure 2.3, the robot autonomy axis is inversely proportional to the human control involvement axis. Within these two axes, the manual control mode is situated at the bottom-left extreme, while the autonomous control mode is located at the top-right extreme. Between these two extremes is the continuum of semi-autonomous control. Within this continuum, varying degrees of sharing and trading control can be achieved based on varying nested ranges of action as proposed by Bradshaw et al. [70]. They are: *possible actions*, *independently achievable actions*, *achievable actions*, *permitted actions* and *obligated actions*. They are described in Table 2.5. Based on these five actions, constraints can be imposed so as to govern the robot autonomy (e.g. defined using a set of perception-action units) within each level of control modes.

Table 2.5: Degrees of autonomy based on varying nested ranges of action (adapted from Bradshaw et al. [70])

Ranges of Actions	Descriptions
Possible Actions	This refers to the theoretical maximum possible actions a robot can act with its given operating and decisional autonomy.

Independently Achievable Actions	This refers to a subset of possible actions that the robot could be expected to achieve independently with minimum human intervention.
Achievable Actions	This refers to a larger set of actions nested within the range of possible actions that could be achieved by the robot if it is able to work interactively with the human.
Permitted Actions	This refers to the actions nested within the range of possible actions that the robot is allowed to act (i.e. permitted by the human).
Obligated Actions	This refers to a subset of permitted actions that the robot is compelled to act.

Another perspective of relating the degree of robot autonomy to human control is based on Sheridan's [71] ten-level formulation of robot autonomy presented in Table 2.6. This formulation views the robot as a highly intelligent system that is capable of performing a whole task by itself in a given context. Here, the degree of robot autonomy is scaled accordingly based on human "decision" and "approval" when performing the task. Through this, a fine-grained presentation of a continuum of control between the robot and the human can be achieved.

Table 2.6 Ten-level formulation of robot autonomy (adapted from Sheridan [71])

Low ↑ Degree of Robot Autonomy ↓ High	1. Robot offers no assistance: the human perform the whole task	High ↓ Degree of Human Control ↑ Low
	2. Robot may assists by determining the multiple options of performing the task	
	3. Robot assists by narrowing down the options to a few, which human need not follow	
	4. Robot selects one action and the human may or may not approve	
	5. Robot selects action and implements it if the human approves	
	6. Robot informs and allows the human some time to veto task execution	
	7. Robot performs the task and necessarily informs the human what it did	
	8. Robot performs the task and informs the human what it did only if human explicitly requests	
	9. Robot performs the task and informs the human what it did, if it decides human should be informed	
	10. Robot does whole task autonomously	

The discussion in this section has related robot autonomy to human control. However, it does not address how a human control input can be integrated into a robot control system. To facilitate, first it is important to look into robot control architectures to have a basic understanding of how a robot system is designed. This is discussed in Section 2.3.2. Subsequently, issues relating to the coordination of different processes in robot control architectures are discussed in Section 2.3.3. Discussions in these two sections serve as a basis for Section 2.3.4 for describing the incorporation of human

control inputs into a robot system.

2.3.2 Robot Control Architectures

The principal part of a control system in a robot is the control architecture; it is in charge of all the possible movements, coordination and actions of the robot in order to achieve its goal. To facilitate, different approaches for robotic control have been proposed over the years. These approaches can be characterised in accordance to Figure 2.4, which depicts a spectrum of the current robot control strategies (Arkin [45]). The left side represents strategies that employ control through deliberative reasoning and the right represents reactive control.

Deliberative control is purely symbolic and representation-dependent but with high-level of intelligence that often requires strong assumption about the world model the robot is in. This control approach is suitable for structured and highly predictable environment. However, in complex, non-structured and changing environments, serious problems (such as computation, real-time processing, symbol grounding (Brooks [72]), etc.) emerge while trying to maintain an accurate world model. On the other hand, reactive control, which is highly reflexive and representation-free, couples the perception and action tightly, typically in the context of motor behaviours, to produce timely robotic response in dynamic and unstructured worlds (Arkin [45]). However, this approach has its own difficulty, i.e. the problem of designing a set of perception-action processes or behaviours in order to plan and to specify high-level goals.

Figure 2.4: Robot control system spectrum (adapted from Arkin [45])

Many researchers feel that hybrid systems capable of incorporating both deliberative reasoning and reactive execution are needed to deliver the full potential of robotic systems (Arkin [45]). The hybrid control approach is developed based on the advantages of both the traditional deliberative and reactive approaches, and it is the prevalent approach in robotics research today. Basically, this control system consists of three layers: a deliberative layer, a reactive layer and a control execution layer. This is illustrated in Figure 2.5.

Figure 2.5: Hybrid robot control architecture (adapted from Arkin [45])

The knowledge-based deliberative layer, which has a broader perspective and scope, transforms the mission into a set of tasks that performs a plan. The data driven reactive layer on the other hand is highly modular in development and capable of real-time robust performance within a dynamic world that is tightly coupled with arriving sensory data. The task of the control execution layer is to coordinate the processes between the deliberative and the reactive layer. Although this system allows the three layers to run independently, it is considered centralised because it requires all the three layers to work together. Typical examples of the hybrid control system are the Autonomous Robot Architecture (AuRA) from Georgia Institute of Technology [73] and Three-Layer Architecture (3T) from California Institute of Technology [74].

To cope with the complicated processes, the control execution layer is normally decomposed into small independent perception-action processes with limited decision-making capabilities. However, when multiple processes are active some may produce conflicting actions that are not compatible. Hence, a major issue in the design of robot control system is the formulation of effective coordination mechanisms for arbitrating and/or fusing the response of different processes to ensure rational and coherent robot actions (Arkin [45]). This is discussed in Section 2.3.3.

2.3.3. Coordination Mechanisms

Due to the complex interaction between the robot system and the real world, the controversial issue of selecting an emergent action arises when different perception-action processes are coordinated. The selection of an emergent action can be viewed as a product of complexity of the relationship between a robot system and the real world that resists analytical modelling. Thus, ways for resolving conflict are required when multiple processes are active with its own independent response. Basically, there are two predominant classes of coordination mechanisms (Saffiotti [75]). They are: *Arbitration* and *Command Fusion*, respectively.

Arbitration Coordination Mechanisms

This class allows one or a set of processes at any instance to take control for a period of time until another process or a set of processes is activated. Normally, the arbitrator needs to determine the most appropriate perception-action process (or a set) for each situation at hand from a group of competing processes. Hence, it is suitable for coordinating between the set of processes in accordance with the robot's changing objectives and requirements under dynamic conditions. Arbitration coordination mechanisms are further classified into priority-based, winner-take-all and state-based (Pirjanian [76]). Their characteristics and some of their standard approaches are described in Table 2.7.

Table 2.7: Arbitration coordination mechanisms

Mechanisms	Characteristics of coordination	Approaches
Priority-based	An action is selected by a central module based on predefined assigned priorities.	□ <u>Subsumption</u> [77]: This approach uses suppression and inhibition in a descending manner to select the output of a single process. Normally, higher priority process is allowed to take over the control.
Winner-take-all	An extension of priority-based using varying weight. Instead of using fixed assigned priorities, action is selected from the interaction of a set of processes that compete until one process wins the competition and takes over the control.	□ <u>Action Selection Dynamics</u> [78]: Here, processes connected in a form of network actively compete with each other through the use of activation levels (from 0 to 1) driven by both the agent's goals and incoming sensory information. □ <u>Voting</u> [79]: This approach involves processes' generating votes (from 0 to 1) for actions, with the action that receives the most votes being the single process chosen.
State-based	A set of processes that is adequately competent of	□ <u>Discrete Event Systems (DES)</u> [80]: Interaction of process is formalised by the DES formalisation in a

	handling the situation corresponding to the given state is allowed to take over the control.	<p>modular and hierarchical manner. Action selection is done using state-transition, where upon detection of certain event, a shift is made to a new state and thus a new action.</p> <p>□ <i>Temporal Sequencing</i> [81]: A variant of DES. It uses finite-state automation for sequencing between a series of processes based on perceptual triggers that cause transitions from one state to another.</p> <p>□ <i>Bayesian Decision Analysis</i> [82]: Based on Bayesian decision and utility theory. The aim is to choose the action that maximises the expected utility of the process. Action selection of different processes is determined in accordance to their perception actions states evaluated in a cost/benefit manner. The action to be selected is associated with the information provided by the perception actions states.</p> <p>□ <i>Reinforcement Learning</i> [83]: In contrast to the approaches above, the arbitration coordination of the action selection mechanism involves some machine learning algorithm of reward reinforcement. Here, the robot learns the perception-action mapping by exploring actions that lead to some rewards. In this way, actions that get a higher reward are selected and actions with lower reward are suppressed.</p>
--	--	---

Command Fusion Coordination Mechanisms

This class allows multiple processes to combine actions given by all the processes and generates an output to control the robot. Thus, it facilitates all the processes to contribute simultaneously to the control of the robot in a cooperative manner, rather than in a competitive manner. Hence, this class is often suited for control problems, which are multiple objectives in nature. Command fusion coordination mechanisms can be further classified into superposition-based, fuzzy and multiple objectives based. They are differentiated based on the techniques used for: (a) the generation of the recommended actions according to some desired perception-action criteria; (b) the combinations of the recommended actions from all the processes; and (c) the selection of an appropriate action based on the combined recommendations (Pirjanian [76]). Their characteristics and some of their standard approaches are described in Table 2.8.

Table 2.8: Command fusion coordination mechanisms

Mechanisms	Characteristics of coordination	Approaches
Superposition-based	Recommended actions from	□ <i>Potential Fields</i> [84]: This approach uses the total potential from all the processes to make decision, i.e., the action selected

	different processes are combined using linear combinations.	<p>corresponds to the direction indicated by the negated gradient of the total potential from all the processes.</p> <p>□ <u>Motor Schemas</u> [85]: In this approach, a process response of a schema⁹ is an action vector using the potential fields' method. All the relative strengths of each process determine the robot's overall response. Coordination is achieved through vector summation and normalisation.</p> <p>□ <u>Utility Fusion</u> [86]: Here, each robot process is assigned a utility measure. The utilities are combined with measures of uncertainty to evaluate actions based on kinematics and dynamics of the robot.</p>
Fuzzy	Recommended actions from different processes are combined using fuzzy inference techniques	<p>□ <u>Fuzzy Logic</u> [75]: Here, each process is synthesized by a rule base controlled via an inference engine to produce a multi-valued output as a fuzzy membership function that encodes the desirability of each action from the process' point of view. Coordination is performed by combining the fuzzy outputs of the processes using operator such as max operator. Then defuzzification (e.g. via centre of gravity) is used to select a final crisp action ultimately used for control.</p>
Multiple objectives	Recommended actions from different processes are combined using formal decision-theoretic approach.	<p>□ <u>Multi-Objective Coordination</u> [87]: In this approach, multiple processes are blended in a single more complex process that seeks to select the action that satisfies simultaneously all the objectives as much as possible. Coordination is comprised of generating and selecting a feasible set of satisficing solutions known as Pareto-optimal solutions.</p> <p>□ <u>Multi-Objective/Fuzzy</u> [88]: Here, standard fuzzy inferencing approach is replaced with multiple objective decision-making while maintaining the advantages of fuzzy coordination.</p>

Nevertheless, these two classes of coordination methods described above can be integrated if desired.

2.3.4 Incorporation of Human Control Inputs into a Robotics System

In accordance to the control modes and coordination mechanisms depicted in Section 2.2.1 and 2.3.3 respectively, this section provides an overview of how shared and traded control can be implemented in a robot system. The main idea of achieving this is to view human control inputs as one of the system processes in the robot system.

Model of Shared Control

As described in Section 2.2.1, the sharing of control allows both the human and the autonomous function to control the robot simultaneously. The model presented here to facilitate shared control is depicted in Figure 2.6, based on the motor schemas coordination approach (Table 2.8).

Figure 2.6: A model of shared control using motor schemas coordination approach (adapted from Arkin [59])

The overall approach of schema-based robotics is to provide process primitives that can act in a distributed, parallel manner to yield intelligent robotic actions in response to environmental stimuli (Arkin [59]). Each schema operates as a concurrent, asynchronous process initiating motor schemas that react proportionally to sensory information perceived from the environment. All the relative strengths (denoted as $V_1 \dots V_N$) of each process are weighted (denoted as $W_1 \dots W_N$) accordingly to determine the robot's overall response. In this context, shared control involves mixing the human control (e.g. control via an input device such as joystick) signals (denoted as V_H) directly with other processes. Coordination between the human control and other processes signals is achieved through vector summation as depicted in Figure 2.6.

Model of Traded Control

As described in Section 2.2.1, it is inevitable that control responsibilities between the human and the robot must be traded over time to achieve task objectives. One such model to facilitate this is depicted in Figure 2.7, based on the subsumption coordination approach (Table 2.7).

The methodology of the Subsumption approach (Brooks [77]) is to reduce the control architecture into a set of processes. Each process is represented as separate layers working on individual goals concurrently and asynchronously, and has direct access to the sensory information. Following this, human control is also a separated process, viewed as sensor input to the robot system. As shown in Figure 2.7, layers are organised

⁹ Schemas are defined as the mechanism of understanding sensory perception in the process of storing knowledge (Arkin [85]).

hierarchically. Higher layers have the ability to inhibit (I) or suppress (S) signals from the lower layers. In this context, the trading between human and robot can be considered as activating or deactivating the processes for suppression or inhibition, as specified by the human (e.g. via control switches (Bruemmer et al. [47]), (Kortenkamp et al. [67])) or the robot (e.g. via autonomous switching (Wasson & Gunderson [36]) or via dialogue (Fong et al. [17]) (see Section 2.4.2)).

Figure 2.7: A model of traded control using subsumption coordination approach (adapted from Graves & Czarnecki [89])

Suppression eliminates the control signal from the lower layer and substitutes it with the one proceeding from the higher layer. When the output of the higher layer is not active, the suppression node does not affect the lower layer signal. On the other hand, only inhibition eliminates the signal from the lower layer without substitution. Through these mechanisms, higher-level layers can subsume lower-levels.

Summary

This section illustrates the modelling of shared control and traded control. If traded control is to be applied in conjunction with shared control, or vice versa, different types of arbitration (depicted in Table 2.7) and command fusion (depicted in Table 2.8) coordination mechanisms must be integrated appropriately to provide varying degree of sharing and trading of control. This is further discussed in Chapter 5, where both arbitration and command fusion are employed in the implementation of a telerobotics system to provide varying degree of sharing and trading of control.

2.3.5 Summary

Given the different type of human control modes in Section 2.2, this section has discussed how robot autonomy is related to human control (Section 2.3.1). In addition, to facilitate the understanding of how human control inputs can be integrated into a robot control system, essential features that constitute the design of a robot control system are identified and discussed (Section 2.3.2). This includes a reviewed of different coordination mechanisms for coordinating human control inputs and robot autonomy (Section 2.3.3), and a discussion of how these mechanisms can be applied (Section 2.3.4).

In summary, to facilitate the achievement of seamless HRI as defined in Section 1.1.1, this section has provided a perspective of how robot might response to human control. However, how robot may response to human control in each of the five human-robot roles and relationships (summarised in Table 2.1) have not been discussed. Without a characterisation of how robot responses to human control in each of the five human-robot roles and relationships, it not possible to address how seamless HRI can be achieved when all the five interaction roles are integrated under the same interaction framework of an HRS, as envisaged in this thesis. Such a characterisation is still missing in the current literature of robotics and HRI. Hence, to facilitate, there is a need to have a framework to characterise how robot responses to human control in each of the five human-robot roles and relationships, which constitutes one of the main focuses of this thesis. This is further discussed in Chapter 3.

2.4 Human-Robot Communication

To ensure that the robot responds to the correct control mode when varying its degree of autonomy, issues pertaining to Human-Robot Communication (HRC) are important. In Human-Human communication, humans communicate with each other easily through the same language. They can communicate effectively through electronic communication devices or face-to-face. However, in the case of HRC, it is not that straight forward, because the human cannot communicate with the robot directly. A well-defined communication channel is required to address the different modes of interactions between the human and the robot. Some of the basic considerations in HRC are: methods of communication, communication format and the purpose of

communication as discussed in the following sections.

2.4.1 *Methods of Communication*

This relates to how information is transferred from the human to the robot (or vice versa). This issue is controversial because the current-state-of HRC encompasses a spectrum of methods, such as workstation, Personal Computer (PC) based control interfaces, Personal Digital Assistant (PDA) as interface devices (Ong et al. [32]) and haptic interface which enables “drive-by-feel” (Fong et al. [90]) capability. In addition, methods such as speech and gesture (vision), that is analogous to human form of communication, are also widely used (Fong et al. [90]). The use of these methods is problem-specific or application-specific. However, regardless of the method used, effective communication exchange between the human and robot is paramount.

2.4.2. *Communication Format*

This pertains to the communication language used for information trading between the human and the robot. Zhai and Milgram [49] proposed the notion of “continuous” and “discrete” languages as two different coding mechanisms to describe human-robot information trading. According to Zhai and Milgram, continuous language is used to represent information that is distributed continuously in quantitative or qualitative form, either along a spatial or a temporal dimension. In the context of robot communicating with human, examples include sending of raw sensors data, video images, etc. (i.e. perceived by the human). In the context of human communicating with robot, examples include sending of continuous signal (e.g. via input devices such as joystick) to control the robot. On the other hand, discrete language is used to represent information, which consists of separate or distinct elements. Examples of discrete language are signs, symbols, written text, etc. used for communicating with the robot. As compared to continuous language, discrete language is normally used when the available information bandwidth is low or the communication delay is high. However, this implies that the robot must have sufficient autonomy (see Section 2.3) to perform the task.

A good example of using discrete language for HRC is through the use of dialogue. The concept of using dialogue has recently received considerable research attention.

Emerging from the research of mixed initiative artificial intelligent systems (Myers & Morley [91]), it was subsequently adapted for HRC (Fong et al. [17], Green & Eklundh [51]). An example of dialogue adapted from Green & Eklundh [51] in defining a task during human intervention is as follows:

Human: Robot!
Robot: What is the task?
Human: Patrol Area A
Robot: Patrol Area A?
Human: Yes
Robot: Going to Area A

The idea of using dialogue is natural as it is very similar to human-human conversation. The purpose of confirming the human question (e.g. Patrol Area A?) is to ensure that the human has given the right command. If a wrong command is given, the human has an opportunity to correct his mistake. Using “confirmation” helps to prevent errors (i.e. giving wrong commands) and allows the robot to assist the human to learn from the mistake. Although this method is intuitive, it is difficult to decide how and when the robot should provide assistance or request for help. This issue is task specific and can only validate using human subject experiments.

2.4.3 Purposes of Communication

This pertains to what type of information is shared and traded between human and robot during communication and what is the purpose of the exchange of information (Klingspor et al. [50]). In performing a task, the human must provide the robot with accurate information about the task to be performed. On the other hand, the robot should communicate to the human any information regarding its state and provide a feedback of the current status of the task to allow the human to evaluate the robot task’s successes and faults. In addition, it is important for the robot to convey any difficulty it encounters during the task (therefore needs human’s assistance). A simple illustration of information sharing and trading between a human and a robot in a fetch-and-carry task is conveyed in Figure 2.8.

The types of information presented in Figure 2.8 are classified as follows (Scholtz [92]): *task information*, *environment information* and *robot state information*. In the

context of human communicating with the robot, task information is the knowledge of the task as specified and described by the human to be performed by the robot (Figure 2.8(a) & 2.8(e)). Task information is shared between the human and robot as follows: in Figure 2.8(a) & 2.8(e), the human performs a communicative act ‘r’ (e.g. via any one of the communication method introduced in Section 2.4.1), addressed to the robot. Through this, the following information is accessible to the human and the robot: ‘r’ means task specification (in this case the object to be handled, its location and destination), which are necessary for the task execution. By describing the task, the human provides the necessary instructions to the robot, about how to specify the task. Hence, the task information specified by ‘r’ is shared.

Figure 2.8: An illustration of information sharing and trading between a human and a robot in a fetch-and-carry task

In the context of the robot communicating with the human, task information is the knowledge of the robot with respect to the overall task defined by the human during task execution. This includes the robot's knowledge of its current location, its destination (Figure 2.8(b)) and its next task execution decision (Figure 2.8(d)). Environment information consists of information in the robot's working environment (Figure 2.8(c)). Examples of environment information are the objects (static or dynamic) in the environment and the robot's location relative to these objects. The robot state information is the information pertained to the robot's status (e.g. speed, sensors status,

health, etc.) and configurations (e.g. maximum sensing distance, available behaviours, etc.). In Figure 2.8(b) – 2.8(d), information is shared between the human and the robot via monitoring the execution of the tasks by the robot.

Figure 2.8(f) presents a scenario where information is shared and traded between the human and the robot. In this scenario, the robot takes the initiative to inform the human about its problem by performing a communicative act ‘n’, and the human responds to this communicative act by performing a communicative act ‘o’. Through this, the following information is exchanged between the robot and the human: ‘n’ means the robot status (low fuel) and ‘o’ means “advises” (recharge and task specification). Hence, the meanings of ‘r’ (from the robot) and ‘o’ (from the human) is shared and traded. In fact, the robot may engage the human in communication at multiple task execution points to resolve differences in an entirely dialogue manner (see Section 2.4.2).

2.4.4 Summary

In summary, from the perspectives of a human communicating with a robot, the human needs to be given a useful and understandable mechanism to make joint interactions with the robot environment references, so as to command, assist and give advice to the robot. To facilitate, the communication tools (Section 2.4.1) provided must: (1) be easy to use; (2) be adaptable to a variety of contexts; (3) afford negotiating tasks, goals and constraints directly. This implies that the control interfaces should be “multi-modal” so as to provide the human with a variety of control modes (e.g. for individual actuator, coordinated motion, etc.) and displays (e.g. text, visual, etc.). These are useful for applications, which demand context specific actions; i.e. for selecting control modes and displays based on situational requirements (Ong et al. [32]). Instead of overwhelming the human with huge amounts of unprocessed raw data, the displays should combine information from several sensors or data sources to present a single integrated view. Finally, the style of interaction with the robot should be consistent, e.g. in terms of human control input, form of robot’s output, and the communicative roles held by the human and the robot.

On the other hand, from the perspectives of a robot communicating with a human, the communication tools should provide mechanisms so that the robot conveys its states,

intents, actions and goals to the human. It should also facilitate the attraction of the human's attention when the robot requires assistance. The robot's actions should be observable, and the reasons for those actions should be clearly explicated in human-understandable form (e.g. via speech). As each action taken by the robot is embedded in a context, which renders that action understandable, the most salient aspects of the context must be represented. The intention is to interpret human actions and generate intelligible actions of its own during HRC. At a minimum, this context may be defined in terms of the robot current state, intents, actions and goals. For example, if the robot is assisting the human in a navigation task, the robot should state clearly what actions are being corrected, why it is corrected and how the actions are being corrected to the human. Finally, the robot must make clear its capabilities and limitations by providing appropriate feedback to the human so that the human may delegate tasks to the robot in a reasonable manner.

2.5 Chapter Summary

The aim of this chapter is to review existing work of sharing and trading so as to affirm that there is the need of a general framework of sharing and trading for getting a human and a robot to interact seamlessly in an HRS as identified and defined in Chapter 1. The review is based on the interaction strategies and requirements derived from the five HRI roles and relationships. They are *master-salve*, *supervisor-subordinate*, *partner-partner*, *teacher-learner* and *fully autonomous* mode by the robot in different types of HRS. To provide a roadmap, Figure 2.9 provides a summary of the work presented in this chapter by highlighting the essential points and their dependencies. The essential points from the review with regards of determining whether there is the need of developing a general framework of sharing and trading for seamless HRI is as follows.

First, the review has shown that the different types of interaction strategies derived from the five HRI roles and relationships are in the form of different human control modes (Section 2.2), ranging from manual control to autonomous control; where within this continuum, the common approach of achieving semi-autonomous control is to envisage how a human and a robot shared and/or traded control. Although, each control modes has different characteristics, their basic objective remains the same, i.e. to give human the flexibility to control a robot to perform an HRS task. At the same time, the

review of issues pertaining to control modes (Section 2.2.1) and control modes transitions (Section 2.2.2) showed that the designed of an effective control mode for giving human the flexibility to control a robot requires the robot to have the required autonomy to work with the human. This is further discussed below.

Figure 2.9: Summary of Chapter 2

Second, the review in Section 2.3 has shown that to facilitate vary degree of sharing and trading of control between a human and a robot, it not only requires giving human the flexibility to control a robot to perform a task, but also requires the adaptability of the robot autonomy to response to the human control. To achieve, human control intervention must be incorporated as an integral part of the robot's system. This shows another facet of sharing and trading, i.e. the robot needs to share and/or trade its *autonomy* so as to let the human share and/or trade control with the robot. This is important because it suggests that the form of sharing and/or trading can also be in the context of robot autonomy a part of human control. However, most of the work in the literature normally discusses these two elements together in the context of a control mode, even though they are already acknowledged as two separate elements. However, to address how a human and a robot shared and/or traded in a holistic manner, these two elements must be differentiated. Hence, to facilitate, there is a need to have a framework to describe how “human control” and “robot autonomy” are involved in the process of

sharing and trading. Particularly to characterise how a human and a robot share and trade in each human-robot roles and relationships (established in Section 2.1.1) so as to let them work as a team, which is missing in the literature of robotics and HRI.

Third, the review in Section 2.4 has shown that to facilitate sharing and trading, both a human and a robot must convey their states, intents, actions and goals to each other. This shows another facet of sharing and trading, i.e. sharing and trading of *information*. In accordance to the review, information sharing between a human and a robot is a phenomenon during HRI. On the other hand, information trading is normally used as a means for coordinating their actions. Although this concern is closely related to how a human and a robot shared and/or traded, but it is not conveyed in the existing work of sharing and trading. Hence, to address, there is also a need to have a framework to describe how information sharing and trading is related to human control and robot autonomy.

Based on the review, it is reasonable to conclude that there is a lack of an agenda to synthesise the key elements (summarised in Table 2.9) of the process of sharing and trading between a human and a robot in a holistic manner, let alone a unified general framework of sharing and trading. This affirms the need of a general framework of sharing and trading to assist in the design and development of an HRS for seamless HRI proposed in Chapter 1.

Table 2.9: A summary of human control, robot autonomy and information

Element	Attribute	Feature	Covered In
Human Control	Mode	manual-control, semi-autonomous control and autonomous control	Section 2.2.1
		complement and redundancy	Section 2.2.2
		Characteristics: engagement conditions, disengagement conditions, operation modifications and control properties	
		Strategy: abstract description ... detail description	
	Transitions	Human initiated, robot initiated, mixed initiated	Section 2.2.3
	Monitoring	observing ... inspecting	
	Intervention	Purpose: commands ... guidance ... repair ... stop	
		Level: skills-based, rules-based and knowledge-based	
		Frequency (situation-dependent): low ... high	

Robot Autonomy	Type	operating autonomy and decisional autonomy	Section 2.3
	Variability	fixed autonomy ... adjustable autonomy	
	Degree – <i>via varying nested ranges of action</i>	possible actions, independently achievable actions, achievable actions, permitted actions and obligated actions	Section 2.3.1 (Table 2.3)
	Level	human maximum autonomy ... robot maximum autonomy (e.g. Sheridan's ten-level formulation of autonomy)	Section 2.3.1 (Table 2.4)
Information	Source	continuous ... discrete	Section 2.4.2
		quantitative ... qualitative	
		Spatial ... temporal	
	Type	Task information, environment information and robot state information	Section 2.4.3

This chapter has shown the importance/role of sharing and trading in the design and development of an HRS for providing effective HRI, what forms they take in an HRS and how they can be implemented to enable HRI. The following chapter will address the concept of sharing and trading, the main focus of this thesis.

Chapter 3

A Framework of Sharing and Trading

In Chapter 2, key elements namely *human control*, *robot autonomy* and *information* involved in the process of sharing and trading between a human and a robot are identified. The aim of this chapter is to use these elements to develop a framework of sharing and trading for studying how a human and a robot can interact seamlessly in an HRS. Based on the discussion in Chapter 2 concerning the unique advantages of the symbiosis of the human and the robot in developing a cooperative HRS, the main issue is to determine which aspects of their capabilities should be exploited and combined. This issue is widely known as the problem of *task allocation* [16, 18, 21]. It is a central component of system engineering and its aim is to provide a rational means of determining which system-level task should be performed by the human and which by the robot in accordance to their capabilities and limitations. Hence, to develop a conceptual framework of sharing and trading for the design and development of an HRS, the issues pertain to task allocation between the human and the robot must be considered.

Although task allocation is important, it does not gain much attention in the domain of HRI. To date, research effort in HRI mostly concentrates on the development of HRS architectures and the incorporation of human-robot interfaces as means for human to control the robot, as discussed in Chapter 2 (e.g. [16, 18, 19, 43]). The consideration of task allocation in HRI is normally done in an informal manner. HRI designers normally make allocation decisions implicitly based on the unique advantages possess by both the human and the robot. For example, prior knowledge of a task, “common sense” in reasoning and perception are attributes that are possessed by humans but not by robots. On the other hand, rapid computation, mechanical power, diverse sensory modalities and the ability to work in hazardous environment are great advantages of robots that humans do not possess.

Although the informal allocation of task can be used to make allocation decisions reasonably well, it may not be able to provide a judicious provisional allocation decisions; i.e., looking into: *when* problem arises during task execution and *how* might human and robot cooperate to resolve the problem. Successful resolution of task

allocations often requires not only an understanding of fundamental issues concerning the capabilities and limitations of humans and robots, but also of a number of subtle considerations when both human and robot interact in performing an assigned task. This view is based upon the literature from human factors engineering for Human-Machine Interaction (HMI) and Human-Computer Interaction (HCI) in automated system [5, 93]; such as flying an airplane [23, 26], supervising a flexible manufacturing system [25, 94] or monitoring a nuclear power plant [2, 6].

The intention is to establish a basis for describing the task interaction between the human and the robot in an HRS. The aim is to address the role of sharing and trading in task allocation, which is lacking in the literature of HRI. Hence, Section 3.1 first provides an overview on the concept of task allocation. Here, a *task* can mean the complete performance of a given procedure, or the totality of effort to perform a given thing, to control a given process, or to diagnose or solve a given problem (Sheridan [93]). Alternatively a task can mean a small sub-element such as a particular movement or measurement. In this thesis, the types of tasks concerned are those involving reasoning, sensing, and acting (i.e. manipulation and mobility) which use the human and the robot capabilities of performing these functions. The concern here is to look into how task can best be shared and traded by both human and robot acting as a team (i.e. *task sharing and trading between the human and the robot*), as compared to assignment of a whole task to either the human or the robot. As the literature of HRI does not provide any insight for studying task allocation, the concept of task allocation discussed is based upon the literature from human factors engineering for HMI and HCI in automated system. The discussion views HRI as an evolution of human interacting with technological systems, from machine/computer in automated system to current-state-of HRS (Table 2.1) brought about by advancement in technology, and theory and practice of robotics. Hence, the discussion includes human interacting with “*machine*” and “*computer*” when discussing work from the perspective of HMI/HCI in automated system.

Based on the concept of task sharing and trading presented in Section 3.1, to address the process of sharing and trading between a human and a robot in a structured basis, basic activities within an HRS is presented in Section 3.2. This is followed by a detail exposition of each basic task activity presented in Section 3.2.1 to 3.2.5 respectively.

Consequently, a discussion regarding the framework formulation is presented in Section 3.3. Finally, in Section 3.4, the summary of this chapter is presented.

3.1. Task Allocation

It is proposed that after a task definition of a particular application is established, a systematic approach to the design of an HRS can be based upon task allocation. Here, *task definition* is defined as the task model of an application goals and requirements which includes the analysis of the underlying technological and environmental constraints (Hancock [5]); and *task allocation* is “*the assignment of various tasks either to humans or machines that are capable of doing those tasks*” (Sheridan [93]). This perspective is based upon Fitts [95] and is regarded by many as an essential component in systems engineering process [93]. In this approach, the attempt is to identify which comparable capabilities are humans and machines “better at”, and subsequently “match” their best capabilities with aspects of the overall task at hand. This has come to be known as the “Fitts’ Men-are-better-at - Machines-are-better-at (MABA-MABA) List”, presented in Figure 3.1. This list is often referred to as the first well-known basis for task allocation in the human factors literature [5, 93].

Generally, the MABA-MABA approach uses the criteria embodied in a list (e.g. the Fitts’ List as presented Figure 3.1) as the sole basis for making “*who does what*” mandatory allocation decisions. If the advantages in performing a particular task rested with either human or machine, then that task was allocated accordingly. However, if the total sum of such allocation decisions resulted in imbalances in tasks assignment (e.g. human overloads with tasks or shortages in machinery), tasks would be transferred between human and machine until a viable solution was reached (Sheridan [93]). Although this approach has gone through a sequence of different instantiations, e.g., published by Bekey [96] and Meister [97], the fundamental principle of allocation according to Fitts’ List does not vary (Hancock [5], Sheridan [93]). That is, the input for this approach is typically a list of abstract tasks the HRS needs to achieve and the output is typically the same list categorised in terms of whether human or machine should perform the task (Hancock [5], Sheridan [93]). The use of the MABA-MABA approach (or other variants) that lends itself to the rigid a priori assignment of tasks to human and machine constitutes the *static task allocation* perspective, i.e., once a task is allocated to

the human or the machine, either of them is responsible for the task at all times.

Figure 3.1: The Fitts' List (adapted from Fitts [95])

Although the MABA-MABA approach provides a formal and rational way for making allocation decisions, it has been criticised by many researchers [5, 93, 98, 99]. The main concern is that there are large numbers of possible interactions between human and machine for consideration, not simply just “human *versus* machine” based on rigid *a priori* pre-determined allocation based on human and machine capabilities and

limitations characteristics [5, 93, 98, 99]. Probably, the first to point out this drawback is Jordan [98]. He suggested that allocations of tasks between human and machine would only become useful if human and machine were looked at as *complementary*, rather than *comparable* as in the MABA-MABA approach. He argued that the idea of *what human does best* and *what machine does best* discussed above remains useful. This is because the inherent capabilities and limitations of human and machine are complementary [5, 93, 99]. However, the idea of comparing the human with the machine should be discarded. Instead, the consideration should be how they might cooperate based on their complementary capabilities and limitations. Specifically, to achieve human-machine cooperation, human and machine should be seen as the unit of concern rather than dichotomising them into separate units [5, 93, 99] as in the MABA-MABA approach. Sheridan [100] shared the same view and stated that: “*to cast the task allocation problem in terms of humans versus computers is simplistic, unproductive and self-defeating. We should be concerned with how they can cooperate*”. Bradshaw et al. [99] purport that the point is not to think so much about which tasks are best performed by humans and robots but rather how tasks can best be shared and traded by both humans and robots working together. This is further discussed in the next section.

3.1.1 Task Sharing and Trading

Although it is reasonable to allocate tasks to both human and robot based on their inherent capabilities and limitations (i.e. based on the “who does what” mandatory allocation decisions), such allocation decisions may not hold at all times. To illustrate, consider the following situations. In performing an HRS task, a human may get tired or bored after long hours of operations, or a robot may fail to perform an allocated task due to a lack of prior task knowledge or sensors malfunction. If any of these situations happen and the HRS is designed solely based on mandatory allocated decisions that do not anticipate any interaction strategies that allow human to exchange control with the robot, the overall HRS performance may degrade or the HRS may breakdown physically. This implies that such decisions are not efficient and efficacious under certain situations. A successful task allocation scheme must also include considerations of *timeliness* and *pragmatism* of the situation for making provisional task allocation decisions (e.g. when a robot fails to perform its allocated task during operation, how does human assists the

robot; or when the human has problem performing a task, how does the robot provides appropriate assistance to the human).

A Paradigm of Robot Assists Human – Human Assists Robot (RAH-HAR)

An approach useful for making timeliness and pragmatic decisions is the concept of sharing and trading invoked by Sheridan [2] discussed in Section 1.1.2. Here, Sheridan concept of sharing and trading is adopted and extended to provide a finer grain of sharing and trading of task between a human and a robot. This is depicted in Figure 3.2. In contrast to Sheridan's concept that only emphasised on how technological system (i.e. computer) might aid the human, the concept of sharing and trading envisaged here looks into the possibilities of letting both the human and the robot to assist each other. This perspective is important because virtually all applications of HRS (Table 2.1) require human to provide appropriate assistance to the robot (e.g. object recognition, decision-making, etc.), as the goal of building robots with fully autonomous capability has not yet been met in robotics research (Murphy & Rogers [16], Arkin [45]).

Figure 3.2: Concept of task sharing and trading between a human and a robot. H the human, R the robot and T is the task (adopted and extended from Sheridan [2], Figure 1.3)

As shown in Figure 3.2, both human and robot might engage in different interaction “roles and relationships”, namely *master-slave*, *supervisor-subordinate*, *partner-partner*, *teacher-learner* and *fully autonomous* (as established in Section 2.1.1, summarised in Table 2.2) based on different situations for them to share and trade task; ranging from “*no assistance provided to the human by the robot*” (i.e. human replaces robot) to “*no assistance provided to the robot by the human*” (i.e. robot replaces human) during task execution. An overview of the different types of cooperation strategies between human and robot based on how the human and the robot might assist each other in Figure 3.2 is presented in Table 3.1.

Table 3.1: Different types of cooperation strategies between a human and a robot based on how the human and the robot might assist each other as shown in Figure 3.2.

Human-Robot Cooperation Strategies	Characteristics
No assistance provided to human by robot.	This strategy is useful when human wants to perform a task by him/herself manually.
Robot assists human by extending his/her capability.	This strategy is useful to let the robot extends the human capability so that he/she can perform a task that is beyond his/her ability.
Robot assists human by dealing with different aspects of a task.	This strategy is useful to let the human and the robot cooperate to deal with mutually complementary parts of a task.
Robot assists human by providing appropriate support to the human.	This strategy is useful to let the robot provide active (i.e. constant or continuous) assistance to the human so as to reduce his/her burden or task demands.
Robot assists human by taking over the task from the human.	This strategy is useful to let the robot take over a task from the human when the human fails to perform a task or it can be the human who want the robot to perform the task by itself when he/she find that the robot has the ability to perform the task.
Human assists robot by providing appropriate support to the robot.	This strategy is useful to let human provide the require assistance to the robot when the human perceived that the task performance of the robot is not satisfactory or the robot request for human assistance.
Human assists robot by taking over the task from the robot.	This strategy is useful to let the human take over a task from the robot when the robot fails to perform the task.
No assistance provided to robot by human.	This strategy is useful to let the robot perform a task by itself with minimal or no human intervention.

Guidelines for RAH-HAR

A good guide to envisage RAH-HAR discussed above is by Woods [101]. He proposed a new perspective for looking into how the advantages of the human and the robot can be exploited through appropriate forms of cooperation between them, called “Un-Fitts List”. This is presented in Table 3.2 as summarised by Hoffman et al. [102].

Table 3.2: Woods' Un-Fitts List (adapted from Hoffman et al. [102])

Robot	
Are constrained in that:	Need Human to:
i. Sensitivity to context is low and is ontology-limited	i. Keep them aligned to context
ii. Sensitivity to change is low and recognition of anomaly is ontology-limited	ii. Keep them stable given the variability and change inherent in the world
iii. Adaptability to change is low and is ontology-limited	iii. Repair their ontologies
iv. They are not “aware” of the fact that the model of the world is itself in the world	iv. Keep the model aligned with the world
Human	
Are not limited in that:	Yet they create robots to:
v. Sensitivity to context is high and is knowledge-and attention-driven	v. Help them stay informed of ongoing events
vi. Sensitivity to change is high and is driven by the recognition of anomaly	vi. Help them align and repair their perceptions because they rely on mediated stimuli
vii. Adaptability to change is high and is goal driven	vii. Effect positive change following situation change
viii. They are aware of the fact that the model of the world is itself in the world	viii. Computationally instantiate their models of the world

From the “Un-Fitts List”, the important themes are the issue of sensitivity to limited ontologies and changing context, and how human can alleviate these deficiencies in robots; and also the emphasis on how robot can extend the capabilities of human and relieves the human task through appropriate task sharing (Bradshaw et al. [99]). In other words, it looks into the possibilities of letting “*human assists robot*” (depicted in Table 3.2: i. to iv.) and “*robot assists human*” (depicted in Table 3.2: v. to viii.) in an HRS in order to achieve human-robot cooperation. However, to provide the required assistance to each other, the human and the robot must be aware of and recognised what task each other is doing, why each other is doing it, etc. Examples of the kinds of questions that the human and the robot need to be able to know so as to assist each other during task execution are presented in Table 3.3.

Table 3.3: Examples of the kinds of questions that the human and the robot need to be able to know so as to assist each other (adapted and modified from Bradshaw et al. [99])

Questions about the shared representation of the problem state of a task	Questions about the representation of the state of human and robot in perform a task
What type of problem is it?	How did the human/robot get into this state?
Is the problem routine or difficult?	What is the human/robot doing now?
Is the problem high or low priority?	Why is the human/robot doing it?

What types of cooperation strategies are appropriate? (e.g. those strategies depicted in Table 3.1)	Is the human/robot having difficulties? Why?
What dependencies must be considered?	What is the human/robot doing to cope with difficulties? Is the human/robot likely to fail?
How is the problem state evolving?	What will the human/robot do next?

Summary

In short, the concept of task sharing and trading envisaged here is based on how RAH-HAR. Within this paradigm, a human and a robot *cooperate* to perform tasks and to achieve common goals via assisting each other. Instead of using only one fixed role and relationship strategy, both the human and the robot can engage in different roles and relationships (as summarised in Table 2.2) to compensate for the unique kind of failures and limitations possessed by each other. Through this, instead of serving the human as a mere device, the robot can operate more like a *teammate* of the human. With this paradigm, the human and the robot have more freedom during task execution and are more likely to find good solutions when they encounter problems, as compared to using fixed a priori task allocation strategies that does not facilitate HRI.

3.1.2 The Role of Task Sharing and Trading in Task Reallocation

Section 3.1.1 has discussed the role of task sharing and trading (i.e. based on how RAH-HAR) in addressing the issues of fixing a priori task allocation during task execution. As one major characteristic of this process is non-static, it is essential to address the role of task sharing and trading in *task reallocation* during task execution, i.e., *the reallocation of a particular task dynamically between the human and the robot in accordance to the overall system task performance*. This view is based upon the research from adaptive task allocation for the dynamic assignment of tasks between human and computer in an automated system [5, 23, 103]. The *overall system task performance* is defined as the achievement of the overall goal of a system task in accordance to both human and computer performances. To facilitate further discussion, the following sub-section first provides an overview of adaptive task allocation that is useful in addressing the role of task sharing and trading in task reallocation. It is useful in that it provides general heuristics in understanding how human and technological system might interact during task execution.

Adaptive Task Allocation

Suppose in an automated system, a human and a computer are requested to perform assigned tasks for some period of time. As time passes by, the operating environment may change, or performance of the human may degrade gradually as a result of psychological or physiological reasons. If the overall performance of the system is to be achieved, it may be wise to reallocate tasks between the human and the computer because the situation has deviated from the original one. An approach that modifies task allocation dynamically depending on situations is called adaptive task allocation⁵ (also called “Adaptive Automation” [23] or “Adaptive Aiding” [103]). The concern of adaptive task allocation lies in the implementation of “*computer-aided adaptive allocation strategies*”, in which both static and dynamic representations of human and automated system capacity are presented for continual evaluation and manipulation [5]. Achievement of such allocation strategies promises benefits from the best abilities of both human and computer while providing neither with incompatible task demands or excessive task loading. Examples of the types of questions and allocation responses that might be addressed in adaptive task allocation are shown in Table 3.4.

Table 3.4: Examples of the types of questions and allocation responses to be addressed in adaptive task allocation (adapted from Hancock et al. [104])

Questions	Allocation Responses
Who	IF: A human performs within predetermined criteria THEN: The human continue to perform the task, otherwise the task is allocated to the computer, if it is capable of performing the task.
What	IF: Only parts of tasks are being performed poorly THEN: Only these parts shall become available for dynamic allocation.
When	IF: Certain time periods are associated with increased demand, error, or loss of situation awareness THEN: These periods will be appropriate for dynamic allocation.
Where	IF: Particular environments or combinations of environmental variables are associated with increased task demand or error THEN: Encountering these environments triggers dynamic allocation.
Why	IF: Extended periods of allocation have detrimental effects (objective or subjective) THEN: Allocation shall periodically return control to the human.
How	IF: Human performance, environmental attributes, and psycho-physiological indexes are important for human-computer interaction THEN: All of these are inputs for allocation shift.

Research in adaptive task allocation described above provides a basis for understanding how to perform task reallocation dynamically between human and

computer throughout task execution. However, it may not be feasible for HRI in an HRS. This is further discussed in the following sub-section.

Task Reallocation in HRS via Task Sharing and Trading

In the context of HRI in an HRS, when reallocating tasks adaptively between human and robot, it is vital to know that dynamic HRI role adjustment comes at a cost. This is because it may interrupt the ongoing dynamic task process of human and/or robot. A major challenge is to ensure that the task sharing and trading between human and robot is continuous and transparent so as to achieve seamless HRI during task execution. As the task sharing and trading between human and robot in an HRS may not be predictable and may occur in an arbitrary manner depending on the ongoing task performance and situation¹⁰, it is not feasible to employ pre-programmed decision rules (as in adaptive task allocation for automated system, depicted in Table 3.4) to trigger task reallocations dynamically based on pre-defined conditions. For successful accomplishment of a particular task during task execution in an HRS, both human and robot should cooperate through varying degree of human control (see Section 2.2), robot autonomy (see Section 2.3) and appropriate human-robot communication (see Section 2.4) when problem arises. This implies that the role of task sharing and trading in task reallocation not only require to reallocate task responsibilities among human and robot (i.e. via role changing) but also to coordinate the interaction process between them. Examples include, resolve their conflicts, actions and intentions, arbitrate human/robot request for assistance, etc.

In short, the decision to perform a task reallocation via task sharing and trading discussed above is invoked either by a human or a robot during task execution. By specifying task reallocation in this manner, the original definition of task reallocation based solely on the overall system task performance use in automated system may not be suitable. Here, *task reallocation* is defined as the reallocation of a current desired input task that is allocated to the human and the robot with a completely new task specification. The conditions for task reallocation can be based on the ongoing task performance of the human and the robot, changes in task environmental or simply changes in the task plan that causes the current desired input task to be discarded.

¹⁰ This view is based upon the review of literature of different robotics applications discussed in Chapter 2.

3.1.3 Summary

This section argues that making prior task allocation decision solely based on classifying the human capabilities and limitations with respect to the robot capabilities and limitations (i.e. “*who does what*”) is not sufficient in addressing problems that arise when the human and the robot work together during task execution. This is because the boundaries that define the task interactions between the human and the robot are “fluid” and “dynamic” in nature. This implies that to be flexible, the a priori task allocation decision during design stage must also consider timeliness and pragmatic allocation decisions for resolving arising conflicts/problems between the human and the robot. To facilitate, the concept of *task sharing and trading* is introduced (Section 3.1.1). This concept is an extension of Sheridan [2] concept of sharing and trading (discussed in Section 1.1). The concept proposed here not only considers how a robot might assist human but also the consideration of how the human might assist the robot. Through this, a spectrum of task interaction modes ranging from “no assistance provided to the human by the robot” to “no assistance provided to the robot by the human” (Figure 3.2) can be envisaged to address contingencies that emerge when the human and the robot work together during task execution. In other words, this concept serves as a guide to assist in making prior timeliness and pragmatic task allocation decisions by providing insight into the design of different cooperation strategies based on how the human and the robot might assist each other during task execution (Table 3.1). Table 3.5 provides an abstract description of the prior task allocation in an HRS based on the considerations of capability of the performer but also on the timeliness and pragmatism of the situation.

Table 3.5: A flexible prior task allocation based on “who does what” mandatory allocation and “when and how” provisional allocation decisions

Task Allocation	Determine By
1. Tasks that are best performed by the human.	“Who does what”
2. Tasks that require human-robot cooperation but may require the robot to assist the human.	Timeliness and pragmatism of the situation
3. Tasks that require human-robot cooperation but may require the human to assist the robot.	Timeliness and pragmatism of the situation
4. Tasks that are best performed by the robot.	“Who does what”

In summary, the sharing and trading between a human and a robot in an HRS discussed here is in the context of a task. By *task* implies the required human’s and

robot's functions and the goals they are attempting to accomplish. This means that the "things" that the human and the robot can share and trade is placed within the context of a task. As suggested in Section 1.1.4 and affirmed in Chapter 2, the "things" that a human and a robot shared and traded is in the context of *human control*, *robot autonomy* and *information*, which constitute the key elements involved in the process of sharing and trading between them. However, to consider how these elements constitute the task sharing and trading between the human and the robot, there is a need to look into the basic activities within an HRS. The aim is to describe how seamless HRI can be achieved via task sharing and trading. This is further discussed in Section 3.2.

3.2 Defining Sharing and Trading in a Human-Robot System

To address the concept of task sharing and trading presented in Section 3.1.1 in a holistic manner, basic task activities within an HRS is presented in this section to define the sharing and trading between a human and a robot. Here, a *task activity* is defined as a specification of a part of task to be accomplished within an HRS. Each single task activity that is distinguished within a human or a robot task process may be divided into a number of *functions*. Functions are used to identify small parts of task activities in a way that is necessary in achieving a particular task goal. To illustrate, consider the fetch-and-carry task depicted in Figure 2.8 (Section 2.4.3). In this context, the "task activities" of this task includes: human commanding the robot to fetch a red box from room A to room B; execution of the task by the robot; feedback the necessary information (i.e. task, environment and the robot state information) to the human for monitoring; and human control intervention so as to provide assistance to the robot when its encounter any problems. To perform these task activities, the robot must have the required "functions" such as mobility and manipulability (i.e. operating autonomy, Section 2.3) to execute the task "physically", and also perception and reasoning abilities (i.e. decisional autonomy, Section 2.3) to act safely and independently in its task environment. Following the discussion in Section 3.1 regarding the task allocation between the human and the robot, task sharing and trading between them, and task reallocation, the basic task activities within an HRS may consist of:

- Desired task as input task, T_I
- Task allocated to the human, T_H

- ❑ Task allocated to the robot, T_R
- ❑ Task sharing and trading between the human and the robot, $T_{S\&T}$
- ❑ Task reallocation, T_{RE}

These basic activities may be related as shown in Figure 3.3.

Figure 3.3: Activities within a Human-Robot System

In Figure 3.3, it is suggested that there are three main paths to describe the activities within an HRS. The first path shown in Figure 3.3 defines the input task (T_I) allocated to the human (T_H) and/or the robot (T_R) (as discussed in Section 3.1 and 3.1.1). The second path defines task sharing and trading ($T_{S\&T}$) between the human and the robot (as discussed in Section 3.1.1). The third path represents task reallocation (T_{RE}) of the T_H and/or the T_R with a completely new T_I specification (as discussed in Section 3.1.2). Each of these five activities (i.e. T_I , T_H , T_R , $T_{S\&T}$ and T_{RE}) in Figure 3.3 is further discussed in Section 3.2.1 to 3.2.5 respectively.

3.2.1 Input Task (T_I)

Given the task definition of particular application tasks, such as large area surveillance, reconnaissance, objects transportation, objects manipulation, exploration of unknown environment, hazardous waste cleanup, to name a few, the next stage is to determine whether human, robot, or some combination of both should perform the T_I (i.e. prior task allocation) as follows. First, identify which tasks can *only* be allocated to either human (T_H , Section 3.2.2) or robot (T_R , Section 3.2.3) based on “who does what” mandatory allocation decisions. Subsequently, provisionally allocate tasks based on

timeliness and pragmatic decisions, so as to take advantage of the symbiosis of the human and the robot capabilities to achieve task goals during task execution, while achieving the concept of task sharing and trading ($T_{S\&T}$) discussed in Section 3.1.1.

Generally, the considerations of making provisional task allocation based on the human and the robot capabilities during task execution can be characterised along several dimensions as follows. This characterisation is based on the review of literature of different robotics applications discussed in Chapter 2.

- *Reasoning*: This includes attributes such as decision-making, task planning, situation understanding, and error detection and correction, to name a few. Consider an example of a robot performing a navigation task of traversing from one location to another location get “trap” in the task environment causing the robot not able to reach the specified location. To let the robot perform this navigation task successfully requires human assistance such as decision-making and situation understanding to guide the robot out of this trouble scenario.
- *Perception*: This includes attributes such as multi-modalities sensing, object recognition/discrimination/classification, to name a few. Consider a situation in which a robot attempting to move into a room and encounters door curtains directly in its path. Depending on the robot sensors suite, the robot’s perception system may have difficulty determining if the curtains are obstacles or whether its path is block. Thus, the robot may not be able to traverse into the room. However, if human perceives this situation, he/she can assist the robot by overriding the robot’s perception system and command the robot to drive through the door.
- *Mobility*: This includes attributes such as traverse distance, mission duration, repetitive/unique mission, consequence of failure, moving with minimal disturbance to environment, and complexity of working environment (e.g. distribution of targets/obstacles, accessibility (e.g. small spaces), slope variability, soil/surface consistency, degree of uncertainty, etc.), to name a few. For example, these attributes are important considerations when a human is delegating a navigation task to a robot or providing appropriate assistance to the robot when its encounter problems (as discuss above).

- *Manipulability*: This includes attributes such as object shapes (standard/unique), repetitive/unique motion, precision/dexterity motion, consequence of failure, moving with minimal disturbance, and complexity of motion, to name a few. For instance, these attributes are important considerations when a human is delegating a manipulation task to a robot or providing appropriate assistance to the robot when its encounter problems while performing the task.

The discussion above has provided an abstract view of T_I and attributes for making provisional task allocation decisions during task execution. This is essential because it provides a basis for describing T_H and T_R in Section 3.2.2 and 3.2.3 respectively.

3.2.2 Task of Human (T_H)

The primary T_H in an HRS is to *control* a robot to perform particular application tasks. In general, this encompasses the following functions that the human might require to perform as identified in Section 2.2:

- *Decision-making* - to decide whether the robot has the ability to perform the desired task. The considerations for making this allocation decision can be based on task attributes discussed in Section 3.2.1. For example, to control a robot to perform a navigation/manipulation task, the human must first determine whether the robot has the “physical” functions or *operating autonomy* (Section 2.3), such as mobility/manipulability (Section 3.2.1) to execute the desired task. Next, if the robot has the required operating autonomy to perform the task, the human must decide whether the robot has the required *decisional autonomy* (Section 2.3) to carry out the task by itself. If the human decides that the robot has the ability to carry out the task, then the human will proceed to task planning (discuss below). However, if the human find out that the robot does not have the required knowledge, he/she may teaches the robot to perform the task (see below).
- *Task planning* - to schedule the task process and how it is carried out. For instance, setting goals, which the robot can comprehend.
- *Teaching* - to transfer task knowledge to the robot if the robot does not have the prior knowledge to perform the desired task.
- *Monitoring* - to ensure proper the robot task execution and performance.

- *Intervention* - to provide appropriate assistance to the robot if any problems arise during task execution. Problems can include hardware failures, software failures, and human manual configuration requests for unscheduled support, to name a few.

The above are the conceivable tasks that can only be allocated to human based on the human roles in an HRS, e.g. as supervisor, partner or teacher of the robot as discussed in Section 2.1.1 (summarised in Table 2.2). The human roles in an HRS in turn determine how a robot might perform the HRS task. This is discussed below.

3.2.3 Task of Robot (T_R)

The primary T_R in an HRS is to response to human control and in turn adapts its *autonomy* to perform the application tasks. This encompasses two basic functions that the robot requires to perform as identified in Section 2.3 and 2.4:

- *Physical task execution*: In general, how a robot might execute an HRS task depends on how human control the robot; i.e. based on the human-robot roles and relationships in an HRS as established in Section 2.1.1 (Table 2.2). For example, if the human adopts the master-slave paradigm to control the robot, then the robot will just mimics the human control actions exactly in performing the HRS task. On the other hand, if the human adopts the supervisor-subordinate paradigm to control the robot, then the robot will perform the HRS task planned by the human with minimum human intervention.
- *Feedback information*: To facilitate human monitoring and intervention of the robot task execution, the robot must feedback information to the human. This includes task information, environment information and the robot state information as discussed in Section 2.4.3.

Section 3.2.2 and this section have provided an overview of T_H and T_R . This is essential because it provides a basis for describing the $T_{S\&T}$ between the human and the robot in the following section.

3.2.4 Task Sharing and Trading ($T_{S\&T}$) between Human and Robot

The concept of $T_{S\&T}$ discussed in Section 3.1.1 is based on how RAH-HAR (Figure

3.2). Within this paradigm, both the human and the robot may work as a team by engaging in different roles and relationships (Section 2.1.1, Table 2.2) so as to exploit each other capabilities and/or compensate for the unique kinds of limitations (e.g. the Un-Fitts List, Table 3.2) of each other during task execution. Although the concept of $T_{S\&T}$ invoked in Section 3.1.1 is able to describe the different types of cooperation strategies between a human and a robot based on how they might assist each other (as depicted in Table 3.1), it does not provide much insight into the achievement of seamless HRI given the interaction roles they might adopt during task execution. To facilitate this, it is important to consider the dynamics of the $T_{S\&T}$ process so as to address the contingencies that arise when the human and the robot work together during task execution. To address, there is a need to characterise the underlying basic elements that constitute the $T_{S\&T}$ between the human and the robot.

Basic Elements of $T_{S\&T}$

In accordance to the key elements namely *human control*, *robot autonomy* and *information* involved in the process of sharing and trading between a human and a robot identified in Chapter 2; it is defined here that for a human to perform $T_{S\&T}$ with a robot, he/she must select the right control mode (see Section 2.2.2) to share and trade control with the robot. On the other hand, for a robot to perform $T_{S\&T}$ with human, the robot must adapt to the right degree of autonomy (see Section 2.3.1) so as to respond to the selected control mode (i.e. sharing and trading its autonomy with the human). This implies that “human control” and “robot autonomy” are placed within the context of a task collaboration for the human and the robot to accomplish their respective goals. By *task collaboration* means that both T_H and T_R are performed via appropriate human control (see Section 3.2.2), and varying degree/level of robot autonomy (see Section 3.2.3) respectively. Thus, both “human control” and “robot autonomy” are the basic elements that a human and a robot can share and trade with each other respectively to achieve $T_{S\&T}$. In both cases, to perform the appropriate actions (i.e. changes in human control and robot autonomy), it invariably involves sharing of information (see Section 2.4.3). If the human and the robot have different perceptions regarding the shared information, they must trade information to clarify any doubt before actual actions can be performed (see Section 2.4.3). In short, information sharing and trading is to find out

what the other party is doing, what the intention of the other party might be and to resolve any conflict if it arises during task execution. Hence, $T_{S\&T}$ is classified into *human control*, *robot autonomy* and *information sharing and trading* respectively to depict what can be shared and traded between a human and a robot during task execution. A summary of the above basic elements (i.e. human control, robot autonomy and information) can be found in Table 2.9.

The basic elements discussed above are important because they provide the basic constructs towards the characterisation of $T_{S\&T}$ in different HRI roles and relationships established in Section 2.1.1 (Table 2.2). The intention is for describing how seamless HRI can be achieved based on the concept of $T_{S\&T}$. This is discussed below.

Characterisation of $T_{S\&T}$ in Different HRI Roles and Relationships

The main corollary of the concept of Human-Robot Team (HRT) discussed in Section 2.1.1 is it requires the flexibility in HRI roles transition in order to let both human and robot work as a team. As discussed in Section 2.1.1, both human and robot are related in several possible roles. They are *master-slave*, *supervisor-subordinate*, *partner-partner*, *teacher-learner* and *fully autonomous* mode by the robot. Given these HRI roles, the concern here is: how are these roles related to the process of $T_{S\&T}$ between human and robot. Here, it is posited that different kinds of HRI roles and relationships will inherently induce different phenomenon of $T_{S\&T}$, ranging from pure task decomposition to more complex task or sub task interactions. This is depicted in Figure 3.4, in accordance to the basic elements (i.e. human control, robot autonomy and information) of $T_{S\&T}$.

$T_H \leftrightarrow T_{S\&T} \leftrightarrow T_R$	<p>Task sharing and trading ($T_{S\&T}$) between human and robot, extracted from Figure 3.3. It is further break down into the basic elements (i.e. human control, robot autonomy and information) to describe the five roles and relationships below.</p>
<p><u>Master-Slave</u></p> $T_H \leftarrow I_S \leftarrow T_R$	<p>Trading is not involved. Sharing is in the form of information sharing (I_S) to extend human's sensing to a remote location. For example (Roman [34]), a submersible robot operated via teleoperation extends the human's sensing (e.g. through video images relay from the robot on-board cameras) to a remote location so that he/she can work safely in the hazardous environment.</p>
<p><u>Supervisor-Subordinate</u></p>	<p>Control trading (C_T) from human to robot is to let human delegate control to robot (see Section 2.2.1). On the other hand, the robot needs to adjust its autonomy (i.e. autonomy trading from the robot to the human, A_T) to respond to the human control. Information sharing (I_S) from the robot to the human is for the human to monitor the robot task execution. However, if problem arises, both the human and the robot need to trade information (I_T) to clarify any doubts.</p> <p>An example to illustrate this is the Mars Rover developed to perform planet exploration, scientific measurement and data collection (Pedersen [3]). The subordinate robot can achieve safe and effective navigation to the designated location and perform the desired tasks by augmenting its actions based on its local intelligence and the commands from the human supervisor. In this context, once the control is delegated to the robot (C_T) by the human, the robot will response to the delegation by adjusting its autonomy (A_T) to perform the task. Human typically assumes a monitoring role based on the information feedback by the robot (i.e. I_S). However, if the human has doubt regarding the task or robot status, he/she may exchange information (i.e. I_T) with the robot via dialog.</p>
<p><u>Teacher-Learner</u></p>	<p>Human teacher and robot learner shared and traded to transfer task knowledge (Klingspor et al. [50]) in the form of information sharing (I_S) and also information trading (I_T) to certain extent to seek understanding and clarification through inquiry by the learner (e.g. in a dialog manner, discussed in Section 2.4.3).</p> <p>Depending on the teaching method, human may either teaches the robot via shared control (C_S) (Suomela & Halme [37]) or traded control (C_T) (Papanikolopoulos & Khosla [56]) (see Section 2.2.1, parallel & serial type). In this context, the robot needs to share its autonomy (A_S) or traded its autonomy (A_T) to respond to the human control while learning the task. Shared control is normally used if the robot is required to perform the task along with the human and to provide appropriate assistance to the human. For this research, this teaching method is employed to demonstrate the teacher-learner role and relationship in Chapter 6.</p>

Figure 3.4: Phenomenon of sharing and trading induce by different human-robot roles and relationships described in Table 2.2.

Figure 3.4: Continue.

As depicted in Figure 3.4, each of the human-robot roles and relationships concentrates on different aspects of $T_{S\&T}$. Therefore, it will be advantageous if they can be integrated under the same framework of sharing and trading. One possible way to envisage these different roles and relationships of the human and the robot within an HRS is to provide multiple levels of human control and robot autonomy. In this context, each level of human control and robot autonomy will map in accordance to roles and relationships, such as those classified in Figure 3.4. Issues pertaining to this topic are further discussed in Section 3.3. Experiments conducted for assessing the achievement of seamless HRI, via the concept of $T_{S\&T}$, based on these five human-robot roles are presented in Chapter 6.

3.2.5 Task Reallocation (T_{RE})

Section 3.1.2 has discussed the role of $T_{S\&T}$ in T_{RE} . To reiterate, T_{RE} is defined as the reallocation of a current desired input task that is allocated to the human and the robot with a completely new task specification. The consideration of T_{RE} as one of the activity within an HRS leads to the differentiation of two types of $T_{S\&T}$. To distinguish, the terms *local* and *global* are introduced. *Local* $T_{S\&T}$ is defined as the ongoing HRI in performing a desired input task with the aim of improving the current HRS task performance. On the other hand, *global* $T_{S\&T}$ is defined as the reallocation of the desired input task that may involve HRI roles and relationships changes; where the change of role has completely different type of task specifications (e.g. change of role from supervisor-subordinate to master-slave, Figure 3.4). By differentiating $T_{S\&T}$ in this manner, if interaction roles transition occurs within the same task, it is considered as local $T_{S\&T}$. This implies that to achieve seamless HRI, the envisaged framework of sharing and trading must take into the consideration of both local and global $T_{S\&T}$. Experiments conducted for assessing the concept of seamless HRI due to local and global $T_{S\&T}$ are presented in Section 6.1 and 6.2 respectively.

3.3 Discussion on Framework Formulation

Given the concept of $T_{S\&T}$ introduced in Section 3.1.1 and defined in Section 3.2, the aim of this section is to discuss how this concept can be formulated into a framework. Specifically, to show how the seamless HRI can be modelled. To facilitate, Table 3.6 presents a list of basic questions for considering how a framework of sharing and trading can be used to assist in the design and development of an HRS for seamless HRI. Each of these questions is further discussed in Section 3.3.1 to 3.3.6 respectively.

Table 3.6: Basic questions to be addressed for the formulation of a framework of sharing and trading

Questions
1. Why should human and robot share and trade?
2. When should human and robot share and trade?
3. How does human and robot know when to share and trade?
4. How does human and robot share and trade?
5. What triggers the change from sharing to trading (or trading to sharing)?
6. Who is in charge of the sharing and trading process?

3.3.1 Why should Human and Robot Share and Trade?

In the context of performing an HRS task, $T_{S\&T}$ between a human and a robot is essential to let the human and the robot work together in different task situations and to ensure the overall system performance is achieved during task execution. By specifying in this manner, it does not mean the human and the robot share and trade only to deal with errors or contingency situations. They may even share and trade to provide appropriate assistance to each other during “normal operation”, e.g., to let human assists a robot in object recognition, decision-making, etc. or to let a robot assists human in remote sensing such as obstacle avoidance and guidance. This implies that they may simply share and trade to strive for better system performance or to ensure that the system performance does not degrade when the other teammate is performing the HRS task. As discussed in Section 3.1.2, such $T_{S\&T}$ process between the human and the robot may occur in an arbitrary manner, it is not feasible to automate such $T_{S\&T}$ process. The “conditions” to invoke $T_{S\&T}$ must be based on the human and the robot current awareness and perception of the ongoing task execution. This topic is discussed below.

3.3.2 When should human and robot share and trade?

An intuitive view of looking into this question is based on the invocation of specific task events. It is possible to envisage a range of invocation events in accordance to the application tasks and invoke them based on the available information in the HRS. An advantage of this is that it directly addresses the possible sharing and trading strategies. From the extreme of initial task delegation to task completion, a spectrum of events can occur during task execution. Within this spectrum, three types of events to invoke or initiate a $T_{S\&T}$ process are distinguished. The first is termed *goal deviations* where the $T_{S\&T}$ process would be invoked by human intervention. This highlights how human assists’ robot. The notion of *goal* here does not necessarily refer only to the goal of achieving a specific task, but also to the goal of attaining the overall task of the HRS. The word *deviation* refers to the departure from normal interactions between the robot and its task environment resulting in the robot being unable to achieve the goal. This also includes abnormalities arising during task execution. This may be due to either unforeseen changes in the working environment that cannot be managed by the robot; where an undesirable functional mapping from perception to action causes the robot to

“misbehave” (e.g. due to sensing failures).

The second event is *evolving situation* in which the $T_{S\&T}$ process would be invoked by the robot to veto human commands. This highlights how robot assists’ human. The types of robot’s veto actions can be loosely classified into *prevention* and *automatic correction*. Prevention implies that the robot will only impede the human actions but make no changes to it. The human is responsible for correcting his own actions. An example is when the robot simply stops its operation in a dangerous situation and provides the necessary feedback to the human to rectify his commands. On the other hand, automatic correction encompasses prevention and rectification of human commands simultaneously. Depending on the task situation, the robot may or may not inform the human how to correct his actions. For example, to prevent the human from driving into the side wall when teleoperating through a narrow corridor, the mobile robot maintains its orientation and constantly corrects the side distance with respect to the wall to align with it. In this case, the human may not be aware of this correction action and he/she is able to drive the robot seamlessly through the corridor. Based on Sheridan’s ten-level formulation of robot autonomy (see Section 2.3.1, Table 2.6), both prevention and automatic correction are positioned at level seven or higher. This is because it is the robot that judges whether the situation is safe or unsafe, as the human is unable to judge.

Finally, the third event is when both the human and the robot *explicitly request assistance* from each other. In such an event, the $T_{S\&T}$ process between the two is mixed initiated, where each one strives to facilitate the individual activities in accordance to the task situation.

3.3.3 How does Human and Robot Know When to Share and Trade?

Given the characterisation of $T_{S\&T}$ in different HRI roles and relationships in Figure 3.4 (Section 3.2.4), a basic concern towards the achievement of seamless HRI is the need for each teammate to be able to determine and be aware of and recognise the current capabilities/limitations of each other’s during the process of $T_{S\&T}$ (as depicted in Table 3.3, Section 3.1.1). The ability for the human and the robot to recognise and identify when to share and trade (Section 3.3.2) so as to provide appropriate assistance to each other is essential in developing an effective HRT. To enable the robot to assist human,

the robot needs to develop a model of the interaction process based upon readily available interaction cues from the human. This is to prevent any confusion during mode transition (see Section 2.2.3). Just as robots need to build a model to ensure effective $T_{S\&T}$, it is also important for human to develop a mental model regarding the overall operation of an HRS (e.g. the operation procedures/process, robot capabilities, limitations, etc.), to operate the system smoothly.

A good guide in ensuring that the human is in effective command within a scope of responsibility is the principles from Billings [26] (pp. 39-48): the human must be involved in the interaction process, he/she must be informed of the ongoing events (to provide as much information as the human needs from the robot to operate the system optimally), he/she must be able to monitor the robot or alternatively, other automated processes (i.e. information concerning the status and activities of the whole system) must be able to track/know the intent of the robot in the system. A good way to let human know the intention of the robot is to ensure that, the feedback from the robot to the human indicates the “reason” for the invocation or initiation action during HRC (see Section 2.4.3, Figure 2.8(f)). This implies that if the robot wants to override the human commands, the robot must provide clear indication for the human to know its intention to prevent any ambiguities. For example, during manual teleoperation, when the robot senses that it is in danger (e.g. colliding into an obstacle), the robot may stop the operation and send a feedback to warn the human in the form of a simple dialog (see Section 2.4.2).

3.3.4 *How does Human and Robot Share and Trade?*

As discussed in Section 3.1.1, the considerations of how does a human and a robot share and trade to response to changes in task situation or human/robot performance is based on the paradigm of RAH-HAR as depicted in Figure 3.2. Given the different types of cooperation strategies invoked by this paradigm (as depicted in Table 3.1), the challenge is how $T_{S\&T}$ based on RAH-HAR capabilities can be envisaged. To address, consider the characterisation of $T_{S\&T}$ in different human-robot roles and relationships in Figure 3.4 (Section 3.2.4). Based on this characterisation, Figure 3.5 is presented to depict how these human-robot roles and relationships can be employed in designing a range of task interaction modes from “no assistance provided to the human by the robot”

to “no assistance provided to the robot by the human” (Figure 3.2) for the human and the robot to share and trade.

Figure 3.5: Range of task interaction modes in accordance to the characterisation of $T_{S\&T}$ in different human-robot roles and relationships depicted in Figure 3.4

As shown in Figure 3.5, to characterise the five human-robot roles and relationships, four discrete levels of interaction modes namely, *manual mode*, *exclusive shared mode*, *exclusive traded mode* and *autonomous mode* are defined. By defining sharing and trading in this manner does not mean that trading does not occur in sharing, or vice versa in the case of sharing in trading. Here, the term “exclusive” is used to highlight that the shared mode is exclusively envisaged to let the *robot assists human*, while the traded mode is exclusively envisaged to let the *human assists robot*.

The reason of placing the shared mode below the traded mode is based on the degree of human control involvement (Figure 2.3, Section 2.3.1). This implies that in exclusive shared mode, human is required to work together with the robot by providing continuous or intermittent control input during task execution. On the other hand, in exclusive traded mode, once the task is delegated to the robot, the human role is more of

monitoring rather than of controlling or requires “close” human-robot cooperation, as compared to the exclusive shared mode. Therefore, the interactions between the human and the robot in this mode resemble the supervisor-subordinate paradigm instead of a partner-partner like interaction as in the exclusive shared mode. However, this does not mean that these two primary modes employed pre-defined robot autonomy. Within these two modes, a range of sub-modes can be incorporated with varying degree of robot autonomy (e.g. designed in accordance to varying nested ranges of action, Table 2.5, Section 2.3.1) for providing a finer grain of $T_{S\&T}$ as needed for particular applications.

Given the range of task interaction modes defined in Figure 3.5, to achieve seamless HRI, concern pertained to what triggers the change from sharing to trading (or trading to sharing) must be addressed. This is discussed in the following section.

3.3.5 What Triggers the Change from Sharing to Trading (or Trading to Sharing)?

In accordance to the range of task interaction modes defined in Figure 3.5, a transition from sharing to trading (or vice versa) may involve in a total new task specifications (i.e. global $T_{S\&T}$) or within the context of a same task specifications (i.e. local $T_{S\&T}$). Both global and local $T_{S\&T}$ are defined in Section 3.2.5 respectively. To discuss what triggers the change from sharing to trading (or vice versa) in both types of $T_{S\&T}$ process, two types of trigger, namely *mandatory* and *provisional* for global $T_{S\&T}$ and local $T_{S\&T}$ respectively are distinguished.

- *Mandatory Triggers* are invoked when there is a change of task plan by the human due to environmental constraints that may require different control strategy (e.g. from shared control to traded control), when the robot has completed performing a task leading to the specification of a new task that may required different control strategy or when task performance of the robot is perceived to be unsatisfactory resulting in human to use other control strategy, to name a few.
- *Provisional Triggers* are invoked when the human or the robot wants to assist each other to strive for better task performance. In this context, a change from sharing to trading can be viewed from a change from the robot assisting human to human assisting the robot, or vice versa in the case of from trading to sharing.

3.3.6 Who is in Charge of the Sharing and Trading Process?

The paradigm of RAH-HAR requires that either the human or the robot be exclusively in charge of the operations during $T_{S\&T}$. This means that the robot may be in authority to lead certain aspect of the tasks. This may conflict with the principle of human-centered automation, which emphasises that the human must be maintained as the final authority over the robot (i.e. the main premise of Billings' [26] principles adopted in Section 3.3.3). As this issue of authority is situation dependent, one way to overcome this is to place the responsibility of that $T_{S\&T}$ process to the human. This means that the human retains as the overall responsibilities of the outcome of the tasks undertaken by the robot and retains the final authority corresponding with that responsibility. To facilitate, apart from giving the flexibility to delegate tasks to the robot, the human need to receive feedback (i.e. what the human should be told by the robot) on the robot intention and performance¹¹ before the authority is handed over to the robot. To delegate tasks flexibly, the human must be able to vary the level of interaction with specific tasks as desirable to ensure that the overall HRS performance does not degrade. Ideally, the task delegation and the feedback provided should be at various levels of detail, and with various constraints, stipulations, contingencies, and alternatives.

3.4 Chapter Summary

The aim of this chapter is to formulate a framework of sharing and trading to assist in the design and development of an HRS for seamless HRI. The concept of sharing and trading proposed here is in the context of a task, called task sharing and trading ($T_{S\&T}$). The essential points of this concept with regards to the development of a framework of sharing and trading are as follows:

First, the approach towards the formulation of the concept of $T_{S\&T}$ seeks to provide a basis of the role of sharing and trading in task allocation (Section 3.1). This is important for explaining how this concept can be applied in the design and development of an HRS in a structured and systematic manner; which is lacking in the current literature of robotics in adopting the concept of sharing and trading for human-robot cooperation. Although the concept of $T_{S\&T}$ is proposed as a means to overcome the limitations of fixed

¹¹ Robot performance can be in term of the time to achieve the goal, number of mistakes its make, etc.

a prior task allocation during task execution (Section 3.1.1), it is not the intention for this research to offer a general process for characterising the task allocation between the human and the robot. This is because criteria for judging the suitability of various human-robot cooperation strategies are usually difficult to quantify and often implicit. In addition, it is application-specific. Rather, the intention here is to offer a means to look into different ways of HRI based on the paradigm of RAH-HAR.

Second, to address the contingencies that emerge when a human and a robot work together during task execution in a structured basis, basic activities (i.e. T_L , T_H , T_R , T_{RE} and $T_{S\&T}$) within an HRS are established to define the sharing and trading between the human and the robot (Section 3.2). As one major characteristic of the process of $T_{S\&T}$ is dynamic in nature, basic elements (namely human control, robot autonomy and information) involved in the process of sharing and trading identified in Chapter 2 is characterised within the context of a task collaboration to explain how a human and a robot might share and trade (Section 3.2.4). The purpose is to provide a basic construct towards the characterisation of $T_{S\&T}$ in different HRI roles and relationships, as established in Section 2.1.1.

Third, based on the basic elements of $T_{S\&T}$, each of the five HRI roles and relationships is characterised to define how a human and a robot might share and trade. This characterisation is substantiated by illustrating with typical scenarios from different robotics applications discussed in Chapter 2. The purpose of this characterisation is to illustrate how seamless HRI can be achieved via the concept of $T_{S\&T}$ when these five roles are integrated under the same framework of sharing and trading as discussed in Section 3.3.

In accordance to the discussion in Chapter 2 and this chapter, a framework is presented in Figure 3.6 to outline the approach of using the concept of sharing and trading for the design and development of an HRS. One of the major properties of the framework is to take into account the requirements of the human (Section 2.2 & Section 3.2.2), the robot (Section 2.3 & Section 3.2.3), and the interactions between them (Section 2.4 & Section 3.2.4).

Figure 3.6: An application-driven framework for the design and development of an HRS

This framework comprises of three phases: application requirements and analysis phase, human and robot integration phase, and the implementation and evaluation phase. The two main components in the application requirements and analysis phase, human and robot are discussed in Section 2.2 and 2.3 respectively, which are essential inputs to understand how a human and a robot share and trade. The essential inputs are the characteristics of the human and the robot. This encompasses the human and the robot roles, responsibilities, the robot functional requirements and their relationship in accordance to the task specifications. For the human and robot integration phase, the analysis of sharing and trading between the human and the robot are discussed in Section 3.1.1, 3.2.4 and 3.3, which are in turn provides essential guidelines and methods for the design and development of a target HRS for seamless HRI.

The reason of for outlining the framework in this way is to highlight the difficulties involved in this area. For example, to conduct research in this area, researchers need to deal with technical challenges such as achieving intelligent control, mobility and other special requirements of a robot, while providing a seamless interaction between the human and the robot to enable useful communication exchanges in an effective and efficient way on a variety of levels. As discussed in Chapter 2, the recent advances in robotics, AI, and other disciplines have made robots more applicable to our current society thereby increasing the opportunities for humans and robots to work together in

various ways. Hence, it is important that the design and development of a target HRS is followed by a contemporary in-depth understanding and knowledge of the related consequences and implications (i.e. scientific, technical, social and economic implications). In accordance with the framework and paradigm presented in this chapter, the concept of sharing and trading may be able to shed some light on some of the related consequences and implications for addressing the fundamental issues pertaining to the achievement of seamless HRI in a holistic manner.

This chapter has discussed the systematic development of a framework of sharing and trading. To exemplify how this framework can be applied in the design and development of an HRS for seamless HRI, the following chapter discusses its application in the modelling of a telerobotics system.

Chapter 4

Concept of Sharing and Trading in Telerobotics

In Chapter 3, a framework of sharing and trading had been established to provide a basis for the design and development of an HRS for seamless HRI. The aim of this chapter is to show how this framework can be applied in the modelling of a telerobotics system. The application illustrates how seamless HRI can be realised via the concept of $T_{S\&T}$ (Section 3.1.1 & 3.2.4) with different human-robot roles and relationships (characterised in Figure 3.5). They are master-slave, supervisor-subordinate, partner-partner, teacher-learner and fully autonomous mode by the robot as established in Section 2.1.1. The modelled telerobotics framework serves as the basic construct for the implementation of a “physical mobile” telerobotics system in Chapter 5.

This chapter begins with a discussion of the development of a sharing and trading telerobotics framework in Section 4.1. To facilitate the implementation of the telerobotics framework in Chapter 5, a discussion of the essential requirements for the design and development of a sharing and trading telerobotics system for seamless HRI is presented in Section 4.2. By examining these requirements and categorising their respective contents, a discussion of the necessary capabilities that a robot should have is discussed in Section 4.3. The intention is to imbue a robot with the necessary capabilities to become an active “team member” of HRS. The chapter concludes with a summary of the essential concepts regarding the application of the framework of sharing and trading in modelling the telerobotics system for seamless HRI.

4.1 Development of a Shared and Traded Telerobotics Framework

In accordance to the application-driven framework presented in Figure 3.6, the first phase towards the development of a sharing and trading telerobotics framework is the *application requirements and analysis phase*. The emphasis of this phase is to identify and characterise the desired application tasks for task allocation between human and robot. Given the desired inputs tasks for allocation, the second phase towards the telerobotics framework development is the *human and robot integration phase*. The primary approach of integrating human and robot is via the concept of $T_{S\&T}$, in

accordance to how human and robot assist each other. This chapter discusses these two phases; presented in Section 4.1.1 and 4.1.2 respectively. The final phase which is the implementation of the telerobotics system is discussed in detail in Chapter 5.

To provide an overview of how the first phase and second phase described above are involved in the development of the sharing and trading telerobotics framework, a conceptual structure of an HRS is depicted in Figure 4.1. This conceptual structure is derived from the activities within an HRS as defined in Section 3.2 (Figure 3.3).

Figure 4.1: A conceptual structure of an HRS

4.1.1 Application Requirements and Analysis Phase

The first component in Figure 4.1 is the task definition of a particular application goals and requirements which involves the translation of a target application goals and requirements into a “task model” that defines how a telerobotics system will meet those goals and requirements. This includes conducting studies to assess the general constraints of the potential technology available (e.g. different types of sensing devices) and environment constraints (e.g. accessibility, type of terrain) that may be useful for the telerobotics system under design. For this research, the type of applications considered are those based on mobile telerobotics concept, such as planetary exploration [3], search

and rescue [7], military operation [8], automated security [11] discussed in Section 2.1 (Table 2.1). This implies that the characteristic of the desired input task (i.e. T_I , Figure 3.3) of such applications is to command a mobile robot (by a human) to move from one location to another location while performing tasks such as surveillance, reconnaissance, objects transportation, etc.

4.1.2 Human and Robot Integration Phase

The second component in Figure 4.1 is the allocation of the desired input tasks to human (i.e. T_H , Figure 3.3) and robot (i.e. T_R , Figure 3.3). Possible analyses to the type of tasks that can only be allocated to human and robot (i.e. “who does what”) are discussed in Section 3.2.2 and 3.2.3 respectively. The difficult part is to consider tasks that can be performed by both human and robot. For example, the T_I discussed in Section 4.1.1 (i.e. *moves from one location to another location*) implies three fundamental functions: *path planning*, *navigation* and *localisation*. Both human through control of robot by teleoperation and robot have the capabilities to perform these functions. The main consideration is who should perform these functions or is it possible for human and robot to cooperate to perform these functions. In accordance to the paradigm of RAH-HAR (Section 3.1.1), this is not a problem because this paradigm takes into the consideration of timeliness and pragmatic allocation decisions for resolving conflicts/problems arising between human and robot. Therefore, it allows human and robot to perform the same function. The advantage of allowing human and robot perform the same function is that they can assist each other by taking over each other task completely when the other team member has problem performing the task. To achieve such a complementary and redundancy strategy, the approach by RAH-HAR is to develop a range of task interaction modes for human and robot to assist each other in different situations as depicted in Figure 3.2 (Section 3.1.1). The consideration for the development of a range of task interaction modes is based on the characterisation of $T_{S\&T}$ in different human-robot roles and relationships as depicted in Figure 3.4 (Section 3.2.4). Examples of a range of task interaction modes for the envisaged telerobotics system are depicted in Figure 4.2. The four main task interaction modes, namely *manual mode*, *exclusive shared mode*, *exclusive traded mode* and *autonomous mode* in Figure 4.2 are designed in accordance to the interaction modes characterised in Figure 3.5 (Section 3.3.4). The two extreme modes (i.e., manual and autonomous) are independent to each

other. They are also independent to the exclusive shared and traded modes. On the other hand, the dependency of the exclusive shared and traded modes depends on how human use the interaction modes to perform the application task. For example, to command a robot to a desired location based on environment “landmarks” (i.e., one of the navigation approach in the exclusive traded mode described in Section 6.2.2), the robot must first learn to recognise the landmarks so as to perform this navigation task. In this situation, the exclusive traded mode is dependent on the exclusive shared mode. This is because this mode facilitates robot learning of the environmental features to the desired location via teleoperation by the human.

Figure 4.2: Examples of a range of task interaction modes in the telerobotics system

The third component in Figure 4.1 is $T_{S\&T}$ between human and robot *during task execution*. This encompasses both local (for resolving human and robot conflicts,

actions and intentions) and global $T_{S\&T}$ (i.e. for task reallocation, Section 3.3.5). As this topic is related to how the concept of $T_{S\&T}$ can be implemented, it is further discussed in Section 4.2; where the essential requirements for facilitating effective $T_{S\&T}$ between human and robot are discussed.

4.2 Requirements

The development of a telerobotics framework of sharing and trading to support seamless HRI places severe requirements on the underlying system. Without a properly designed telerobotics system, the system architecture based on the concept of $T_{S\&T}$ can be ineffective and may even be dangerous. Hence, the purpose of this section is to discuss the essential requirements and the possible approaches to satisfy those requirements for the implementation of a shared and traded telerobotics system in Chapter 5.

Based on the review of literature of different HRI work in Chapter 2 and the formulation of framework of sharing and trading in Chapter 3, the essential requirements are as follows:

- To share and trade effectively, a human or a robot needs to perceive and be aware of:
 - (i) The capabilities and limitation of its team member (i.e. its counterpart);
 - (ii) The action(s) of its team member;
 - (iii) The goals or intentions of its team member; and
 - (iv) The status of its team member.

First, these four points correspond to the requirements of *shared representation* between the human and the robot. Second, they highlight the needs for both the human and the robot to *monitor* each other's actions and states so as to develop and update a model¹² of each other. Third, to alter and negotiate their communication/interaction strategies in accordance to the task or performance of each other, both the human and the robot need to *learn* from the interaction.

- To operate efficiently in response to $T_{S\&T}$ processes and changing situations, the

¹² Models define the possible set of states and their relationships.

underlying HRS should be designed to allow a shift from manual to autonomous operation dynamically. Within this manual-autonomous continuum, the Human-Robot Team (HRT) is allowed to engage in a tightly coordinated shared and traded operation to promote team cooperation. To support, the fourth requirement is, the system must be able to *resolve any arising conflicts* flexibly and dynamically. These four requirements and their associated approaches are elaborated in Section 4.2.1 to 4.2.4 respectively. To provide a roadmap, Figure 4.3 provides an overview of the four essential requirements for achieving effective $T_{S\&T}$ by highlighting the essential points and their dependencies.

Figure 4.3: An overview of the essential requirements for achieving effective $T_{S\&T}$

4.2.1 Shared Representation

To facilitate role changing during $T_{S\&T}$, the communication framework for changing responsibilities (or level of task interaction mode) must support both requesting and accepting in an interaction mode change. To facilitate this requires a shared frame of reference for effective $T_{S\&T}$, where there must be a “representation” available for sharing

and trading. This view is purported by the “Un-Fitts List” (Section 3.1.1, Table 3.2), which shows that to facilitate human and robot to share and trade effectively, they must adopt/share common ontologies. This implies that to establish what will be shared and traded during $T_{S\&T}$, they must share the same representation of the knowledge domain.

Role of Ontology in $T_{S\&T}$

There exist different kinds of ontology definitions in the literature, depending on the academic background of the researchers, e.g. philosophy, AI, knowledge engineering, etc. In the field of philosophy, ontology refers to the study of the different kinds of things that exist. In AI, ontology refers to “*the way in which a system conceives of the world external to itself, the internal representation of what is and what happens in the world*” (Messina et al. [105]). In the field of robotics or AI, it is used as basic construct in planning, learning, problem-solving, decision-making and communicating. A primary goal is to make knowledge sharable, by encoding domain knowledge using a standard vocabulary based on the ontology (Chandrasekaran et al. [106]).

A good reference that provides a vigorous analysis of the term “ontology” is from Guarino and Giaretta [107]. Here, the following definition for ontology is adopted: *an explicit account or representation of some part of a conceptualisation* (adapted from Guarino and Giaretta [107]). Generally, a conceptualisation is a world view/model corresponding to the context of the domain of interest. In the context of $T_{S\&T}$ between human and robot in an HRS, ontology is viewed virtually as the manifestation of a shared perception while performing a task (e.g. same world view/model) that is agreed between the human and the robot. Such “agreement” facilitates accurate and effective communication of task, environment and robot state information (see Section 2.4.3), which in turn facilitates the process of $T_{S\&T}$. With the adoption of the same ontologies representation, both human and robot can provide appropriate assistance to each other via *implicit* and/or *explicit* communication during task execution. This is further discussed below.

4.2.2 Monitoring

Given Section 4.2.1, to be an active team member; both human and robot need to

monitor the behaviours of each other. To provide appropriate assistance or avoid commanding the robot beyond its abilities, the human must be able to monitor the robot so as to be aware of its capabilities and limitations. To facilitate, the human needs: (1) information on the physical surroundings of the robot; (2) to be given a reasonable perspective view of the robot's current awareness of those surroundings; (3) to have some method to evaluate (and possibly negotiate solutions for) robot task successes and faults. If possible, it will be useful if the monitoring task can be viewed from different sensory perceptions. However, this requires that the underlying display interface is provided with the ability to generate and integrate multiple perspectives and representations from the robot.

To monitor the human's behaviour so as to provide appropriate assistance, the robot must have an idea of what the human wants or what his/her particular action and intention are in a particular situation (e.g. defined by the perceptions of internal and external sensors of the robot). Ideally, the human can explicitly convey his/her action and intention to the robot, e.g. by means of menu-based control interfaces (see Section 2.4.1) with intuitive feedback. This implies that the robot gets to know the action and intention of the human directly without any uncertainty. This is called *explicit communication*. If the action and intention of the human cannot be accessed directly, the robot needs to infer his/her action and intention. This is called *implicit communication*. For example, a human teleoperating a mobile robot may give "inaccurate" control signal (e.g. via a hand joystick) due to poor video feedback or perception. In this case, based on an interpretation of its sensor information of the environment together with the human control signal, the robot must infer the human action and intention and to make its own decisions to get to the desired goal safely. Following this, a challenge here is to find an approach for this implicit communication to capture the human's action and intention.

Approaches to implicit communication can be loosely classified into *model-based* and *behaviour-based* paradigms respectively. The model-based paradigm requires a formal model of human control behaviours and normally confined to the specific application tasks. Some promising approaches of model-based paradigm from the field of AI are: (1) Partially-observable Markov decision problems (Geffner & Wainer [108]); (2) Hidden Markov models (Hovland & McCarragher [109]); (3) Cascade neural

networks (Nechyba & Xu [110]); (4) Bayes rules (Demeester et al. [111]). Approaches to behaviour-based paradigm are normally based on the coordination of the different possible entities about the human's action and intention. Possible coordination mechanisms to achieve this are discussed in Section 2.3.3.

A model-based paradigm may be preferred over behaviour-based for modelling the human control strategy, but this is not suitable for the current implementation of the exclusive shared mode. A model-based paradigm requires the robot to adopt a formal model of the human user's control behaviour so as to infer his/her control action and intention efficiently. To achieve this requires many experimental trials to generate a reliable human control model. The experimental evaluation conducted (i.e., on robot assists human) in this thesis research does not allow such experimental trials. The evaluation requires the robot to capture the control action and intention of different human users in a very short time. Hence, for this research, the behaviour-based paradigm is employed to infer the human's action and intention based on his/her desired travelling direction and speed from a 2-axis joystick. This is demonstrated in Chapter 6, Section 6.1.2.

4.2.3 Learning

Learning is often viewed as an essential part of an intelligent system because the ability to learn is considered as a fundamental attributes of intelligent behaviour. It is important to incorporate "learning" into an HRS, because it includes the possibility that a robot can learn from a human and the human can learn from the robot during $T_{S\&T}$. For example, through monitoring, the human can learn from the robot by observing the actions it performs. Experience is gained through these observations and the human can learn to react appropriately to similar events when these arise. On the other hand, the robot can be seen as an apprentice that learns from the human when the human is demonstrating his/her intention by explicitly performing the corresponding operation. Work efficiency will be improved, if the best capabilities of both the human and the robot can be fully maximised in an HRS. "Learning" during $T_{S\&T}$ provides the essential mechanism to achieve this. Through learning, both the human and the robot can gradually adapt to the situation and perform the task more effectively.

Basically, there are two learning approaches under the teacher-learner paradigm (Nicolescu [112]); *learning by observation* (or watching) and *learning from experience*. For both approaches the human teacher functions as the facilitator of learning the task knowledge. In the first approach, the robot learner passively observes the teacher's performance and attempting to reproduce the observed actions. Normally, this requires the use of complex computer vision techniques to interpret the teacher's actions Kuniyoshi et al. [113]. As the teacher's demonstration in the real-world is normally noisy and partial observable, the concern for this approach is to perceive the teacher's demonstration accurately. In addition, the robot as a "learner" must also have the ability to interpret the observations and map the observed actions to its underlying programs structure. However, this approach may not be suitable for the telerobotics applications envisaged here (see Section 4.1.1). This is due to the fact that the current-state-of computer vision techniques are not robust and fast enough for the robot to perceive and interpret the teacher's demonstration accurately in a dynamic changing environment (Nicolescu [112], Kuniyoshi et al. [113]).

Instead of watching the teacher passively, the second approach requires the robot to take an active part in the demonstration. This requires performing the task along with the teacher and experiencing it through its own sensors. This avoids the problem that pertains to the first approach. This is because the robot now has an example of the task in terms of its own sensory-motor capabilities, rather than trying to map it through its own observation. Examples include learning via following the teacher (Nicolescu & Matarić [22]), mobile-robot teleoperation (Jian [114]), and so forth. For this research, the second approach is employed to demonstrate the teacher-learner role and relationships (Table 2.2) within the telerobotics framework. This is demonstrated in Chapter 6, Section 6.2.2.

4.2.4 Conflict Resolution

A framework of sharing and trading must have a conflict resolution unit, both "locally" in terms of resolving conflicts, actions and intentions (i.e. local $T_{S\&T}$), and "globally" in terms of coordination of different interaction roles (i.e. global $T_{S\&T}$) during $T_{S\&T}$. Different coordination methods can be found in Section 2.4.3. In general, approaches can be found in the framework of arbitration and command fusion presented

in the Tables 2.7 and 2.8 respectively.

4.3 Robot Capabilities

It is reasonable to argue that a human is currently the most valuable agent for linking information and action. Therefore, in an HRS, the intelligence, knowledge, skill and imagination of the human must be fully utilised. On the other hand, robot itself is a “passive component”, its’ level/degree of autonomy depends on the respective robot designer or developer. For a human-robot team, the considerations are no longer just on robotic development but rather more complex interactive development in which both the human and the robot exist as a cohesive team. Therefore, for the robot to assume appropriate roles to work with the human counterpart, the robot must have the necessary capabilities. The capabilities required by a robot (either a telemanipulator or a mobile robot) are numerous but may be classified along the following dimensions (Arkin [45], Russell & Norvig [115]):

- *Reasoning*: A robot must have the ability to reason so as to perform tasks delegated by human. In AI, the term “reasoning” is generally used to cover any process by which conclusions are reached (Russell & Norvig [115], pp. 163). By specifying in this manner, reasoning can be used for a variety of purposes, e.g. to plan, to learn, to make decision, etc. In robotics, *planning* and *learning* have been identified as the two most fundamental intelligent capabilities a robot must be imbued with so as to build a “fully autonomous robot” that can act without external human intervention (Arkin [45]). However, due to the fact that a robot must work in a real-world environment that is continuous, dynamic, unpredictable (at least from the robot’s point of view), and so forth, the goal of building such a fully autonomous robot has not yet been achieved (Giralt et al. [13], Murphy & Rogers [16], Arkin [45]).

In the context of planning and learning in an HRS, human may assist the robot in performing these functions. For example, the human can assist in solving “nontrivial” problems by decomposing the problem that must be solved into smaller pieces and let the robot solve those pieces separately. In the case of learning, human may teach the robot to perform a particular task via demonstration as discussed in Section 4.2.3. However, apart from learning from human, the robot must also learn from its task environment (e.g. through assimilation of experience) so as to sustain

itself over extended periods of time in a continuous, dynamic and unpredictable environment when performing a task. A good discussion of different robot learning approaches and considerations can be found in (Sim et al. [116]). However, to plan or learn, the robot must have a perception system to capture incoming data and an action system to act in its task environment. This is discussed below.

- *Perception*: A perception system of a robot is in the front line against the dynamic external world, having the function as the only input channel delivering new data captured from outside world (i.e. task environment) to an internal system (e.g. the software agents) of the robot. The perceptual system can play a role of monitoring actions by identifying the divergence between observed state and expected state. This action monitoring to ensure correct response against a current situation by examining an action in-situ should be an indispensable capability particularly in dynamic uncertain environments, in which the external world may change. Therefore, competent perceptual system would significantly contribute to improve the autonomy of a robot. Specifically, this is essential for the robot to monitor human control behaviours so as to provide appropriate assistance to the human as discussed in Section 4.2.2.
- *Action*: An action system of a robot is the only output channel to influence the external world with the results of deliberation (i.e. reasoning) from the internal system. Taking appropriate action in a given moment is one of the fundamental capabilities for an intelligent robot. How long a robot deliberates to find a sequence of actions to attain a goal is a critical issue particularly in real-time task domain. Sometimes it is rational to take actions reactively without planning if the impacts of those actions are minor to the whole accomplishment for the goal and easy to invoke, in emergent situations that require immediate actions.
- *Task-oriented behaviours*: A robot must have a set of behaviours to perform particular application tasks. This can range from basic behaviours such as point-to-point movement, collision prevention, obstacle avoidance to more complex task behaviours such as motion detection, object recognition, object manipulation, localisation (i.e. determining robot own location), map building, to name a few.

The discussion above provides an overview of what constitutes the robot capabilities.

A detail exposition of each of the dimensions described above is further presented in Chapter 5.

4.4 Chapter Summary

The aim of this chapter is to illustrate how the framework of sharing and trading established in Chapter 3 can be applied in the design and development of a telerobotics system for seamless HRI. Seamless HRI implies flexibility in human control through which a human interacts with a robot in different situations, and the adaptability of robot's autonomy in response to human control. The essential concepts regarding the achievement of *flexibility* in human control and *adaptability* of the robot autonomy for seamless HRI are identified as follows:

In the context of the achievement of *flexibility* in human control, the framework of sharing and trading offers a range of task interaction modes for human to control a robot in different situations. The interaction modes based on the characterisation of $T_{S\&T}$ in different human-robot roles and relationships are *complementary* mechanisms by human and robot to deal with different aspects of an HRS task. In addition, the interaction modes are also *redundant* to provide more options for the human to develop strategies to perform the HRS task. This implies that if any of the task interaction mode failures, the human can use another mode to perform the desired task. For example, to navigate from point A to point B, the human can have the option to control the robot to point B using manual mode or exclusive shared interaction mode or he/she can specify waypoint(s) for the robot to navigate to point B using exclusive traded mode or autonomous mode.

In the context of the achievement of *adaptability* of robot autonomy, the robot must have the necessary capabilities (Section 4.2) to work interactively with the human teammate. This not only encompasses functions (path planning, navigation, learning, etc.) for performing the HRS task but also functions such as monitoring human control behaviours (Section 4.1.2) in order to provide appropriate assistance to the human.

In summary, to address the problem of seamless HRI in an HRS, first of all there must be “multiple strategies” for human and robot to interact. This is because without different interaction strategies, there are no alternatives for the human to share and trade

with the robot during task execution. The approach towards the development of different interaction strategies is via the concept of $T_{S\&T}$ based on RAH-HAR. The main corollary of $T_{S\&T}$ is to strive for providing different ways for human intervention and also for the highest level of robot autonomy as possible so as to let human and robot rely on mutual support when necessary. In other words, it maximises the capability to let human and robot interact in different situations.

This chapter has provided the basic construct towards the design and development of a telerobotics system. The following chapter provided a detailed discussion of how the system is implemented.

Chapter 5

Implementation of a Sharing and Trading Telerobotics System

This chapter presents the implementation of a telerobotics system configured as a testbed based on the system framework established in Chapter 4. The system described here serves as an experimental platform to study $T_{S\&T}$ in different human-robot roles and relationships (Figure 3.4). The paradigm of RAH-HAR (Section 3.1.1, Figure 3.2) and system design in the framework of sharing and trading established in Chapter 3 are applied. To provide a roadmap, Figure 5.1 provides an overall view of the implementation of the sharing and trading telerobotics system described in this chapter by highlighting the essential points and their dependencies.

Figure 5.1: An overall view of the implementation of the sharing and trading telerobotics system

This chapter is structured as follows. The description will begin at a functional level, by describing the subsystems of the telerobotics system in Section 5.1. Subsequently, a detail description of the individual subsystems and their relationships are presented in Section 5.1.1 to 5.1.6. The description provides a basis for describing the components and coordination of the task interaction modes characterised in Section 4.1.2, Figure 4.2. This is presented in Section 5.2. Next, the software framework for implementing the components and communication protocol for such a system is described in Section 5.3. Finally, in Section 5.4 the summary of this chapter is presented.

5.1. Design and Development of the Subsystems for the Telerobotics system

Basically, there are three main subsystems in a telerobotics system. These are the human interface, the communication link and the mobile robot. The brief descriptions of these subsystems are as follow:

- ❑ *Human interface*: It is viewed as a means for interacting with robot via a communication link. The human will monitor the robot status (e.g. sensors information) through one or more displays and uses input devices such as keyboard, joystick etc., to control the robot.
- ❑ *Communication link*: It provides the underlying means for the communication between human and robot. To control the robot remotely, the communication link employed should be reliable in a typical environment.
- ❑ *Mobile robot*: A typical mobile robot for telerobotics requires:
 - Communication transceivers to interface with the human interface via the communication link.
 - Adequate sensors, such as range sensors for obstacles avoidance, detection, and location sensors to determine the location of the robot.
 - Embedded computation and program storage for local control systems, and to interpret commands, from the human interface and translate these into signals for actuation.

Schematically, the relationship of these three subsystems is depicted in Figure 5.2. This schematic is adopted from a typical autonomous mobile robot (e.g. Durrant-Whyte [117]) with an inclusion of the human interface for HRI.

Figure 5.2: Schematic of the relationship between different functional subsystems of a telerobotics system (adapted and modified from Durrant-Whyte [117])

The telerobotics system presented in Figure 5.2 is complex and composed of a number of subsystems. Each of these subsystems presents its own challenges. In order to provide a roadmap of how the telerobotics system is developed, the subsystems are classified into:

1. *Robot Hardware* – that consists of all the actuators, sensors and communication devices.
2. *Navigation* – that concerns the movement of the mobile robot.
3. *Localisation* – that estimates the position and orientation of the mobile robot.
4. *Planning* – that describes the planning of actions to be performed by the mobile robot.
5. *Interfaces* – that provides the necessary mechanisms for the human to monitor and control the mobile robot.

Schematically, the components of the telerobotics system architecture in terms of these five subsystems are organised into five levels as depicted in Figure 5.3. The decomposition of the system architecture into its subsystems encourages modularity in system development. The modular approach allows easy replacement of components in the event of component failures. It also allows experimentation of components with

similar functions, an important consideration in the system development.

Figure 5.3: Telerobotics system architecture (adapted from Ong et al. [32])

5.1.1 Robot Hardware Subsystem

This subsystem concerns the actuation, sensing and communication of the mobile robot, i.e. the ATRV-Jr™. An overview of this robot and its hardware is presented in Figure 5.4. A description of the hardware toolsets employed and the robot drive system (including the equation of motion for motion control) is given in Appendix A. The “actuation” in this case is simply the linear and angular velocity of the mobile robot as shown in Figure 5.3. Mobile robot requires a range of different type of sensors for navigation and guidance to perform useful tasks. The general problems of navigation can be summarised by three questions: “*Where am I?*”, “*Where am I going?*”, and “*How do I get there?*” (Borenstein et al. 118). The first question concerns the determination of the robot position and orientation. The second and third questions are concerned with the reasoning behaviours of the robot that normally involves planning, guidance and control. Discussions on various approaches to the required sensing hardware to address these three questions are given in Section 5.1.2, 5.1.3 and 5.1.4 respectively.

Location Sensors

The sensing methods (Durrant-Whyte [117], Borenstein et al. [118]) commonly used in determining the robot position (“Where am I”) are dead-reckoning, active beacons and artificial landmark recognition. Dead reckoning is a relative positioning system. Some common dead reckoning sensors are potentiometers, incremental and absolute encoders (optical, inductive, capacitive, and magnetic), tachometer, synchros, resolvers and inertial sensors such as gyroscopes and accelerometers (Borenstein et al. [118]). On the other hand, active beacons and artificial landmark recognition are absolute positioning system. For active beacons, the absolute position of the robot is computed from measuring the direction of incidence of three or more actively transmitted beacons. The transmitters usually use light or radio frequencies. The most common method in active beacons is the Global Positioning System (GPS). As for artificial landmark recognition method, distinctive artificial landmarks are placed at known locations in the environment for a mobile robot to reference it.

Figure 5.4: ATRV-Jr™ system description chart

Sensors for relative or absolute positioning with acceptable performance have recently become available at affordable price for commercial use. In this implementation, both relative and absolute positioning methods are used. The relative positioning is via dead reckoning and the absolute positioning is via active beacons.

Dead reckoning utilises odometry (wheel encoders) and inertia sensor for calculating the relative distance travelled (this is detailed in Section 5.1.3). The absolute position of the robot is obtained via compass (for heading) and GPS. The usage of these sensors can be classed as being either for indoor or outdoor use. For indoor environment, only odometry, inertia sensor and compass¹³ are used. On the other hand, all the four types of location sensors are used in outdoor deployment. The characteristics and operating principles of these sensors are documented in Appendix A.

Range Sensors

The mobile robot also requires various types of ranging sensors to sense/perceive its surrounding so as to acquire sufficient information to navigate safely. Currently, the most commonly used ranging sensors in mobile robots are sonar, laser range finder, radar and infra red ranging sensors. In this implementation, both sonar and laser range finder sensors are employed. Their characteristics and operating principles are documented in Appendix A. Their main applications are for obstacle avoidance and guidance (e.g. stay-on-the-path). The implementation details of the obstacle avoidance and guidance algorithms are described in Appendix B, Section B4.2. Other usages of the laser range finder are for map building and localisation. These are discussed in Section 5.1.3.

Communication

Communication plays an important role in any telerobotics system in which there is a high demand for communication between human and robot. The communication will need to be two-way simultaneous with high data rate so that command data can be transferred from the human interface to the robot, and at the same time vision, sensor, and other status information can be conveyed back from the robot to the human interface (see Section 2.4).

The requirements of the communication for telerobotics applications are numerous,

¹³ Preliminary tests show that using compass in indoor environment is affected by the electromagnetic sources (e.g. power lines) or large-magnetic structures (this is also observed in (Borenstein et al. [118], Suksakulchai et al. [119])). This causes the heading deviates from the actual reading by 15°. Hence, it is not use for localisation (see Section 5.1.3) but only as static reference. However, these magnetic interferences can be used as environment landmarks for landmarks learning. The implementation of this learning approach is described in Section 5.1.5.

requiring both digital and analogue channels, and high bandwidths, especially for streaming of video images from the robot back to the human. The successful communication network must have the following features: (a) A standard communication language and message packets between the control station and the remote mobile robots; (b) A global timing signal that allows communication of real-time data between the control station and the remote mobile robots; (c) The communication via wireless must be able to deliver the level of quality of its hard wired equivalent; (d) Degree of reliability and robustness to accommodate variations in operational requirements; (e) The speed of the communication technology must be fast so as to provide high-speed content distribution.

One common communication method that supports the above features is the Wireless Local Area Network (WLAN) using radio modem (or Ethernet). Currently, WLAN is most widely used in the robotics community. The common protocol used is IEEE 802.11b. Although this communication technology is normally used indoor (e.g. office environment), it can also be used for outdoor applications (e.g. the Man Portable Robotic System project (Manouk [120])). This communication technology has the advantage of high frequency of up to 2 GHz and high data rate of up to 3 Megabits per second (Mbps) [121], but it requires line-of-sight operation, reducing its range coverage. However, it can be overcome by using repeater(s) to increase the effective distance (Manouk [120])). The main advantage of using WLAN is derived from the fact that the most modern infrastructures already have well-established communication network supporting this communication medium. Therefore, using WLAN as a communication medium is cost effective without the need to set up another customised communication network. For this research, communication through WLAN using a radio modem, i.e. BreezeNET SA-10 PRO.11 (see Appendix A for its characteristics) is employed.

Vision System

To perform tasks such as visual surveillance, intrusion detection and intruder pursuit, a vision system is incorporated. The vision system on the ATRV-Jr™ consists of a colour video camera and a framegrabber. The camera, a Sony EVI-D30 Pan/Tilt/zoom (PTZ) NTSC video camera, is mounted on the top of the laser scanner (Figure 5.4), and the BT848 framegrabber is installed inside the on-board

computer PCI slot. Their characteristics are documented in Appendix A. Current work uses motion detection for visual surveillance and colour segmentation for intrusion detection and intruder pursuit. Their applications are discussed in Chapter 6, Section 6.2.3.

The vision system is also used for providing video feedback to the human. This is important, as video is the main source of visual feedback in many telerobotics systems (Sheridan [2]). To facilitate this, video frames are captured in 24-bit colour (RGB) with the image size 160x120, with a total of 57,600 bytes/frame (i.e. 160x120x3). To minimise bandwidth consumption, the image is compressed into a JPEG image before sending it (i.e. via the wireless Ethernet) to the display console at a rate of 3-4 frame per second (fps), depending on the network traffic. This results in an image stream which approximates to video. This visual feedback is adequate when the mobile robot is operating in either semi-autonomous or autonomous mode (Figure 4.2) where the control is intermittent or discrete (see Section 2.4.2). However, to teleoperate the mobile robot at a continuous high speed (e.g. > 0.5 m/s), 3-4 fps is not sufficient. In addition, an image size of 160x120 is too small to provide adequate visual feedback for the human to control the robot remotely at high speed. To overcome this limitation, another video channel using the Premier wireless video system (see Appendix A for its characteristics) is employed. This system provides real-time visual feedback to a 14 inch television at 30 fps.

5.1.2 Navigation Subsystem

This subsystem provides guidance and control for the mobile robot in response to information from sensors concerning the state of the environment in which the robot is situated or operating. Before proceeding to discuss the navigation approaches adopted in this research, it is important to provide a basic understanding of how the ATRV-Jr™ mobile robot moves and navigates. To facilitate, an overview of the robot reference system, basic motions and basic navigating behaviours are depicted in Figure 5.5.

Figure 5.5: Robot reference system, basic motion and navigation behaviours

Navigation Approaches

The mobile robot must be capable of both “global” and “local” navigation. Global navigation is concerned with the strategies in covering large area. To achieve, a global path planner is required. This is discussed in Section 5.1.4. On the other hand, local navigation is concerned with the details that are encountered along the way. If avoidance of obstacle is a part of local navigation, then the task of local planning is to avoid obstacles, reacting to sensory data as quickly as possible while driving towards a goal (see Section 5.1.4, reactive path planner). Approaches for robot navigation can be differentiated in accordance to how the robot actions are executed:

1. *Sense-act paradigm*, which considers navigation as purely reactive¹⁴, e.g. “avoid obstacles” (see Figure 5.5(e)), “stay-on-the-path” (see Figure 5.5(f)), for example. This approach has been widely implemented over the last decade in “behavioural”

¹⁴ This approach to navigation does not use any of the explicit representation of the environment. It also does not resort to recognise of objects or extraction of features in the environment for navigation as in the observation-based navigation approach, discuss in the following paragraph.

control of robots (Arkin [45]). The behavioural units employed in this research are further discussed in Section 5.2.2 (Table 5.2);

2. *Sense-think-act paradigm*, which requires certain degree of planning in accordance to the robot working environment. The sense-think-act paradigm can be further classified into (DeSouza & Kak [122]):

- 2a. *Observation-based navigation*: This approach uses no explicit representation about the space in which navigation is to take place (i.e. without any map), but uses objects or salient features found in the environment for navigation. It can be as simple as static or dynamic object following (see Figure 5.5(g)) to more complex techniques that use sophisticated feature extraction or computer vision algorithms. The recognition of objects or extraction of the salient features can be based on visual observation (Gaussier et al. [123]), non-visual observation (Lambrinos et al. [124]) or a combination of both (Jian [114]);

- 2b. *Map-based navigation*: This approach depends on pre-defined model of the environment for navigation. The model may be as simple as two-dimensional (2D) binary occupancy grid (see Figure 5.6(a)) or a three-dimensional (3D) CAD model of the environment (see Figure 5.6(b)).

- 2c. *Map-building-based navigation*: This approach uses range sensors to build up a local model of the environment and then uses this model for navigation. According to Leonard & Durrant-Whyte [125], this approach is extremely difficult. An example of map building is presented in Figure 5.6(c). This is because to navigate precisely, the mobile robot must have an accurate environment map; however to build an accurate map, the mobile robot's sensing location must be known precisely. The issue is which came first - a chicken-and-egg problem. Current research solution to this is based upon the simultaneity of activities to break the chicken-and-egg-loop, via the Simultaneously Localisation and Mapping (SLAM) approach (Leonard & Durrant-Whyte [125]). Details of this approach and a survey on map building can be found in (Nebot [126]) and (Thrun [127]) respectively.

Figure 5.6: Example of a 2D binary occupancy map, a 3D CAD modelled of the environment and map building¹⁵. The reference environment is the Robotics Research Centre (RRC).

Except for the map-building-based navigation approach, all the other navigation approaches are adopted. The application of the navigation approaches is task-specific. For example, reactive control (i.e. sense-act paradigm) which is fast in response is suitable for assisting (e.g. react to obstacles) human during teleoperation. Observation-based navigation is useful for performing task autonomously, such as to pursue an intruder. The application of these two approaches can be found in Chapter 6. For this research, the target operating environment is known in advance, so that map building from scratch is not required. This implies that it is possible to use a pre-defined map of the working environment for localisation, instead of the need to concurrently build map and localise at the same time. Hence, the map-based navigation approach is adopted. The application of using the pre-defined map for localisation is discussed in Section 5.1.3, *on-line map localisation sub-section*. A main problem of using a pre-defined map is it does not take into the consideration of “moving obstacles”. This issue is further discussed in Section 5.1.4, where the pre-defined map is used for global path-planning.

¹⁵ The map is built using the Carnegie Mellon Robot Navigation Toolkit (CARMEN), an open-source collection of software for mobile robot control. Available at: <http://www-2.cs.cmu.edu/~carmen/index.html>. Last visited: April 10, 2003

5.1.3 Localisation Subsystem

This subsystem estimates the position and orientation of the mobile robot. This problem is recognised as one of the most fundamental problems in mobile robotics (Borenstein et al. [118]). This is because without an accurate estimation of the robot location, performing navigation task such as waypoint(s) driving will be difficult [117, 118]. Here, waypoint is defined as an intermediate location through which the mobile robot must pass, within a given tolerance, en route to a given goal location. According to Durrant-Whyte [117], localisation is an output-only function when viewed by the rest of the system. This means that the development of localisation ability can often proceed independently from that of other system components. Three approaches to localisation, namely *dead-reckoning-based*, *on-line map-based* and *human-assisted localisation* are employed in this research. They are discussed in the following sub-sections.

Dead-reckoning-based Localisation

If only odometry is used for localisation, the mobile robot's motion controller hardware has no way to detect wheel skid or errors in wheel tracking. In our case, assumption of no-slippage between the wheels and the surface does not hold true. This is because there is considerable slippage between the pneumatic tires and the floor especially in rotation. Hence, the slippage would account for the discrepancy in the commanded rotated angle. This error has serious implications in navigation and planning. This is because any small momentary orientation error will cause a constantly growing lateral position error [128]. Hence, it would be of great benefit if orientation errors could be detected and corrected immediately. To overcome this, an inertia sensor is employed for calculating the robot orientation¹⁶ and reducing the (x, y) position errors derived from the odometry. Approaches on combining data from odometry and inertia sensor for reducing the odometry errors due to non-systematic errors such as travel over uneven floor, wheel slippages, etc. can be found in (Borenstein & Feng [128]). Current combination of the odometry and the inertia sensor data is done in a very simple calculation¹⁷ as follows: First, calculate how far the robot believes it has travelled. This

¹⁶ This is required because if the robot turns 360 ° on-the-spot, the orientation obtained from the odometry deviates from the actual reading by 30°.

¹⁷ Application of more sophisticated filtering techniques (e.g. kalman filtering) to assist in the combination of the data more effectively is beyond the scope of this research. However, the implementation facilitates

is achieved by finding the distance between the two points based on the (x, y) position from the odometry using Pythagoras' Theorem. Next, the angle from the inertia sensor gyro is used to determine the robot orientation. Combining these two information give an augmented vector for the direction of the robot. If this vector is applied to the previous (x, y) position of the robot, a new (x, y) position is derived.

On-line Map-based Localisation

The errors in the dead-reckoning system, such as systematic errors due to robot modelling errors (e.g. unequal wheel diameters) in odometry (Borenstein & Feng [128]) and data from the inertia sensor drift with time (because of the need to integrate rate data to yield orientation) (Borenstein et al. [118], Barshan & Durrant-Whyte [129]), are usually unacceptable. To overcome, with a known initial robot location, an on-line "localiser" is introduced to correct small, incremental position and orientation errors induced by the dead-reckoning system. The operating principles of the localiser are as follows:

1. Compute the robot location by the dead-reckoning system.
2. Compare the observed laser range finder values (which give the actual distance between the robot and its environment elements, e.g. wall, table, etc.) with a pre-defined map (using binary occupancy grid) of the working environment.
3. Correct the robot location from the dead-reckoning system by minimising the position and orientation differences between modelled and measured values from the laser range finder.

The implementation of the localisation algorithm is based on the Monte-Carlo Localisation (MCL)¹⁸ algorithm described by Fox [130]. The application of this approach for path planning is further discussed in Section 5.1.4.

future extension as this is a module by itself (see Section 5.3.2). This implies that filtering algorithms can be added and tested easily without the need of changing the other related modules.

¹⁸ MCL is a family of algorithms for localisation based on particle filters, which are approximate Bayes filters that use random samples for posterior estimation.

Human-Assisted Localisation

If, the on-line map-based localisation approach fails to estimate the robot location, causing the robot to lose its way, the final approach is to resort to human-assisted localisation. The approach to this is to teleoperate the robot to a known location (e.g. the starting point) and reset/reinitialise the underlying dead-reckoning system and the localiser accordingly. Through this, the complexity of the localisation problem is reduced by using the human capacities in perception and decision-making to diagnose a failure (Sheridan [2], Murphy & Rogers [53]). However, to teleoperate the robot to the known location accurately requires robot assistance. For this implementation, the human drives the robot “roughly” to the target location and let the robot align itself autonomously with the target location using its ranging sensor (i.e. laser), vision and compass (e.g. align perpendicular to the front and side walls within the specified distances respectively). This highlights the cooperation between the human and the robot. The operation of this localisation approach that illustrates the $T_{S\&T}$ between human and robot is further discussed in Chapter 6, Section 6.2.4.

Here, the decision of determining when the “robot is lost” and hence robot needs assistance can be initiated by the human or the robot. The situation of the “robot is lost” can be caused by:

- i. localisation errors (as discussed above), causing the deviation of the target goal point to an invalid goal point, e.g. out of the target navigation area;
- ii. the specification of wrong waypoint by the human, e.g. specified the waypoint too near to an object; and
- iii. getting struck in a deadlock situation (e.g. all its way are blocked) that the robot cannot recover from it.

Human determination of whether the robot has lost its way is based on monitoring of the robot navigation behaviours (Figure 5.5). In the current implementation, if the robot cannot reach the goal point due to the first and second situations, it will move in an arc manner near the target point. For the third situation, the robot will either stop or turn-on-spot. If the human is unable (unaware or not free) to determine that the robot is lost, the responsibility is delegated to the robot. Robot determination of whether it has lost its

way is based on a “timeout” approach, i.e. in accordance to the required time (required distance/travelling velocity) to reach a particular waypoint within a given tolerance. If the mobile robot cannot reach the specified waypoint based on the above condition, it will request for human assistance. Requirements for defining the contents of the exchange of information and the information feedback to the human are based on the discussion in Section 2.4.3 and Section 4.2.1.

5.1.4 Planning Subsystem

This subsystem composed of three major components as shown in Figure 5.2. They are mission planner, path planner and a learning component. The mission planner and path planner are discussed in this section, while the learning component that is concerned with providing different strategies for planning is discussed in Section 5.1.5. The purpose of the mission planner (or deliberative planner) is for establishing high level goals for the robot and the constraints within which the robot must operate. To achieve it, a relatively complete knowledge about the world is required for the deliberative planner to predict/plan the outcome of the robot actions (see Section 2.3.2). However, current implementation does not employ a deliberative planner; instead the responsibility of performing its task is given to the human. This implies that the human is not only responsible for specifying the overall system goal but is also responsible for planning, global world modeling, and localisation. On the other hand, the mobile robot is concerned with lower level goal, for example to find the “best” path via the path planner (e.g. the shortest available path) and responds to external stimuli reactively while moving towards the goals set by the human. Hence, the system architecture can be categorised under hybrid intelligence as discussed in Section 2.3.2.

Given the desired waypoints is specified by the human, the task of the path planner is to ensure that the mobile robot navigates to these waypoints safely and efficiently. For this research, two path planning strategies, namely *reactive path planning* and *global path planning* are adopted. They are discussed in the following sub-sections.

Reactive Path Planning

The implementation of the reactive path planning is based on the concept of

behavioural control (Arkin [45]) using a goal seeking and an obstacle avoidance behaviours. It is useful in situation when prior map is not available as it avoids the use of explicit representational knowledge of the world for planning. The operating principles are as follows: A straight line is plotted (using Pythagoras' Theorem) from the start point to the goal point based on the information from the dead-reckoning system (see Section 5.1.3). This provides a direction for the robot to steer towards the goal point (i.e. a goal seeking behaviour). When the robot moves towards the goal and encounters objects in the environment, an obstacle avoidance algorithm (explained in Appendix B, Section B4.2) will take over. After the robot has steered clear of any obstacles, it will recompute the line plot again and proceed in accordance with the direction given by that plot towards the goal. The cycle will continue until the robot reaches the goal point. This is depicted in Figure 5.7.

Figure 5.7: An illustration of reactive path planning

In the context of navigating in a cluttered environment, this approach is only used in multiple waypoints navigation where the distance between waypoints is short. Its application can be found in Chapter 6, Section 6.2.1. For single long distance waypoint navigation, a global path planner is employed to ensure that the robot will not lose its way or be trapped in a dead-lock situation. This is discussed in the following subsection.

Global Path Planning

In contrast to the reactive path planning, the global path planning requires the use of explicit representational knowledge for planning (i.e. a prior map). It generates a feasible and safe path from the current robot location to a goal position in an accurate grid map provided by the human. The implementation is based on an A*-based algorithm (i.e. a classic search algorithm) (Latombe [131]). It is employed to generate the optimal¹⁹ path that provides enough space between the robot and the obstacles, and efficient navigation in the immediate environment. The operating principle is as follows. The algorithm for searching the optimal path is derived by minimising the cost of moving from a given state to the goal state. The criterion²⁰ used is the distance from the given state to the goal state. The search is based on a matrix of cells from the pre-defined binary occupancy grid (the same map (i.e. Figure 5.6(a)) used for the on-line map-based localisation discuss in Section 5.1.3). Each cell has a row and column address. The cell includes the following information for planning:

1. The status of the cell to determine if the robot can move to the cell, where '0' represent empty and '1' represents occupied.
2. The direction the robot is facing when it moves to the cell. This is needed to determine the cost of moving to the next cell.
3. The previous cell the robot came from. This is needed for the reconstruction of the path from initial cell to goal cell. To facilitate this, the cell from which the robot came is stored.
4. The cost of reaching the cell and the estimated remaining cost to get to the goal cell. The estimation of the remaining distance is simply the Manhattan distance between the current cell and the goal cell, irrespective of obstacles. Manhattan distance is the distance between two points measured along axes at right angles. In a plane with p_1 at (x_1, y_1) and p_2 at (x_2, y_2) , it is $|x_1 - x_2| + |y_1 - y_2|$.

¹⁹ The optimal path is chosen by taking into account obstacles, shortest path and dead-lock situation (e.g. one way path) in accordance to a pre-defined map.

²⁰ Application of more advance criterions for constraining the cost function, such as taking account the difficulty to localise the robot, difficulty to drive the robot in a cluttered environment, etc. is beyond the scope of this research. However, the implementation facilitates future extension as this is a module by itself (see Section 5.3.2). This means that it is possible to add and test other criterions easily without the need of changing the other related modules.

As the path planning is done in a pre-defined map, if the robot encounters object that is not modelled, an obstacle avoidance algorithm (same as the one use in the reactive path planner) will take over. After the robot has steered clear of any danger, it will try to proceed towards the goal in accordance to the planned path. The application of global path planning can be found in Chapter 6, Section 6.2.1.

Application of Localisation in Path Planning

As discussed in the preceding sections, to plan a trajectory and to reach the goal, the robot must know where it is. If the path planner (either the reactive or global path planner) only uses dead-reckoning-based localisation for planning, it will suffer from the lack of positional accuracy. A path that is over 25m can essentially be rendered useless by accumulation of error in the dead-reckoning system. This is also influenced by the number of “steering” correction required to reach the goal point. The more steering correction that is required the larger the accumulation of errors. This is more of a problem for skid-steering robot (Borenstein & Feng [128]). Therefore, arriving at the desired goal point may not be possible. To overcome, one approach is to let the human plan the “best” path to reach the goal that requires the minimum number of steering actions and with the least number of obstacles via multiple waypoints. In this context, each waypoint can be viewed as a sub-goal to assist the robot to reach the final goal point. Another approach is to use the on-line map-based localisation discussed in Section 5.1.3. However, due to the extensive computations required to perform the localisation algorithms, the on-line map-based localisation takes a much longer time to output the desired robot location to the planning component as compared to the dead-reckoning system. This implies that path planning may fail due to lack of positional information.

To perform path planning effectively, there is a need to combine both the dead-reckoning-based and on-line map-based localisation data concurrently. The idea is to use the dead-reckoning data as the primary source for planning. But once there is a new on-line map-based localisation data, the distance/path to the goal point will be re-computed. As the on-line map-based localisation data is based on the global coordinate system and the dead-reckoning system is in the local coordinate (i.e. uses a relative frame), coordinate transformation is required before their data is combined. The transformation

is done in two steps. The first step takes the robot and goal points in the on-line map-based localisation coordinate system and computes the goal point relative to the robot. Given this data and the robot dead-reckoning data, the second step then computes the goal point in the local coordinate system. This information is then used for path planning. This cycle is repeated until the robot reaches the goal point.

An Alternative to Waypoints Path Planning

So far, the discussion on commanding the mobile robot to a particular location autonomously is via waypoints navigation. As discussed in the preceding sections, this approach requires an accurate localisation system to perform path planning. Here, the interest is to find an alternate way of commanding the robot to a particular location without the use of (x, y) positional information. Recent studies suggest that humans prefer angular information over distance information in the context of remembering/learning places (Healy [132]). This implies that for human to navigate from one location to another location, using angular information is sufficient. The question is: “is this applicable to robot navigation by using only angular information?” Using only angular information, the robot has the direction to navigate, but the robot may not know when it has reached the desired location. One approach to this is through the recognition of landmarks or distinct features/objects in the environment based on its sensory inputs. This is analogous to human navigation. For example, the human knows that he/she has reached the desired location because he/she has been to that location before or through other means of information (e.g. refers to the location sign board, ask someone, compare to the picture, etc.).

In robotics literature, landmarks recognition is widely used for localisation (Borenstein et al. [118], Wijk & Christensen [133]). In this context, landmarks are carefully chosen to have a fixed and known position, relative to which a robot can localise itself. In this work, the motivation of using landmarks recognition is not only for localisation, but also as a means for *location identification*. Two types of landmarks, namely artificial and natural landmarks are distinguished (Borenstein et al. [118]). According to Borenstein et al. 118, artificial landmarks are specially designed objects or markers that need to be placed in the environment with the only purpose of assisting robot navigation. On the other hand, natural landmarks are those objects or features that

are already in the environment and have a “function” (e.g. tables, chairs, pillars, etc.) other than to assist robot navigation. Here, natural landmarks are used for location identification. These “identified locations” that does not contain any (x, y) positional information is then used by the human to plan a sequence of paths to assist in robot navigation. To facilitate, a learning component is incorporated in the telerobotics system for robot landmarks learning. This is discussed in the next section, Section 5.1.5.

5.1.5 Learning

To support on-line robot landmarks learning during $T_{S\&T}$, the underlying learning system must have the following features (Sim et al. [116]):

1. The learning process must be continuous and on-line, i.e. the robot has to learn to recognise a location while interacting with the environment;
2. The learning strategy must be able to learn models of its sensors from different abstraction of sensory information (i.e. distance, pattern, features, its own internal state, etc.), so as to acquire and represent landmarks information; and
3. The learning algorithms must be robust enough to overcome the problems of noisy input data (i.e. drift in sensors) and uncertain interactions between motor commands and effects in the world.

Fuzzy Logic

One learning technique that supports the above features is learning using fuzzy logic. Fuzzy logic is a logical system that aims at a formalisation of approximate reasoning (Zade [134]). Formally, it is a structured, model-free estimator (i.e. learning the perception-action cycle without learning a model) that approximates a function either through linguistic (obtained from human experts) or numerical (obtained directly from the system) data input/output associations via fuzzy “If-Then” rules. Learning in fuzzy logic has predominantly focused on learning the fuzzy rules themselves [135]. Normally, the learning system is implemented such that it has the capability of rules generation, adaptation (parameter modification of THEN part of rule), and generalisation (modification of IF part). In the context of robot learning, it has been widely used:

- i. to provide on-line adaptation for the robot behaviours [75, 136, 137];

- ii. for finding the best control system parameters to configure the robot [75, 136, 137]; and
- iii. to provide discrete but appropriate switching of behaviours based upon the recognition of new situation via flexible arbitration policies [75, 137].

As this work requires the robot to learn directly from its sensory input, the fuzzy rules generation must be from numerical data instead of linguistic data. A suitable approach to this is from Wang and Mendel [135] that developed a method to generate fuzzy rules from numerical data. It is interesting to note that this method, applied to time-series prediction²¹ has better performance than a neural network. This method consists of five steps [135]:

- i. divides the input and output spaces of a given numerical data into fuzzy regions;
- ii. generate fuzzy rules from the given data;
- iii. assigns a degree to each of generated rules for the purpose of resolving conflicts among the generated rules;
- iv. creates a combined Fuzzy-Associative-Memory Bank (FAM) based on both the generated rules and linguistic rules of human experts; and
- v. determines a mapping from input space to output space based on the combined FAM Bank using defuzzifying procedure.

The method above by Wang and Mendel is adopted by Thongchai [136] to build an on-line learning system for mobile robot. Specifically, Thongchai applied it to develop two robot behaviours, namely obstacle avoidance and landmark detection, using sensory data from a laser range finder and a compass as the fuzzy inputs. Compass reading provides the robot's heading while the laser range finder gives the objects' location related to the current robot's heading. The results of learning to avoid obstacle indicate a good learning performance in dynamic environments. On the other hand, the results of learning to detect landmark show that the robot can detect the learnt landmarks correctly with an average of 83.32% accuracy [136]. These results show that it is possible to use Wang and Mendel method for on-line robot learning, in particular learning to detect

²¹ Time-series prediction is a very important practical problem [138]. Its application can be found in the areas of inventory and production control, signal processing, control, economic and lots of other fields.

landmark in our case.

Landmarks Learning

Here, Thongchai's idea of landmark detection using Wang and Mendel method of fuzzy rules generation from numerical data is adopted for implementing the on-line robot landmarks learning. Apart from using readings from a compass and a laser range finder as the fuzzy inputs, a visual input is also included for extracting salient visual features. The inputs and output used are defined as follows. The input provided by the compass is the robot heading (in degree). Next, three types of inputs are provided by the laser range finder. They are average obstacle distance (in metre), closest obstacle distance (in metre), and direction of the closest obstacle (in degree). The average obstacle distance is computed from all the detected obstacles which relate to the robot's position. The distance and the direction of the closest obstacle give important features that the robot can recognise as a landmark for each location. The input provided by the vision system is the attention focus of the robot represented by a point (column, row) in an image. The approach used to compute the robot visual attention is discussed in the following sub-section, visual attention system. Finally, landmarks are defined as fuzzy outputs using a sequence of number (1, 2, 3 ... n).

An overview of the learning scheme for the identification of the landmarks based on these sensory inputs is depicted in Figure 5.8. The description of the four processes in Figure 5.8 is presented in Table 5.1. Interested readers can refer to Appendix B, Section B1 for a more detailed description. The learning operation (i.e. the application for path planning) during $T_{S\&T}$ between human and robot is further discussed in Chapter 6, Section 6.2.2.

Figure 5.8: Learning scheme for the identification of landmarks

Table 5.1: Description of the four processes in Figure 5.8

Process	Description
Fuzzifier	This process maps the set of crisp sensors readings onto a collection of fuzzy input sets.
Fuzzy Rule Generation and Evaluation	This process maps the fuzzy input sets onto other fuzzy sets according to the rule base and membership functions. It comprises of two phases: rules generation during learning and rules evaluation (i.e. with the rules in the rule base) during path planning. Here, the rules generated are “AND” rules, i.e., rules in which the conditions of the IF part must be met simultaneously in order for the result of the THEN part to occur. For the problems considered, i.e., generating fuzzy rules from the six numerical data, only “AND” rules are required because the antecedents are different components of single input vectors.
Fuzzy Rule Base	This contains a collection of IF-THEN rules. The rules will define the input-output pairs (stated control) which give the specified values of the inputs and the corresponding successful outputs.
Defuzzifier	This process determines a mapping from input space to output space based on the rules. As there is only a single output landmark number, the defuzzification is done directly using rule matching, i.e. while upon detection of a landmark, a match fuzzy rule gives the output as the number that represents the identified landmark.

Visual Attention System

To integrate the data from the vision system and the other sensory inputs (i.e. the laser and the compass), a method to create crisp variables (i.e. the column and row position of the attention focus) from the vision system that could be used in the fuzzification process is needed. The method for computing the crisp variables is based on the attention operator (see Appendix B, Section B2) proposed by Itti et al. [139]. It is a conceptually simple computational model for saliency-driven focal visual attention²² inspired by the behavior and the neuronal architecture of the early primate visual system. The biological insight guiding its architecture proved efficient in reproducing some of the performances of primate visual systems. Interested readers can refer to Appendix B (Section B2) for a more detailed description. This method is demonstrated in the “Beobot Project” [142], which illustrated a mobile robot driven autonomously in real-time towards the most salient locations and objects in a visual scene perceived by the robot vision system. An illustration of this method for performing the above task is depicted in Figure 5.9. This is conducted in our research centre, RRC.

²² Generally, visual attention can be defined as the ability to select a region of interest for extracting information useful for a given task (adapted from Jian [114]). It has long been studied by researchers in psychology (Kim & Cave [140]) as well as those in computer vision (Jagersand [141]).

Figure 5.9: Sample vision attention focus frames

As shown in Figure 5.9, the attention focus is marked in “yellow” within a yellow circle. Basically, it is obtained from a measurement of how much a particular image location differs from its neighborhood (see Appendix B, Section B2). In this implementation, these attention focuses could serve as landmarks for learning (as described in the previous sub-section) and as well as to guide the robot’s navigation in autonomous mode (i.e. autonomous tracking, see Figure 4.2). For robot’s navigation, the column position of the attention focus is used to compute the turning rate that applied as a control action to the robot. In the case of learning of landmarks, the column and row position of the attention focus is used as the visual inputs.

5.1.6. *Interfaces Subsystem*

This subsystem provides the necessary tools for human to monitor and control the mobile robots. The current implementations of the interfaces are being explored on a UNIX workstation, a PC-based station and also on portable devices such as PDA to facilitate different types of HRI (e.g. proximity and remote interactions). Although the interfaces are implemented on different systems, all have similar functions, such as displaying different type of the sensors information from the robot and robot control (e.g. direct manual, semi-autonomous or autonomous control, see Section 2.2, Figure 2.2). The implementation details of these interfaces are described in Ong et al. [32].

5.1.7. *Summary*

Based on the system framework established in Chapter 4, this section has outlined the design approach and described the system architecture of the telerobotics system. The system architecture evolves around five major subsystems/components: robot hardware, navigation, localisation, planning (including learning) and interfaces described in Section 5.1.1 to 5.1.6 respectively. To facilitate $T_{S\&T}$, multiple/different methods for sensing, navigation, localisation and planning are implemented. This is essential because without these methods, there can be no rendering of assistance between robot and human. This section only presents the overall implementation and configuration of the telerobotics system; the $T_{S\&T}$ between human and robot at system-level is described in the following section, Section 5.2.

5.2 Design and Development of the Task Interaction Modes Components

Given the task interaction modes characterised in Section 4.1.2 (Figure 4.2), to understand how these task interaction modes or their sub-modes transit seamlessly at system-level, there is a need to look into the interaction modes components and how they are coordinated. As discussed in Section 2.2.3, modes transition which is situation dependent can be initiated by human, by robot or by both the human and the robot (i.e. mixed initiation). The strategies to effect mode transitions (which also provide pre-conditions for modes coordination) are addressed in Section 3.3.2. Two important attributes involved in modes transition is monitoring and intervention (Section 2.2.3). The requirements and approaches for human and robot monitoring are discussed in 4.2.2.

As for defining the different levels of human and robot intervention, the classic Rasmussen's Skill-Rule-Knowledge control behaviour model (see Section 2.2.3, Figure 2.2) is adopted. The overall concern here is to discuss the components for facilitating modes transition. The components are classified into: *human control*, *robot autonomy*, and *feedback*. The design of these components is discussed in Section 5.2.1 to 5.2.3. The coordination of these task interaction mode components is discussed in Section 5.2.4.

5.2.1 Human Control Components

These components are responsible for integrating the human control input into the robotics system. The primary function is to allow human intervention of robotics operations at different levels. In accordance to the Rasmussen's SRK model, the implementation of the human control intervention is classified into three levels as follows:

- Level 1. *Skill-based control behaviour*: Piloting or co-piloting of the robot which uses the human perception-motor coordination. This is applied in both the manual mode and exclusive shared mode via continuous human input (Figure 4.2, via joystick control). The human inputs at this level are rate control of the robot translation (metre/second) and rotation (radian/second) speed.
- Level 2. *Rule-based control behaviour*: Command the robot which uses the human perception-decision action to define intermediate navigation goal during task execution. This is applied in exclusive shared mode via human intermittent input (Figure 4.2, via camera pan control). The human input at this level is directional control of the robot heading (in degree).
- Level 3. *Knowledge-based control behaviour*: Task-level supervision (e.g. intruder detection, surveillance and perimeter patrol) that requires human perception, decision making and planning. This is applied in exclusive traded mode and autonomous mode (Figure 4.2). The human inputs at this level are as follows: (a) single-point (both relative and absolute) and multi-points (absolute) commands for waypoint navigation; (b) landmarks specification for location-based navigation; and (c) shapes and colours model specification for visual-

based task (e.g. object following). Finally, the human inputs can also be in the form of system configurations, such as sensing distance and navigation speed, activation and deactivation of particular sensors, robot autonomy and feedback components specified before operation. This applies at all three levels.

Apart from performing the above functions, the human control components also tries to derive the human control action and intention in real-time. This is to facilitate robot monitoring of the human's behaviours (Section 3.5.2) through the exclusive shared mode. The purpose is for the robot to provide appropriate assistance to the human. To track human control actions, the ten most recent human's command is kept in a First-In-First-out (FIFO) buffer. The attributes in each command include the navigation direction and the strength of the control signal. In accordance to the current implementation, the navigation direction is the direction the human points to via the joystick. As the joystick is implemented for rate control, to obtain the direction, fuzzy logic (Appendix B, Section B3.1) is employed to process the translation and rotation speed. Given the direction, the strength of the control signal is defined as the number of occurrences of each direction.

5.2.2 Components for Enabling Robot Autonomy

Components for enabling robot autonomy control how the robot operates and cooperates with the human. The components are further classified into components for operating autonomy and decisional autonomy (Section 2.3). The operating autonomy components provide means for sensing and hardware operation (Section 5.1.1). The decisional autonomy components provide means for the navigation, localisation and planning subsystem described in Section 5.1.2 to 5.1.4 respectively. To understand how the robot varies its degree of autonomy and cooperates with the human within each level of interaction modes, there is a need to discuss the navigation subsystem, where all the behavioural units are situated (Section 5.1.2).

The current implemented behavioural units are summarised in Table 5.2. Behaviour such as emergency stop uses a fixed autonomy (0 or 1); while other behaviours, such as collision prevention, obstacle avoidance, goal seeking, etc. operates with a degree of autonomy varies from 0 to 1. The conditions to vary the behaviours autonomy are

dependent on:

- i. the strength of the control signal from the human (discussed above);
- ii. the strength of the corresponding sensors information about the environment; and
- iii. the autonomy of other behaviours.

Table 5.2: A summary of the implemented behavioural units in this research

Behaviours	Functions	Requires
Emergency stop	This behaviour has the highest priority in taking control of the robot. If this behaviour is activated, all the other behaviours are suppressed and the robot stops immediately. It is designed for situation when the collision prevention and obstacle avoidance behaviours failed. This behaviour is employed in all the interaction modes.	Reading from the bumpers
Collision Prevention	This behaviour is designed for safe deceleration (Appendix B, Section B3.3) when a collision is about to occur. This behaviour is employed in both manual and exclusive shared mode.	Readings from Laser-range finder and sonar
Obstacle Avoidance	This behaviour automatically prevents and avoids any obstacle(s) detected. The implementation of the obstacle avoidance is described in Appendix B, Section B4.2. This behaviour is employed in all the interaction modes except manual mode.	Readings from Laser-range finder and sonar
Stay-on-the-Path	This behaviour is designed to let the robot automatically find a path in the environment and stay near its centre (e.g. between two parallel walls). A variant of this behaviour which is not designed explicitly is the emergent of the left or right wall following based on the situation described in Figure 5.7. This behaviour is employed in all the interaction modes except manual mode.	Reading from Laser-range finder
Goal Seeking	Task-level behaviour, designed to let the robot moves towards the target points/landmarks specified by the human (Section 5.1.4) or the target direction the human points to via the joystick. This behaviour is employed in all the interaction modes except manual mode.	Position and orientation data from the localisation system
Object Following	Task-level behaviour, designed to let the robot follow an object (i.e. using shape or/and colours) specified by the human autonomously. This behaviour is employed in exclusive traded mode.	Information from the vision system
Motion Detection	Task-level behaviour, designed to let the robot detect motion. Normally use as a pre-condition to invoke other behaviours or HRI. This behaviour is employed in exclusive traded mode.	Information from the vision system
Autonomous Tracking	Task-level behaviour, designed to let the robot navigate autonomously based on its own perception. The implementation is based on the Visual Attention System described in Section 5.1.5. This behaviour is employed in autonomous mode.	Information from the vision system
Robot State Monitoring	This behaviour is designed to ensure that the overall robot internal conditions are safe. This includes monitoring whether: (1) the robot has sufficient power to complete a particular task; (2) motors are stalled; (3) temperature is too high; (4) sensors are working properly. This behaviour is employed in all the interaction modes.	Battery level, status of all the sensors and actuators

As depicted in Table 5.2, the condition to select which behaviours should be active is

based on the interaction mode the human selected. As this topic is related to the coordination of behaviours, it is further discussed in Section 5.2.4.

5.2.3 *Feedback Components*

Feedback is important to ensure that the human is in effective command during task execution (Section 3.3.3 and 3.3.6). Apart from providing feedback displays for the human to monitor the robot (Section 5.1.6), the feedback components also provide real-time force and audio feedback to the human based on the human control action and intention, and the robot autonomy. Force feedback is used to notify obstacle information to the human. For joystick control, it is only applicable in manual or exclusive shared mode. Current implementation uses the distance (from the range sensors) between the robot and obstacle to calculate the force. As for the audio, two types of feedback, namely sound and speech are provided. Sound feedback, in the form of “alarm” is employed by the robot to attract the human attention for his/her assistance in exclusive traded and autonomous mode during autonomous task execution. On the other hand, speech feedback is used to convey information as follows:

1. *Task status*, e.g. “reach waypoint 1”, “moving to the computer room”, “target object detected”, etc. This is employed in the exclusive traded and autonomous mode;
2. *Robot assisted status*, e.g. “veto forward command”, “decelerating”, etc. This is employed in the exclusive shared mode to inform the human how the robot assists him; and
3. *Robot state warnings*, e.g. “battery low”, “sonar is not working”, etc. This is employed in all the interaction modes.

5.2.4 *Coordination of Task Interaction Modes*

A task interaction modes coordinator is required to coordinate human and robot activities to facilitate $T_{S\&T}$. In accordance to Section 4.2.4, a coordinator is required to coordinate system activities both “globally” and “locally”. In this implementation, global coordination is achieved directly via interaction mode selection by human. This implies that the human will determine which level of task interaction mode is suitable for performing a particular task either through monitoring (Section 2.2.3) or during HRI

(e.g. Figure 2.8). Through this, the coordinator will determine the appropriate robot behavioural units (Table 5.2) to use. This greatly reduces the need of coordinating all the behavioural units locally during task execution and hence requires lower computation. This is because those behaviours that are not in used are set to a lower priority (or off) according to their process ID. This form of coordination, via priority-based arbitration (Table 2.7) is an instance of trading of control as discussed in Section 2.3.3. An illustration of this coordination scheme is depicted Figure 2.7 and an overall view of interaction modes coordination is presented Figure 5.10.

Figure 5.10: An overall view of interaction modes coordination

In local coordination, a key issue is to efficiently coordinate among the behavioural units to achieve a good navigation performance as well as semi-autonomous control (i.e. shared and/or traded control). To facilitate semi-autonomous control of the robot, the human is given as much control as possible within a safe limit. The command from the human is considered as any other behavioral unit in the system and is incorporated within the overall behavior as shown in Figure 5.10. For example, in the case of a detected obstacle, some combination of the human's command (i.e. based on control input direction) and the robot's response (i.e. based on the assessment of environment by

the robot perception system) will be generated. For behaviours' coordination, this work adopted a hybrid approach based on *priority-based arbitration* (Table 2.7) and *superposition-based command fusion* (Table 2.8) as discussed in Section 2.3.3. This hybrid coordination scheme is depicted in Figure 5.11.

Figure 5.11: An example of hybrid coordination in exclusive shared mode. The notation representation is as follows: V – Motor vector, A – Degree of autonomy, Σ – Summation (superposition-based command fusion) and S – Suppression (priority-based arbitration).

5.3 Implementation of the Software Modules for the Telerobotics System

Sections 5.1 and 5.2 describe the telerobotics system at a functional level in terms of a manageable number of subsystems and components that are inter-related through data and control dependencies. The system architecture (Figure 5.3), approaches (such as navigation, localisation, planning, behaviour monitoring, interaction modes coordination, etc.), considerations and system issues describe in these two sections are in general context and can be applied in any mobile robots that uses similar system configuration. In this section, the aim is on the implementation of the subsystems and components (from here on referred to as *modules* in the software implementation context) described

in Sections 5.1 and 5.2 in a software framework. Here, a software framework is defined as a “collection of base classes from which all modules of the telerobotics system should inherit. These base classes implement the possible ways the modules can be integrated into a control program and formalise their communication” (adopted from Waarsing et al. [143]).

A key feature of the telerobotics system architecture in Figure 5.3 is that the mobile robot is controlled via distributed modules. Thus, a distributed software architecture is required to control the mobile robot. One widely used approach in the development of distributed software control in mobile robotics research is based on the Multi-Agent System (MAS) paradigm (e.g. [144], [145]). Here, the MAS paradigm is adopted. MAS and its related area of Distributed Artificial Intelligent (DAI) (see Bond & Gasser [146] for an overview) is based on the agent-based philosophy composed of (semi) autonomous and proactive software agents that share and trade with one another in order to achieve a common or individual goals. MAS paradigm offers numerous advantages over single process/threaded systems. This includes increased flexibility, fail-ability-tolerance and reusability. On the other hand, MAS also introduces new problems due to the complexity and sophistication of software agents’ interactions. These problems are well-known in robotics and are normally overcome through proper implementation of inter-agent process communications among agents (Gowdy [147]). In the context of software implementation here, the notion “agent” is termed as a module²³.

Identifying a suitable paradigm for the software framework, Section 5.3.1 and 5.3.2 present the software framework requirements and the details of the software implementation respectively. The software implementation describe is implementation-specific, i.e. in accordance to the software toolsets employed for this research.

5.3.1 Software Framework Requirements

With respect to the activities/factors enumerated from Section 5.1 and 5.2, the requirements of the software framework based on the MAS are as follows:

- i. to facilitate seamless HRI (incorporating $T_{S\&T}$) and ensure that the robot control

²³ Some of the behavioural units depicted in Table 5.2 are implemented as a module by itself while others that are closely related in functions are situated together in one module (see Section 5.3.2, Table 5.3).

system responds in time corresponding with the activities of the system in its environment, the framework should define all possible manners of communication and coordination between the different modules (including the behavioural units in each module). This includes dealing with different types of sensors and actuators components (Section 5.1.1), where each component is addressed and operated at its own sample frequency;

- ii. to facilitate modularity in system development, the framework should provide mechanisms for easy definition and combination of modules, by clearly defining the interfaces of the modules;
- iii. to facilitate varying degree of autonomy, the framework should ensure that each module is able to function semi-autonomously or autonomously, getting all the information it needs whenever possible. This includes deciding when it should operate at a lower or higher autonomy based on the readily available information; and
- iv. to facilitate the three requirements above, the framework should have a multi-thread structure with concurrency mechanisms.

5.3.2 *Software Implementation*

Generally, the overall software framework based on the MAS paradigm can be classified as human client and robot server software applications where each consists of different distributed modules connected via inter-module process communication as depicted in Figure 5.12. The modules in the robot server that operates on the onboard computer of the mobile robot are for the implementation of the navigation, localisation, planning and low-level/real-time critical tasks (e.g. image capturing, processing, and control functions). On the other hand, the human client modules are for the implementation of the human control interfaces (normally situated at remote location). Robot server modules communicate with corresponding human control interfaces proxies using TCP/IP²⁴ sockets. Both the human client and the robot server are further discussed in the following sub-sections.

²⁴ Transmission Control Protocol/Internet Protocol: a network protocol with delivery guarantees.

Figure 5.12: Software Framework

Human Client

The human client modules are written using Microsoft™ Visual Basic 6.0. Visual Basic (VB) 6 (Petroutsos [148]) was selected due to its ability to build sophisticated windows-style applications. Apart from this, VB is an ActiveX²⁵-enabled programming language. This means that VB has the capability to integrate object-oriented programming ActiveX control software to synthesis the application. Thus, VB6 was selected as the framework for development of the human client modules.

Basically, the human client modules implement the interface subsystem (Section 5.1.6). To facilitate monitoring, the human control interface provides several sensory data display (Ong et al. [32]). The laser range-finder, sonar, compass and robot (x, y) position/orientation sensory displays provide plan view of the robot and its surroundings, while the camera displays the front view of the robot's perspective. To facilitate control intervention, the human control interface includes dialogue, audio feedback (based on Microsoft™ Direct Text-to-Speech control), force feedback (based on Microsoft™ DirectX 9.0b) and control inputs. The control inputs provide means for interaction mode selection, joystick control (also based on DirectX), waypoints control, and landmarks specification (see Ong et al. [32] for an overview). All the sensory displays, feedbacks and control inputs are developed as a set of distributed modules and communicate with the respective robot server modules (described in the next sub-section) via different

²⁵ A reusable, stand-alone software component often exposing a discrete subset of the total functionality of a product or application. An arbitrary number of ActiveX controls can be used as prefabricated components to aid in building a new application. ActiveX controls cannot run alone and must be loaded

TCP/IP ports. The underlining inter-module process communications between modules in the human client are based on the Microsoft™ Component Object Model (COM), and each component can be written in any COM-compatible language, e.g., Microsoft Visual C++, Visual Basic, etc.

Robot Server

The robot server modules are written in MOBILITY™ [149], an object-oriented and distributed robot development system from iRobot Inc. using C++ language. It is developed based on the Common Object Request Broker Architecture (CORBA²⁶) 2.x standard Interface Definition Language (IDL). The IDL is a powerful descriptive language that describes the objects²⁷ and makes no operation definitions, i.e. no lines of executable code (see Henning & Vinoski [150] for an overview). The MOBILITY™ system is selected for the implementation of the robot server modules because it is the de facto programming environment for iRobot Inc. robots. This implies that program developed (based on the MOBILITY™ Class Framework) for one robot platform can be ported over to others without major changes. This is an important consideration because there are three ATRV mobile robots in the Robotic Research Centre. Other reasons of using the MOBILITY™ system include:

- i. the support for the integration of different types of sensors (e.g. Sonar, SICK laser range finder, KVH-C100 compass, etc.) that is widely used in mobile robot researches;
- ii. the support of parallel and distributed processing for robot control, which in turn facilitate the implementation of appropriate control laws with guaranteed operation; and
- iii. the support of high software component reuse making easy to implement modules.

Table 5.3 lists the robot server modules. These modules are implemented based on

into a control container such as Microsoft Visual Basic or the Microsoft Internet Explorer. Formerly referred to as OLE control or OCX.

²⁶ CORBA (<http://www.corba.org>) is a vender-independent specification promoted by the Object Management Group (OMG) (<http://www.omg.org>). A middleware such as CORBA is simply a tool for connecting objects across heterogeneous processing nodes, but does not implement an application by itself.

²⁷ An object is a software entity which encapsulates a set of methods (routines which can be invoked) and attributes (values that can be read and set).

subsystems and components described in Section 5.1 and 5.2 respectively. In total, there are seventeen modules in three principal categories, i.e. *human control*, *robot autonomy* and *feedback* enumerated from Section 5.2.1 to 5.2.3 respectively. In accordance to the current implementation, each robot server module sets up a refreshing rate according to its own requirements, runs asynchronously and provides different “services” either to the other modules or to the human client. To facilitate inter-module process communications, the CORBA Naming Service²⁸ [151] is exploited. This service provides a mapping from module names to object references. For example, given a module name (“DeadReckoning”), the services return the object reference (e.g. the compute robot (x, y) position/orientation, timestamp, etc.) store under the name to the module who subscribed for it. Through this, modules can communicate with each other meaningfully using name instead of having to deal with long naming object references. In short, the communication between two modules can be viewed as a relationship between a publisher (i.e. service provider) and a subscriber (i.e. user).

Table 5.3: Robot server modules

Module Name	Function (or service provided by each module)
<i>Human Control</i>	
HumanControl	Processes the human client control inputs. This includes deriving the human control action and intention.
<i>Robot Autonomy</i>	
Base	Provides abstractions for motor control and also for getting sensor information. The type of sensors information provided in this module includes odometry, sonar and bumpers.
Laser	Provides abstractions for getting the SICK laser range finder information.
Compass	Provides abstractions for getting the KVH-C100 compass information.
AHRS	Provide abstractions for getting the Crossbow Attitude & Heading Reference System (AHRS) 400CA information.
PTZ	Provides abstractions for Sony Pan/Tilt/Zoom (PTZ) NTSC camera control
Framegrab	Provides abstractions for image capture via the BT848 framegrabber
VisualAttention	Provides abstractions for getting the robot visual attention focus (Section 5.1.5).
DeadReckoning	Provides Position and orientation estimation based on odometry and AHRS (Section 5.1.3).
Localiser	Provides Position and orientation estimation based on the on-line map-based localisation (Section 5.1.3).
Navigator	Performs navigation tasks. This module encapsulates the followings behavioural

²⁸ This service is similar to the Internet Domain Name Service (DNS), which translates Internet domain names (e.g. www.atrv1.ntu.edu.sg) into IP addresses (e.g. 155.69.127.35). Both the Naming Service and DNS implement simple mapping from a name to a lookup value are often likened to a white pages phone, which maps subscriber names to telephone numbers (Henning & Vinoski [150]).

	units (Table 5.2): emergency stop, collision prevention, obstacle avoidance, stay-on-the-path and goal seeking. Each of these behaviours runs at different threaded so as to achieve concurrency.
Planner	Performs path planning based on a pre-defined map (Section 5.1.4)
Learning	Performs landmark learning based on fuzzy logic (Section 5.1.5).
Vision	Performs visual tasks. This module encapsulates the followings behavioural units (Table 5.2): object following, motion detection and autonomous tracking.
Monitor	Monitors the robot state (Table 5.2). This includes data recording.
Coordinator	Coordinates the above modules (Section 5.1.2).
<i>Feedback</i>	
Feedback	Provides the necessary information to the human client. This module encapsulates the followings feedbacks: range and location sensors, images, battery level, temperature, messages, etc. To ensure the seamless feedback, each of these feedbacks runs at different threaded with distinct TCP/IP port and communication with the respective human client modules independently.

In general, the implemented robot server allows concurrent control of robot sensors and actuators; and receives commands, executes them, and sends the necessary information (e.g. sensory data, execution results, etc.) back to the human client. This includes proper management of request priorities (between different modules) and execution.

5.4 Chapter Summary

This chapter presents a detailed implementation of a mobile telerobotics system that incorporates concepts of sharing and trading in a seamless way. To facilitate the incorporation of sharing and trading, Section 5.1 first present a typical telerobotics system architecture for describing $T_{S\&T}$ between human and robot at system-level in Section 5.2. The telerobotics architecture (Figure 5.3) is based on the architecture for a typical autonomous mobile robot but with the inclusion of the human control interface for HRI. Although the system architecture is based on an autonomous mobile robot, the approach for the design and development of each telerobotics subsystems (e.g. navigation, localisation, planning, etc.) is based on the considerations of how RAH-HAR invoked by the concept of $T_{S\&T}$. Given the detail description of the subsystems in Section 5.1, Section 5.2 describes the components for implementing the task interaction modes characterised in Section 4.1.2, Figure 4.2. This is follow by a description of the approach for coordinating these interaction modes components. Finally, in Section 5.3, to realise the implementation of the telerobotics system, the subsystems and components

describe in Section 5.1 and 5.2 are implemented into software modules within a software framework. In total, there are seventeen modules in three principal categories, i.e. *human control*, *robot autonomy* and *feedback* as depicted in Table 5.3.

Given the implemented telerobotics system described in this chapter, Chapter 6 describes the experiments conducted. The aim is to illustrate the concepts of $T_{S\&T}$, in particular, the paradigm of RAH-HAR (Section 3.1.1, Figure 3.2) and the concept seamless HRI due to change of human-robot roles (identified in Section 2.1.1 and characterised in Section 3.2.4, Figure 3.4).

Chapter 6

Experimental Evaluation

It is essential that research into robotics be proven by experimental study of concepts and ideas, and validation of algorithms. Many problems in robotics evolve from phenomena that occur during its interaction with a dynamically changing environment which do not lend themselves to a formal treatment (Arkin [45], Brooks [77]). Thus experimental studies of robotics system would preferably be based on real-world experimentation using physical robots, as opposed to the use of simulation (Brooks [77]). In this context, a widely subscribed approach for the evaluation of robotics system is the use of *proof-of-concept*, which is to show that the system is capable of accomplishing the particular task it is designed for (e.g. (Casper & Murphy [7], Bruemmer et al. [18], Nicolescu & Matarić [22], etc.). Consequently, this approach is adopted to investigate the concept of $T_{S\&T}$ (i.e. task sharing and trading) established in Chapter 3; where task sharing is defined exclusively for letting robot provides assistance to human, while task trading is defined exclusively for letting human provides assistance to robot. Hence, to assess the cooperation between human and robot based on $T_{S\&T}$, the experimental evaluation is divided into two main parts, namely *robot assists human* (RAH) and *human assists robot* (HAR) presented in Section 6.1 and Section 6.2 respectively.

The aim of evaluating RAH is to assess how human can control a robot flexibly with varying degree of robot assistance. On the other hand, the aim of evaluating HAR is for assessing how human can delegate control flexibly to a robot with different types of human assisted strategies, such as planning, teaching and localisation. In the context of evaluating seamless HRI due to change of different human-robot roles, the experimental evaluation on RAH addresses human-robot roles transition with same task specification (i.e. local $T_{S\&T}$), while the experimental evaluation on HAR addresses human-robot roles transition with completely different type of task specification (i.e. global $T_{S\&T}$). Both local and global $T_{S\&T}$ are described in Section 3.2.5. Figure 6.1 provides an overall view of the experimental evaluation presented in this chapter by highlighting the essential points and their dependencies.

Figure 6.1: An overall view of the experimental evaluation

6.1 Experimental Evaluation on Robot Assists Human

In this section, a user's evaluation is conducted to investigate the concept of task sharing, from the perspective of how assistance can be provided to the human by the robot. The intention here was to assess the *Master-Slave* and *Partner-Partner* human-robot interaction roles described in Figure 3.4, and the concept of seamless HRI that arises due to change of these human-robot roles with same task specification (i.e. local $T_{S\&T}$). This is depicted in Figure 6.2. This section is structured as follows. First the objectives of the experiments are stated, discussed and formulated in Section 6.1.1. Subsequently, the RAH scheme used in the experimentation is described in Section 6.1.2. Section 6.1.3 describes the experimental design and procedure followed by a presentation of the experimental results and the discussion of the results in Section 6.1.4 and 6.1.5 respectively.

Figure 6.2: An overview of the human-robot roles and roles transitions for the experimental evaluation on robot assists human

6.1.1 Experimental Objectives

To assess how robot assists human in a seamless way, the following two hypotheses are established:

- **Hypothesis 1:** HRS task performance can be improved if the robot can assist the human as compared to when no assistance is provided to the human.

The formulation of Hypothesis 1 is to assess the cooperation between human and robot based on the concept of task sharing, from the perspective of RAH. By specifying Hypothesis 1 in this manner, it is primarily to establish a basis to show that there is a difference in task performance between robot assistance provided to the human and when no robot assistance is provided. The purpose is to provide a basis for assessing human-robot roles transitions from no assistance to maximum assistance provided to the human by the robot as depicted in Figure 6.2.

- **Hypothesis 2:** A better performance can be achieved if the human and robot can change their interaction roles dynamically as compared to the use of fixed interaction role.

The formulation of Hypothesis 2 is to justify the need of using different human-robot roles and flexibility in roles changing. The aim is to assess seamless HRI due to local $T_{S\&T}$. Local $T_{S\&T}$ is the ongoing interaction role changes between human and robot in performing a desired input task with the aim of improving the current HRS task performance (as defined in Section 3.2.5). Following this, to assess the validity of these two hypotheses, it is essential that a set of performance criteria used for the experimental evaluation clearly characterises how the task performance is improved. However, the selection of an appropriate set of performance criteria is specific to the application of the system. Therefore, to evaluate these two hypotheses, it is necessary to relate to them in the context of performing a specific application task so that the task performance can be investigated.

The telerobotics system, described in Chapter 5, is an implementation of the concept of sharing and trading using different interaction roles. The primary application envisaged is for surveillance, reconnaissance, objects transportation, etc. in which the tasks of navigation, obstacle avoidance and target tracking are just some examples (see Section 4.1.1). Thus, the navigation task has been implemented in Section 5.1.2. For this system, a navigation task moving from location A to location B can be performed:

- ❑ Manually by the human. That is, via manual mode, based on the Master-Slave human-robot role, where the human is not provided with any robot assistance.
- ❑ Cooperatively by both the human and robot. That is, via exclusive shared mode, based on the Partner-Partner human-robot role, where appropriate robot assistance is provided to human.
- ❑ Autonomously by the robot. This approach is further described in the experimentation for HAR in Section 6.2.1.

Here, the intention is to use both the manual and exclusive shared modes to evaluate the two hypotheses put forward above. In the context of performing a navigation task via teleoperation, the evaluation of Hypothesis 1 can be achieved by comparing the performance obtained from manual mode and exclusive shared mode. For example, one of the performance criteria in this experiment is the task completion time (Section 6.1.3); the assessment is which interaction mode allows the human to complete the task course

at a shortest time. On the other hand, to evaluate Hypothesis 2, there is a need to have an experimental condition that can facilitate the change of interaction roles. Note that both the manual mode and exclusive shared mode constitute two different interaction roles respectively (Figure 6.2). Therefore, to achieve the change in interaction roles, both the manual and exclusive shared modes must be applied concurrently during task execution. Based on this, the evaluation of Hypothesis 2 can be achieved by the comparison of the performance obtained from experiment on the interaction roles transitions with the experiments on manual mode and exclusive shared mode.

An important requirement for the evaluation of the two hypotheses is the need for a robust RAH scheme in real-time. A basic working principle of the RAH scheme is described in the following section.

6.1.2 Robot Assists Human (RAH) Scheme

The main objective of the RAH scheme is to let the robot provide appropriate guidance (e.g. stay on the path) and obstacle avoidance to the human while performing a navigation task under teleoperation in exclusive shared mode. This scheme is situated in the Coordinator module (Table 5.3) and is implemented based on the following three modules, namely HumanControl, Navigator and Monitor (Table 5.3). The development of this scheme is obtained through a process of iterative refinement. This involves the use of the human control actions and intentions (from the HumanControl module), and the robot behaviours (from the Navigator and Monitor modules). This process is iterated by the human designer until the robot can provide appropriate assistance to the human at a sufficient degree of accuracy.

The RAH scheme proceeds in the following steps:

1. The coordinator in the robot control system concurrently acquires the human control inputs, the robot perceptions of the environment from the range sensors and the robot internal state.
2. Based on the inputs from procedure 1, the coordinator's concurrently derived the human control actions and intentions and calculate the robot behavioral units' autonomy (Section 5.2.1).

3. Based on the computation from procedure 2, the coordinator's selects the most appropriate action. The action in this case is the robot translation velocity (v) and angular velocity (ω).
4. The selected action from procedure 3 is then used to control the robot.
5. The procedure is repeated from step 1 onwards until the navigation task is terminated by the human.

The coordination architecture that realises this procedure is depicted in Figure 5.11. A central element in this procedure is the need for an approach to facilitate robust real-time obstacle avoidance that can navigate around unknown objects in a dynamic environment. In fact, such an approach is widely used in the field of robotics (e.g. a mobile robot or a robotic manipulator), to perform its task safely in the working environment. Here, the aim of HRS is to provide a safe and seamless (i.e. ideally without stopping while avoiding obstacles) navigation for a mobile robot. This implies that the adopted obstacle avoidance approach must not only facilitate the detection of an obstacle but also must allow for the robot to negotiate the obstacle, which requires some kind of quantitative measurements concerning the obstacle's dimensions [152]. Some of the well-known real-time obstacle avoidance approaches include, *edge or boundary detection* [153, 154], *certainty grid* [155, 156], *artificial potential field* [77, 84, 85] and *vector field histogram* (VFH) [157, 158, 159]. A review of these four approaches is presented in Appendix B (Section B4.1). Here, the obstacle avoidance algorithm employed here is based on the concept of VFH but augmented with the capability of decision making using the immediate sensor readings. VFH is selected due to its ability to perform robust real-time obstacle avoidance at a fast speed without stopping, as compared to the other three approaches. The main reason of augmenting VFH with immediate sensor reading is to facilitate behavioural coordination. This is important because the VFH approach is not a reactive methodology as it requires maintaining a world model. Therefore, it is difficult to integrate with other behavioural units (e.g. collision prevention, stay-on-the-path, etc.) which use reactive control that is "representation-free" (Section 2.3.2). The implementation of this obstacle avoidance approach is documented in Appendix B (Section B4.2). Following this, several RAH scenarios are presented in the following sub-section to highlight various aspects of the

RAH scheme used in this experimental evaluation. Figure 6.3 provides an overview of the four RAH scenarios described in the following sub-section.

Figure 6.3: An overview of the robot assists human scenarios

Robot Assists Human Scenarios

Generally, the RAH scheme described here is in the form of providing autonomous guidance and obstacle avoidance. As shown in Figure 6.3, the *first scenario* presents a situation whereby the command by the human causes the robot to deviate slowly towards the left. If the robot complies with the human control action, it might crash with the left object. To overcome this, the robot needs to provide appropriate assistance to the human by constantly correcting its left distance relative to the object autonomously so as to maintain straight line motion. Thus, the human does not need to worry about issuing precise control actions and can concentrate on the overall objective of the task, such as driving through the working environment as fast as possible when performing an exploration task.

The *second scenario* presented in Figure 6.3 shows how the robot assists the human in performing a U-turn task. In this case, the human just needs to indicate the desired direction, i.e. forward-right. Based on this control action, the robot then autonomously makes a U-turn by constantly correcting its right distance relative to the object. If no assistance is provided for this U-turn task, the human needs to make the following decisions so as to complete this task: when to turn; how much turning is required; and when to stop turning and move forward. Performing this task can be difficult, in particular in remote driving where human perception of the remote environment is limited.

The *third scenario* presented in Figure 6.3 shows two cases of how the robot assists the human: (1) entering a narrow passage; and (2) moving through the corridor safely by maintaining near the centre. This assistance is useful as it dispenses with the need for the human to exercise precise command as in first and second scenario. Based on a command of the desired direction by the human, the robot's decision of whether to cross the passage or continuing follow the corridor depends directly on when the human indicates his control action and intention. For example, if the human indicates his intention of entering the passage before the passage's opening, the robot can assist the human and provide fine maneuver to drive through the passage. On the other hand, if the human indicates his control action and intention of entering the passage only after or at the passage opening, the robot will proceed to follow the corridor.

The *fourth scenario* presented in Figure 6.3 shows that when the robot approaches an object perpendicularly, two cases can occur: (1) the human wants to drive the robot as near as possible to the object and park the robot; and (2) the robot assists the human by performing autonomous obstacle avoidance maneuver and continues to move forward in accordance to the direction indicated by the human. These two cases correspond to two different modes of operation, parking and obstacle avoidance. To provide spontaneous assistance, it is important that the robot can differentiate between these two basic operations autonomously without the need of mode switching by the human. The differentiations of these two cases are as follows:

- a. The robot first assumes that the human is directing it to collide with the object and start to decelerate.
- b. During the deceleration phase:
 - i. If the human continues to maintain the direction without increasing the speed (i.e. via the joystick), the robot will continue to decelerate and stop near the object (i.e. parking).
 - ii. On the other hand, if the human maintains the direction and pushes the joystick harder (assuming that the human already knows that there is an obstacle in front based on video feedback from the robot on-board camera and he/she still wants to go forward) and exceeds a predefined speed threshold, the robot will infer the direction (based on the approach discussed in Section 5.2.1) that the human really wants it to go forward and perform the obstacle avoidance (in this case).

Using these four scenarios as the basic constructs, a navigation test course is designed for conducting the experiments. This is further discussed in Section 6.1.3, *test environment sub-section*.

6.1.3 Experimental Design and Procedure

A user evaluation was conducted to investigate Hypothesis 1 and 2 (Section 6.1.1). The task was for a human to drive the ATRV-Jr™ mobile robot remotely through a test course based on the following three experimental conditions:

- ❑ **Experimental Condition A - Manual mode:** Apart from providing safe operation

such as emergency stop in situations such as communications breakdown or an impending collision with an object, the robot does not provide any other assistance to the human.

- ❑ **Experimental Condition B - Exclusive shared mode:** Operation with robot assistance as described in Section 6.1.2.
- ❑ **Experimental Condition C – Adaptive interaction modes:** Operation with mode varying by the human from the manual mode to exclusive shared mode or vice versa during task execution so as to let the human control the robot in different situation. Between the extremes of manual mode to that of exclusive shared mode, there exists a range of transition sub-modes implemented based on different sensing distances and angles.

Basically, the operation based on these experimental conditions requires the human monitoring the robot actions and controlling the robot by initiating and terminating each of the robot actions in sequence based on his/her sensory-motor coordination. The experiment setup to facilitate this is discussed in the following section.

Experiment Setup

One difficulty of conducting experiments in telerobotics is that the interaction between the human and the robot is bounded by a human-robot interface. Therefore, care must be taken to ensure that the human-robot interfaces used in the experiment can provide effective HRI. Ideally, the interfaces employed should be as simple as possible without the need of using sophisticated control interfaces that may be difficult to learn. As this experiment focuses on remote driving (without line of sight), the simplest interfaces to perform this task are to provide the human with a visual display for monitoring the robot remote environment and an input device (e.g. via a joystick, keyboard, etc.) to issue command to control the robot (Sheridan [2]). Current experimental setup provides real-time video from the camera on board the robot for monitoring and uses a joystick to issue a stream of commands that control the translation and rotation speed of the robot. Apart from visual feedback, audio and force feedbacks are also provided in all the interaction modes. This is to let the robot inform the human of any danger and also how assistance is provided in exclusive shared mode. The

implementation of these feedbacks is described in Section 5.2.3. An overview of the experiment setup is depicted in Figure 6.4.

Figure 6.4: Experiment Setup

Test Environment

The experiments took place at RRC. The layout of RRC is shown schematically in Figure 6.5. The test course is approximately 184 square metres, 27 metre long by 4 metre wide. As shown in Figure 6.5, the test course comprises of two narrow straight paths (~1.4 metre in width) and two U-turns. To provide realistic settings for the experimental evaluation, minimal changes were made in the test environment. Obstacles in the test environment include:

1. Chairs and tables of different sizes located on the centre of the test course (Figure 6.6 (a)). The chairs could be removed or occupied.
2. Computer work benches located by the sides of the test course. Obstacles includes cables from the work benches (Figure 6.6 (b))
3. Cabinets located by the sides of the test course (Figure 6.6 (c)). The doors of the cabinets could be opened or closed, determined by the centre occupants.

Figure 6.5: Schematic layout of the test environment – RRC

Figure 6.6: Snapshots of the various perspectives of the test environments

Participants

Six participants (3 men and 3 women), ages from 19 to 44, took part in this experimental evaluation. All participants were unpaid volunteers from RRC and all participants completed the entire study (i.e. the three experimental conditions.). Four participants have driving license and three participants had prior experience with remote driving. However, none of the participants had controlled the mobile robot before. These data was obtained via a pre-evaluation questionnaire documented in Appendix C.

Experimental Design

The experiment is based on the principle of *Latin square* (Kirk [160]), which is common in psychological research. This approach is adopted because it is able to

minimise the effect of any factor that may vary through the duration of the experiment such as environmental factors in the field situation. The effect would be to cancel each other out especially if there is a gradient effect such as training or learning effect in this experiment. In addition, it is useful in serial experiments where different treatments (i.e. the three experimental conditions) are given to the same subjects (i.e. the six participants) in sequence. This is further discussed below.

Latin square design involves the use of two sets of blocks: one of which is organised by rows and the other by columns. The basic idea of this design is to randomly assign the experimental conditions within rows and columns, with each condition once per row and once per column. For this experiment, the rows correspond to the six participants, while the columns correspond to the three experimental conditions. The training and learning effects can be minimised by rotating the participants to different conditions. Each participant was required to complete two trials for each experimental condition. The experiment was carried as a replicated four 3×3 Latin square, as depicted in Table 6.1. Replication was employed to provide an opportunity for the effects of uncontrolled factors to balance out, and thus aids randomisation as a bias-decreasing tool.

Table 6.1: Experiment design based on a replicated Latin square

Participants	Experimental Conditions (A – Manual mode, B – Exclusive shared mode & C – Adaptive interaction modes)					
	Replicate I (Trial I)			Replicate II (Trial II)		
	1	2	3	1	2	3
1	A	B	C	C	B	A
2	C	A	B	A	C	B
3	B	C	A	B	A	C
4	C	B	A	A	B	C
5	A	C	B	C	A	B
6	B	A	C	B	C	A

The performance criteria (or measure) for assessing the experimental conditions (i.e. the different interaction modes) are as follows:

1. Task completion time. This is for determining the efficiency of each interaction mode. Interaction modes with faster task completion times are regarded as more efficient.

2. Number of collisions (i.e., actual contact) with obstacles in the test environment. This is for determining the safety of each interaction mode. Safer interaction modes correspond to fewer collisions.
3. Number of stops (i.e., translation and rotation speed equal to zero for more than one second). This is for determining the ease of operating each interaction mode. Interaction modes that correspond to fewer stop are regarded as easier to operate as the participants are more confident in controlling.
4. Number of turn-on-spots (i.e., translation speed equals to zero for more than one second). Same as criteria 3 above. Interaction modes that correspond to fewer or no turn-on-spot during driving are deemed to be easier to operate (i.e., smooth driving).

The data values for the above criteria are obtained on-line by means of automatic logging during task execution. However, side collision of the mobile robot wheels with obstacles that cannot be obtained automatically by the front and rear bumps is gathered by observation by the experimenter.

Questionnaire

Apart from assessing the experimental conditions based on the above evaluation criteria, a subjective rating is also employed to determine how participants feel about the evaluation, i.e. user satisfaction (Scholtz & Bahrami [161]). This is important because it allows participants to submit their analysis of the control interface and interaction modes, as well as their opinions. The questionnaire design is detailed in Appendix C.

Experimental Procedure

There are three main stages to the experimental evaluation: pre-evaluation, actual evaluation and post-evaluation. This is discussed in the following sub-sections.

Pre-evaluation

At the beginning, each participant was required to fill up the pre-evaluation questionnaire (Appendix C, pp. C3). The purpose of this was to gauge the experience of the participants, which might affect the opinions and results. Subsequently, they were briefed on the hardware of the ATRV-Jr™ and the control interface (Figure 6.4). This

provided a better understanding of the robot that they were controlling. Next, participants were given a verbal explanation accompanied with a demonstration as to how to drive the ATRV-Jr™ in their line of sight using the control interface. The purpose was to allow them to observe what their actions on the control interface would have on the actual robot.

Once the participant was comfortable with the control interface, the session entered a training phase in which the participant practiced all the three interaction modes without line of sight. To minimise training and learning effects, the interaction mode presentation (i.e. the sequence of using the interaction modes) was randomly selected for each participants as in Table 6.1. The selected interaction mode presentation remained the same during the actual evaluation. The participant then practiced all the interaction modes until he/she displayed an acceptable level of competence of each interaction mode. The time spent by the participants on each interaction mode was about ten minutes. All training was done on a small training course, which includes the four navigation scenarios described in Section 6.1.2 (Figure 6.3). It is important to note that training was done off the actual test course so that the participant was not able to learn anything that would assist him/her during the actual evaluation.

Actual evaluation

The actual evaluation was explained to the participant. This explanation included the route of the test course (Figure 6.5) and the goal of the evaluation (i.e. the evaluation of Hypothesis 1 and 2) along with which interaction mode they should start with for each trial (Table 6.1). For each interaction mode, the participant was asked to complete the navigation task as quickly as possible, while making sure the robot was safe. Figure 6.7 shows some sample frames from one run captured by the vision system and Figure 6.8 shows the robot front perception representation of the test environment for the same run as described in Appendix B (Section B4.2, Figure B7(a)).

To determine when the task had been completed and to count the number of side collisions between the robot wheels and obstacles (that cannot be automatically recorded by the data logger), the robot operation was observed throughout by the experimenter. After the task had been completed, the participant was informed by the experimenter and

the system was shutdown. The total evaluation time for each participant to complete two trials was approximately three hours with a fifteen minutes interval between each experimental condition for the documentation of the recorded data.

Figure 6.7: Sample video frames from participant 2 (trial 1) via exclusive shared mode

Figure 6.8: Sample robot front perception representations of the test environment from the same run recorded in Figure 6.7

Post-evaluation

After the accomplishment of the evaluation, participants were asked to answer the post-evaluation questionnaire (Appendix C, pp. C4 to C5). The participants answered each question by ticking one appropriate box in every question. After the questionnaire, participants were then invited to critic the three interaction modes.

6.1.4 Experimental Results

For both trials, all the participants were able to complete the test course. Results of the four performance measures are summarised in Table 6.2. A full listing of the results can be found in Appendix D, Table D1 to D4. To analyse these four performance measures, the Analysis of Variance (ANOVA) technique [160] is employed. ANOVA is a technique by which variations associated with factors can be used for testing of significance. The factors in this experiment are: (a) different types of interaction modes, (b) participants, (c) variation in run to run and (d) random error. The objective of the analysis of these four performance measures is to validate Hypothesis 1 and 2 presented in Section 6.1.1. The ANOVA table for task completion time is presented in Table 6.3 and the ANOVA tables for the other three performance measures can be found in Appendix D, Table D5 to D7.

Table 6.2: Results of the experiments

Performance Measure	Trial	Experimental Conditions (A – Manual mode, B – Exclusive shared mode & C – Adaptive interaction modes)					
		A		B		C	
		Mean	Standard Deviation	Mean	Standard Deviation	Mean	Standard Deviation
Task completion time (seconds)	1	206.17	27.59	182.17	7.36	142.67	18.64
	2	196.17	14.72	174.67	10.77	147.17	17.25
Number of collisions	1	3.67	1.49	0	0	1.5	0.76
	2	3.83	1.57	0	0	0.67	0.75
Number of stops	1	22.67	5.28	0.17	0.37	2.33	1.89
	2	21.17	3.72	0	0	1.83	2.27
Number of turn-on-spots	1	13.67	3.548	0	0	0	0
	2	14.5	3.10	0	0	0.167	0.37

Table 6.3: ANOVA table for task completion time

Sources of Variation	Sum of Squares	Degree of Freedom	Mean Square	F	F _{0.05}
Different types of interaction modes	19215.50	2	9607.75	57.62	3.37
Participants	6432.67	5	1286.53	7.72	2.59
Variation in run to run	441.17	2	220.59	1.32	3.37
Error	4335.67	26	166.76		
Total	30425.01	35			

In Table 6.3, the sources of variation are the factors in the experiment. The “Sum of Squares” column represents the measurement of variation due to the parameters. The “Degree of Freedom” is defined as the number of independent observations for the performance measure. The “Mean Square” is the estimates of variance in “Sum of Squares” per “Degree of Freedom”. The “F” value is the variance ratio of the factors to the error source of variation. The last column, $F_{0.05}$ represents the significant ratio found in F-distribution table (Kirk [160]) at 5% significant level.

From the ANOVA analysis, there are significant differences between the different types of interaction modes for all the performance measures. For the measure of the task completion time, the use of the adaptive interaction modes took the shortest time to complete the navigation course, whilst the manual mode took the longest time. On average, the exclusive shared mode is faster by 22.75 seconds (11.3% improvement) over the manual mode. On the other hand, the adaptive interaction modes are faster by 56.25 seconds (28% improvement) and 33.5 seconds (19% improvement) over the manual mode and the exclusive shared mode respectively. This result is expected since human is given the flexibility to adjust to the required degree of robot assistance as needed between the manual (no robot assistance) and the exclusive shared mode (maximum robot assistance).

For the measure of the number of collisions, exclusive shared mode had zero collisions, whilst the manual mode had the most number of collisions. Therefore, the exclusive shared mode is the safest. The average number of collisions under the manual mode and the adaptive interaction modes is 3.75 and 1.1 respectively. However, for both modes, there was no direct collision with the obstacles in the test environment. Most of the collisions were due to the robot wheels coming into contact with the leg of the chairs observed by the experimenter. This is because such collisions could not be recorded by the data logger automatically.

For the measure of the number of stops, exclusive shared mode had zero stop, whilst manual mode had the most number of stops. This indicates that the participants were more confident in driving the robot using the exclusive shared mode. The average number of stops under manual mode and adaptive interaction modes was 21.92 and 2.08 respectively. For the manual mode, most of the stops occurred in performing the U-turns

(Figure 6.4). As for the adaptive interaction modes, the stops occur during mode transition. However, as compared to the manual mode, the number of stops in the adaptive interaction modes is not significant.

For the measure of the number of turn-on-spots, exclusive shared mode had zero number, whilst manual mode had the most number. This indicates that the exclusive shared mode was easier to operate as the participants did not need to constantly steer the robot on the spots to adjust the navigation so as to stay on path. The average number of turn-on-spots under the manual mode is 14.09. Most of the turn-on-spots is for the correction of over-steering during turning. As for the adaptive interaction modes, the average number of turn-on-spots is 0.08. This number is not significant as compared to the manual mode.

Questionnaire Results

The participants were asked to evaluate the generic control interface and the three interaction modes by giving a score from 1 (worst) to 9 (best) via the post-evaluation questionnaire described in Appendix C (pp. C4 to C5). A full listing of the questionnaire results can be found in Appendix D, Table D8 to Table D11. The average rating of the overall reaction to the control interface was 7.67. This rating indicates that the control interface was sufficient for the participants to perform the task. This was purported by the outcome of the experiment as all the participants were able to complete the test course via this control interface.

As for the interaction modes, the average rating of the overall reaction to the manual mode was 5.59. The average rating of the overall reaction to the exclusive shared mode was 7.88. The average rating of the overall reaction to the adaptive interaction modes was 7.25. To compare the three types of interaction modes, ratings from the questions, “How easy was it to drive the robot?” and “Learning to operate”²⁹ were analysed using an ANOVA test respectively at 5% significant level. The ANOVA tables for these two questions can be found in Appendix D, Table D12 and D13 respectively. From the ANOVA analysis, there were significant differences between the

²⁹ This question is to assess the difficulty in learning how the robot response to the human control during teleoperation.

different types of interaction modes for the question, “How easy was it to drive the robot?”. For this question, exclusive shared mode had the highest average rating of 8.00; followed by the adaptive interaction modes that had an average rating of 7.00, whilst manual mode had the lowest average rating of 5.17. On the other hand, there were no significant differences between the different types of interaction modes for the question, “Learning to operate?”.

Participants Comments

Each participant was requested to list the positive and negative aspects of the interaction modes they evaluated. No participants preferred the manual mode over the exclusive shared mode or the adaptive interaction modes. However, four participants preferred the exclusive shared mode over the adaptive interaction modes. All the participants commented that it was not easy to drive the robot using manual mode. This is supported by the ANOVA test discussed above that shows that manual mode has the lowest usability rating. The common criticisms of the manual mode were the need to be on constantly alert for obstacles and the difficulty to maintain a smooth curvature movement of the mobile robot during turning. The main criticism of the exclusive shared mode was the robot tended to slow down when entering the narrow path way. Another criticism was the overriding of the human command by the robot for safe navigation was frustrating for some participants. Finally, the main criticism of the adaptive interaction modes was the difficulty in deciding the right timing to perform the mode transition for some participants.

The main positive aspect of the manual mode listed by the participants was the robot response exactly to the human command as compared to the exclusive shared mode, that sometimes they were required to “compete” control with the robot (during the overriding phase) to force the robot to continue in the direction and speed they desired. This happened when driving along the two narrow straight paths as the robot constantly reacted to the cables from the computer work benches (Figure 6.6(b)) causing the robot to steer and slow down. The positive aspects of the exclusive shared mode listed by the participants include safe navigation (in particular during turning) and less attention is required in driving. Some participants even commented that real-time video feedback was not required when driving under exclusive shared mode as compared to manual

mode. Finally, the main positive aspect of the adaptive interaction modes listed by the participants was the flexibility in changing interaction mode for different navigation situations.

6.1.5 Discussion

The analyses of the results of the objective measures presented in Section 6.1.4 show that participants were able to operate more efficiently with robot assistance (i.e. exclusive shared mode) than when no assistance is provided (i.e. manual mode). This result provides considerable evidence in support of Hypothesis 1 (Section 6.1.1), which states that task performance can be improved if the robot can assist the human as compared to when no assistance is provided to the human. The advantages of the exclusive shared mode are it automatically adjusts for drift, guidance (e.g. during turning) and obstacles avoidance (see Section 6.1.2) whereas the manual mode does not. However, as compared to the manual mode, the exclusive shared mode could not reach the maximum speed (i.e. 0.6 metre/second for this experiment setup) because the robot was constantly reacting to the obstacles in the test environment to ensure safe navigation. On the other hand, the manual mode allowed the participants to operate constantly at the maximum speed. Although participants were able to drive faster under manual mode, they took a longer time to complete the test course. This was due to the need to make numerous adjustments to compensate for drift when travelling along the narrow paths and also to make fine adjustments to achieve smooth turning.

Taking the advantages of both the manual and exclusive shared modes described above; participants completed the test course in a much shorter time under the adaptive interaction modes as compared to the use of a “fixed” single interaction mode, i.e. either under the manual or the exclusive shared mode (Section 6.1.4, Table 6.3). This result provides significant evidence in support of Hypothesis 2 (Section 6.1.1), which states that a better performance can be achieved if the human and robot can change their interaction roles dynamically during task execution as compared to the use of fixed interaction role. The discussion on the concept of seamless HRI that arises due to change of human-robot roles is further discussed in Section 6.3.

6.2 Experimental Evaluation on Human Assists Robot

The aim of the experimental evaluation reported in this section is to evaluate the concept of *task trading*, from the perspective of how assistance can be provided to the robot by the human. Apart from this aim, this section is intended to assess the concept of seamless HRI due to change of human-robot roles with completely different type of task specification (i.e. global $T_{S\&T}$). The assessment is based on how human assistance can be provided to robot seamlessly in different task situations. To facilitate the evaluation of how assistance is provided to the robot by the human and their interaction roles, four types of experiments were conducted as follows:

- *Waypoints navigation*: The task is for the human to command the robot from one location to another location via waypoint(s) control (both multiple waypoints and single waypoint). Assistance provided to the robot by the human is in the context of planning a safe path for the robot to navigate. This experiment is presented in Section 6.2.1.
- *Landmarks based navigation*: The task is for the human to command the robot from one location to another location via landmarks (i.e. environmental features). As compared to waypoint navigation, this approach does not require a pre-defined map (i.e. a prior knowledge of the environment) for human to plan. However, to employ this approach, the human must first control (i.e. via teleoperation) the robot to the desired location so as to let the robot acquired the environmental features. Assistance provided to the robot by the human is in the context of teaching the robot to navigate to the desired location. This experiment is presented in Section 6.2.2.
- *Vision based tasks*: The tasks for this experiment are as follows: (1) to command the robot to perform a sentry task (which includes following people) and (2) to command the robot to follow another mobile robot, teleoperated by a human. For both cases, assistance provided to the robot by the human is in the context of training the robot to track the desired object on-line if the object specification is not in the robot database. In addition, to retrain the tracking of the object if the robot lost track of the object during task execution. This experiment is presented in Section 6.2.3.
- *Human-assisted localisation*: The task is for the human to assist the robot in localising the robot (x, y) position and orientation (i.e. heading). This experiment is

presented in Section 6.2.4.

Table 6.4 provides a summary of how the concepts of task trading and seamless HRI are illustrated in these four experiments.

Table 6.4: A summary of the four experiments conducted for human assists robot

Experiments	Descriptions
Waypoints navigation (Section 6.2.1)	<p>Here, multiple waypoints control illustrates the <i>Supervisor-Subordinate</i> human-robot role, while single waypoint control illustrates <i>Fully Autonomous</i> human-robot role as follows (diagrams obtained from Figure 3.4):</p> <div style="display: flex; justify-content: space-around; align-items: center;"> </div> <div style="display: flex; justify-content: space-around; align-items: center;"> <p><u>Supervisor-Subordinate</u></p> <p><u>Fully Autonomous</u></p> </div> <p>For both cases above, control trading (C_T) from the human to the robot is for the human to delegate the desired waypoint(s) to the robot. To response to the human control delegation, the robot must adjust its autonomy (i.e. autonomy trading (A_T)) so as to navigate to the desired waypoint(s). Information sharing (I_S) from the robot to the human is for the human to monitor the robot task execution, while information trading (I_T) is to allow human and robot exchange information via dialog. In this experiment, I_T occurs when the robot informs the human that it has reached the first waypoint and request for human permission to proceed to the other waypoints.</p>
Landmarks based navigation (Section 6.2.2)	<p>This experiment illustrates the <i>Teacher-Learner</i> human-robot role as follows:</p> <div style="display: flex; justify-content: center; align-items: center;"> </div> <p style="text-align: center;"><u>Teacher-Learner</u></p> <p>For this teaching method, the human uses shared control (C_S) teleoperation to teach the robot how to get the desired location. In this context, the robot needs to adapt its autonomy (i.e. autonomy sharing (A_S)) so as to respond to the human control while learning the salient landmarks. I_S from the robot to the human is to allow the robot provides video feedback to the human for teleoperation. I_T between the human and robot is via dialogs in which the robot seeks human advice whether a particular identified landmark should be saved or not. In addition, it is also for human to give meaningful name to the identified landmark. I_S from the human to the robot is to facilitate the human to transfer task knowledge (i.e. human correlation of the identified landmarks with the environment via naming) to the robot.</p>

	The assessment of seamless HRI due to human-robot roles transition is based on the change from Teacher-Learner (for learning of landmarks) to Supervisor-Subordinate (for task execution using the learned landmarks). The assessment shows that to provide human the flexibility in assisting the robot (in this case, teaching), the robot must imbue with the required capability (i.e. learning) to response to the human teaching so as to execute the task seamlessly.
Vision based tasks (Section 6.2.3)	This experiment illustrates the <i>Supervisor-Subordinate</i> human-robot role as follows: C_T from the human to the robot is for the human to delegate the tracking task to the robot. To response to the human control delegation, the robot must adjust its autonomy (i.e. A_T) so as to follow the people or robot. I_S from the robot to the human is for the human to monitor the robot task execution. I_T between the human and robot is via dialog for the robot to request for human assistance for object recognition.
Human-assisted localisation (Section 6.2.4)	The assessment of seamless HRI due to human-robot roles transition is based on the change from Partner-Partner (for teleoperation) to Supervisor-Subordinate (for waypoints control). This experiment shows that if the robot does not keep track of its location during teleoperation (i.e. rate control), the robot may not be able to execute the task properly when the human changes to waypoints control. This is because the robot has lost track of its location in its environment. As a consequence, seamless HRI will not be achieved.

For all the four experiments summarised in Table 6.4, the primary task was remote navigation, performed by the robot either semi-autonomously or autonomously. Therefore, the concept/methodology discussed here can be applied to applications, such as planetary exploration [3], search and rescue [7], military operation [8], automated security [11] discussed in Section 2.1 (Table 2.1). These four experiments were conducted using the telerobotics system described in Chapter 5. They were conducted at the RRC.

6.2.1 Waypoints Navigation

In Section 6.1, the navigation task of controlling the mobile robot from one location to another location is performed by means of “constant” rate control, either manually by human (i.e. via manual mode) or cooperatively by both the human and the robot (i.e. via exclusive shared mode). The first problem of using this style of HRI is that it requires high communication bandwidth for the human to monitor the remote environment and to make necessary commands to control the robot’s actions. Secondly, even with sufficient communication bandwidth, the human may experience “cognitive fatigue” due to constant monitoring and commanding. The situation becomes worse if the human requires operating the robot for a long period of time. This exacerbates the fatigue problems that may lead to lower efficiency.

One of the solutions in overcoming these problems is via waypoints navigation, i.e. via planning a set of goal points by the human and let the robot moves to the waypoints by itself, instead of exercising constant control as above. However, to use this style of HRI, the robot must have the required capabilities, such as navigation (Section 5.1.2), localisation (Section 5.1.3) and planning (Section 5.1.4) to perform the task semi-autonomously or autonomously. Although this style of HRI does not require the human to constantly monitor and command the robot's action, prior knowledge is required for the human to specify appropriate waypoints. These include the robot starting location and the location of goal points. Therefore, path planning using this style of HRI via a prior map is not suitable for applications such as exploration of unknown environment or search and rescue. This is because these applications usually do not have a prior knowledge of the environment for the human to plan. This approach is only applicable if the robot is equipped with SLAM capability (Section 5.1.2) for the robot to navigate autonomously. However, using this approach to plan a short path relative to the robot current location for the robot to traverse between waypoints is still possible, as long as the human has an overview of the robot navigation environment.

In accordance to the current implementation of the telerobotics system described in Chapter 5, both multiple waypoints and single waypoint navigation are designed based on the Supervisor-Subordinate and Fully Autonomous human-robot roles (Section 4.1.2, Figure 4.2) respectively. The main difference between the two is that in the case of multiple waypoints navigation, the robot is only capable of local reactive path planning (Section 5.1.4, Figure 5.7). Therefore, it requires human assistance to pre-plan the path in advance for it to navigate. On the other hand, for single waypoint navigation the robot is capable of performing global path planning (Section 5.1.4) to select its own routes, requiring no human assistance except stopping in emergency situation or due to a change of plan by the human. The experiment conducted for both the multiple waypoints and single waypoint navigation strategies are presented in the following sub-sections.

Multiple Waypoints Navigation

The objective of this experiment is to let the robot traverses between a set of waypoints semi-autonomously. The ability to perform this basic navigation task is important for applications that require telerobotic control of remote mobile robot (Gage

[8], Carroll et al. [11]). This experiment was conducted at the premises of the RRC. The outline of this navigation course is shown schematically in Figure 6.9. The navigation course is approximately 1218 square metres, 58 metre long by 21 metre wide. As shown in Figure 6.9, four waypoints were planned by the human to let the robot navigates one round around RRC premises. As simple as this task may seem, it is not possible for the robot to execute it successfully without human assistance in planning the navigation route in advance. This is because this task requires the robot to make critical navigation decisions such as how to traverse from the interior to the exterior of RRC and back to the interior of RRC again. For this scenario, even with a global path planner that uses a prior map, it is also difficult for the robot to plan a path to navigate efficiently (e.g. via the shortest path), as compared to the path planned strategically by the human.

Figure 6.9: Schematic layout of the multiple waypoints navigation course – RRC

This experiment was conducted using the exclusive traded mode, implemented based on the supervisor-subordinate human-robot role. In this interaction mode, the human does not need to manage the remote robot totally, but he must know the status of the robot. Whenever the robot encounters problems, the human must be ready to assist the robot. There are two situations in which the task trading between the human and the robot are invoked in this interaction mode. The first is through “robot-initiate”. This

implies that the robot will acknowledge the human when it has completed a particular task or it requires human assistance. The second is through “human-initiate”. This simply means that if there is an event triggered by the human (i.e. via an intervening control input) the robot will stop and wait for the human command. However, if there is no event triggered by the human to request the robot to perform other tasks or to assist the robot, the robot will continue to perform its current task autonomously. The first situation is illustrated in this experiment. The second situation is illustrated in Section 6.2.4. The exclusive traded mode coordination architecture for performing multiple waypoints navigation is presented in Figure 6.10.

Figure 6.10: Exclusive traded mode coordination architecture for waypoints navigation. The notation representation is as follows: V – Motor vector, A – Degree of autonomy, Σ – Summation (superposition-based command fusion) and S – Suppression (priority-based arbitration)

As shown in Figure 6.10, the three main behaviours employed by the robot to traverse between waypoints are goal seeking, stay on the path (Section 5.2.2, Table 5.2) and obstacle avoidance (Appendix B, Section B4.2). In accordance to these three behaviours, the navigation task proceeds as follows:

1. Once the human has planned the navigation path, he/she then sends the information of the waypoints to the robot for execution. The state flow for describing the robot execution is depicted in Figure 6.11.

Figure 6.11: Multiple waypoints navigation state flow

2. As shown in Figure 6.11, immediately after starting, the robot traversed towards waypoint 1 (Figure 6.9) in accordance to the motor vector from the goal seeking behaviour. The obstacle avoidance behaviour starts to operate at a higher autonomy when the obstacle density was high. On the other hand, the ‘stay on the path’ behaviour starts to operate at a higher autonomy when traversing along path that was wide enough for the robot to stay in the middle. The concurrent execution and coordination of these behaviours constitutes the reactive path planning described in Section 5.1.4. A graphical illustration of how these behaviours are coordinated is depicted in Figure 5.7.
3. Once the robot reaches waypoint 1, the robot will notify the human that it has managed to traverse to the exterior of RRC. If the human did not acknowledge, the robot would proceed to waypoint 2 after the timeout period. The purpose of this notification is to inform the human that the communication between the robot and the control station is still alive. This is important because if the communication was down in this interaction mode, the robot would proceed to navigate autonomously until it had completed the navigation task. Figure 6.12 shows some sample video frames of this experiment.
4. The stay-on-the-path behaviour dominated while traversing from waypoint 1 to waypoint 4. This is because the corridor gap (~2.4 metre) is wide enough for the robot to stay at the middle at all times. However, the obstacle avoidance behaviour

would dominate if there were people walking along the corridor. Once the robot steers clear of people, the robot would proceed to stay at middle of the corridor and travel in the direction of the next waypoint. The whole operation stopped once the robot reached waypoint 4.

Figure 6.12: Sample video frames of multiple waypoints navigation taken by the robot's vision system

Single Waypoint Navigation

The objective of this experiment is to command the robot to perform single long distance waypoint navigation autonomously. The outline of the navigation course is shown schematically in Figure 6.13. The navigation course is approximately 34 metre long by 12 metre wide. The obstacles in the test environment included moving people and obstacles described in Section 6.1.3.

Figure 6.13: Schematic layout of the single waypoint navigation course – RRC

In this experiment the robot adopted the fully autonomous role. The autonomous mode coordination architecture for the single waypoint navigation is similar to the exclusive traded mode coordination for multiple waypoints navigation presented in Figure 6.10. The only difference is the goal seeking behaviour is replaced with a planner that is capable of generating a safe and feasible path for the robot to traverse using a prior 2D binary occupancy map (Figure 5.6(a)). The working principle of this planner is described in Section 5.1.4.

As shown in Figure 6.13, the ideal path the robot should take to reach the goal was to traverse directly towards entrance A. This is because this path is the shortest with respect to the goal point. However, in this experiment the robot traversed towards the upper part at location A instead of travelling straight ahead. The main reason the robot

traversed towards this direction is due to cables from the computer work benches that are not modelled in the occupancy map. In this case, the obstacle avoidance behaviour took over and steered the robot towards the right. When the robot had steered clear of obstacles and tried to proceed in accordance to the planned path again, it was too late for the robot to traverse back to the original planned path. This is because there was insufficient clearance for the robot to traverse due to a static obstacle in front. This navigation route is more challenging as it is longer and also required the robot to perform more steering (see Section 5.1.4 for the consequences) as compared to the original navigation route. This also caused the robot to move towards the dead-lock situation depicted in Figure 6.14.

Figure 6.14: A snapshot of the dead-lock situation in Figure 6.13

Despite the problematic situations discussed above, the robot was able to complete the task successfully. The sample video frames and the actual robot trajectory of this experiment are shown in Figure 6.15 and Figure 6.16 respectively.

Basically, the human assistance provided to the robot in this experiment was in the form of abstract goal point specification. In comparison to the experiment of multiple waypoints navigation, it does not require the human to plan a safe and feasible path for the robot to traverse. In this experiment, the task trading between human and robot occurred as follows:

- ❑ Task initialisation: The human passed control to the robot to perform the navigation task autonomously once the human has initialised the task; i.e. specified the desired goal point as depicted in Figure 6.13.

Figure 6.15: Sample video frames for single waypoint navigation

- ❑ Task completion: The human took over the control from the robot once the robot has reached the goal point as depicted Figure 6.16.
- ❑ Task execution: Trading can occur during task execution when the robot encounters novel situation that it cannot handle (e.g. trap in the dead-lock situation as depicted in

Figure 6.14) and hence requires human assistance, or a situation in which the human notices an opportunity to improve the robot task performance (e.g. a shorter path to the desired goal point).

Figure 6.16: Actual x-y plot of the robot trajectory for single waypoint navigation

As this interaction mode is capable of letting the human to control the robot via abstract goal-oriented commands and required less frequent interactions with the robot, it is suitable for a human to manage a fleet of autonomous mobile robots. One of the potential applications that may require this style of HRI is in the military for supporting future combat systems (Matsumura [162]).

6.2.2 Landmarks Based Navigation

Instead of using waypoints, this experiment illustrates another means of commanding the robot to the desired location using natural landmarks (e.g. tables, chairs, cabinets, etc.) in the environment. The outline of the navigation course is shown schematically in Figure 6.17. The navigation course is approximately 24 metre long by 8 metre wide. The obstacles in the test environment are the same as those described in Section 6.1.3. The objective of this experiment is to command the robot via natural landmarks to location B (Figure 6.17), a cluttered work area that is not modelled in the occupancy map

provided to the robot. Therefore, it is not possible to command the robot to this location with the waypoints navigation strategy.

Figure 6.17: Schematic layout of the landmarks based navigation course – RRC

For navigation using natural landmarks (Section 5.1.5), the robot must first learn to recognise the landmarks. Therefore, this experiment consists of two phases, *landmarks learning* via teleoperation and *task execution*, as described in the following sub-sections.

Landmarks Learning

The purpose is to let the human teacher shows the robot learner (i.e. teacher-learner human-robot role) how to get to the target location. In order for the robot to associate its own sensory signal/ features with particular motors events, the robot must have the ability to reason about the environment. To facilitate this, the exclusive shared mode is employed for teleoperation. Another reason of using this interaction mode is that it provides appropriate robot assistance to the human for safe teleoperation (Section 6.1.2).

As described in Section 5.1.5, the robot identifies each landmark by four different features. One feature is from the vision attention system. It provides the attention focus of the robot represented by a point in the image. The other three features are from the laser range finder for detecting objects in a specific location. These are the average of

obstacle distance around the robot, the closest obstacle direction, and the closest obstacle distance. Hence, these four features are combined with the compass reading which gives the robot heading using the fuzzy learning scheme described in Figure 5.8. The output of this learning process is a sequence of number for defining each specific landmark. This implies that different landmark numbers are needed to represent a particular navigation route (e.g. location 1 to location 2 in Figure 6.17) using this approach. To prevent the robot from generating too many unnecessary landmark numbers, a simple “human decision making” loop is incorporated into the learning process to let the human assists the robot in deciding whether the generated landmark number should be saved. Another purpose is to let the human stipulate a name to each landmark instead of using number. Utilisation of a naming convention is a more meaningful and easier to remember than using numbers. The state flow for describing the learning process above is depicted in Figure 6.18.

Figure 6.18: Landmarks learning state flow

The three main sensory inputs recorded in this phase are shown in Figures 6.19 to 6.21. Figure 6.19 shows some sample frames captured by the vision attention system. Figure 6.20 shows the laser range finder plot. Figure 6.21 shows the compass plot.

Figure 6.19: Sample vision attention focus frames for the learning phase taken by the robot's vision attention system. The attention focus is marked in "yellow" within a yellow circle.

A compass is an absolute heading sensor. This means that the reading is always relative to the North. This is evident in Figure 6.21. Around sample time 16, the compass heading changes from 0° to 360° and back from 360° to 0° at sample time 155.

In accordance to the test environment in Figure 6.17, the North (i.e. 0°) is facing the top of the diagram. Therefore, the start position and the target position are located at the South-East and North-West respectively. As mentioned in Section 5.1.1, a compass is sensitive to the electromagnetic interferences from the environment. This is also evident in Figure 6.21. At the starting position, the robot is facing the East (i.e. 90°), but the compass reading only gives 75° . However, it is also evident from Figure 6.21 that the electromagnetic interferences from the test environment are static as there are no major changes in the compass heading (e.g. angle increased or decreased rapidly). Hence, it can be used to provide directional information to the robot.

Figure 6.20: Sample laser range finder plot for the learning phase in accordance to the sample video attention frame in Figure 6.19

Figure 6.21: Compass heading for the learning phase (direction with respect to the navigation route in Figure 6.17)

Task Execution

The second phase is the execution of the navigation task. With the learned landmarks, the robot was put back into the same environment to carry out the navigation task. This task is carry out using the exclusive traded mode, implemented based on the supervisor-subordinate human-robot role. The exclusive traded mode coordination architecture for the landmarks navigation is similar to the exclusive traded mode coordination for multiple waypoints navigation presented in Figure 6.10. The only difference is that the human input to the goal seeking behaviour is landmarks with directional information obtained from the first phase instead of waypoints. The implementation is the same as the human control of the robot by providing directional information to the robot via a joystick in exclusive shared mode.

For this phase, two evaluation trials were performed to test whether the robot can reach the target location shown in Figure 6.17 using the learned landmarks. The test environment for both trials differs in terms of obstacles arrangement, e.g. chairs placement and moving people, which are determined by the centre occupants. Despite this, the robot succeeded to reach the target position in both trials. The trajectory of the robot in both the trials is shown in Figure 6.22.

Figure 6.22: Actual x-y plot of the robot trajectory for landmark based navigation

The approach described here using natural landmarks for navigating in an unknown environment is similar to the map building approach, which is commonly used in autonomous navigation (Thrun [127]). They are similar in that both approaches required human assistance in exploring the unknown environment via teleoperation first before using it for navigation. As compared to the approach here, map building is more attractive and flexible because the acquired information can be used as a map to be read by the human or as a map that the robot can use for localisation and navigation in the known environment. However, as discussed in Section 5.1.2, to build an accurate map for navigation, the robot sensing location (i.e. position and orientation with respect to the robot working environment) must be known precisely, which is difficult to achieve. In contrast, the approach described here does not require accurate robot sensing location as it only requires absolute direction information for navigation. A disadvantage of this approach is that it requires the human to correlate the identified landmarks with the environment so as to command the robot back to desire explored location. Although the current implementation allows the human to assign name to the identified landmark during the learning process, it will become too extensive for the human to correlate when

there are too many landmarks generated. Therefore, to use this approach the human must be aware of landmarks generated by the robot. If he cannot keep track or remember, the learned landmarks will not be useful.

6.2.3 Vision Based Tasks

One widely adopted approach to perform goal-oriented tasks such as object following, intrusion detection or autonomous surveillance is through the use of machine vision (Jain et al. [163]). To perform such tasks, the behaviours of motion detection and object following are incorporated into the telerobotics system. Robust detection of moving objects by a mobile robot is not easily achievable since there are two independent motions involved: the motions of moving objects in the environment and the motion of the robot (Jung & Gaurav [164]). As the current motion detection behaviour is implemented using simple image differencing³⁰ technique [163], it cannot be used to discriminate the two motions mentioned above. Due to this limitation, this behavior is only employed in static operation, such as performing sentry task to prevent the passage of unauthorised persons. This is further discussed in sub-section, robot sentry.

The object following behaviour implemented uses a combination of colour segmentation³¹ and edge detection³². These are selected due to the simplicity and suitability for real-time implementation. The colour segmentation and edge detection are implemented using Colour Machine Vision (CMVision) and Open Source Computer Vision (OpenCV) libraries respectively (see Appendix E). The implemented tracking strategy first uses color segmentation for defining object areas followed by defining the exact edges of the object using edge detection (Marr [165]). Through this combination, the robot can track object with different colours and shapes more efficiently as compared to using either colour segmentation or edge detection alone.

With the addition of these two behaviours, two experiments, namely sentry and robot following were conducted to demonstrate their working principles and also the types of human assistance provided to the robot in performing such tasks. Both experiments are

³⁰ Image differencing is performed by differencing two consecutive image frames.

³¹ Color segmentation identified object based on specific colour using the RGB (Red, Green, Blue), HIS (Hue, Saturation, Intensity) or YUV colour spaces. The Y component is the luminance part of the signal, and U and V represent the chrominance components.

conducted using the exclusive traded mode. The exclusive traded mode coordination architecture for performing vision based task is similar to the exclusive traded mode coordination presented in Figure 6.10. The difference is that the goal seeking behaviour is replaced by the motion detection and object following behaviours. In this case, the human input to the robot is an abstract command such as “follow that object” and object specification (i.e. shape and colour).

Robot Sentry

In this experiment, the robot was commanded to perform a sentry task, monitoring the front entrance of RRC. If the robot finds significant pixel changes between two consecutive images, it will alert the human and trigger the object following behaviour to follow the object (in this case, a person). If the robot does not have the specification of object to be followed in its database, the robot will request for human assistance. The human assistance in this case is to train the robot to recognise the object on the spot. On the other hand, if the robot has the specification of the object in its database, it will proceed to follow the object until the human commands the robot to stop. During this cycle, if the robot lost track of the object, the robot will request for human assistance to refine the tracking object. The state flow for describing this task execution is depicted in Figure 6.23. The snapshots of the robot following a people (in orange) are depicted in Figure 6.24.

Figure 6.23: Sentry task state flow

³² Edge detection uses relative contrast in nearby pixels to determine boundaries of an object.

Figure 6.24: Snapshots of people following

Robot Following

In this experiment, instead of following a person, the robot was commanded to follow a mobile robot (another ATRV-Jr™), teleoperated by a human. To facilitate this, a square orange board was placed on the teleoperated robot as a marker for the following

robot to track. An advantage of this configuration is that it allows a human to control multiple robots. Snapshots of this task are depicted in Figure 6.25.

Figure 6.25: Snapshots of the robot following

6.2.4 Human-Assisted Localisation

As discussed in Section 2.2.3, a problem in task trading is the robot may not be able to keep up with the state of the world when the human takes control from the robot and performs the task manually. This happens when the human performs one part of the navigation task under teleoperation (either via manual or exclusive shared mode) and let the robot executes the other part of the task semi-autonomously or autonomously (either via exclusive traded or autonomous mode). Under teleoperation, the human uses continuous rate control to drive the robot. During this cycle, if the robot does not keep track of its location, the robot may not be able to execute the task properly when the human changes to waypoint control. To illustrate, Figure 6.26 shows two robot trajectories obtained from the experiments conducted in Section 6.1 via the exclusive shared mode. One case is that in which the robot continuously keeps track of its location during teleoperation using the localisation technique described in Section 5.1.3 and the other case without localisation. It is obvious in Figure 6.26 that the case without localisation could not keep up with the state of the world.

Figure 6.26: Actual x-y plot of the robot trajectories overlaid on the schematic layout of the test course depicted in Figure 6.5

To keep the robot's location aligned with the world, one approach in the current implementation of the telerobotics system is to let the human assists the robot in localising its own location. This approach is described in Section 5.1.3. To demonstrate this approach, an experiment was conducted. This experiment assumed that the human already knew the "robot was lost" and hence needed his assistance. Different strategies

to this situation are discussed in Section 5.1.3. This human-assisted localisation experiment is depicted in Figure 6.27 and proceeds as follows:

Figure 6.27: Snapshots of Human-assisted localisation

1. As shown in Figure 6.27(a), the human only needs to teleoperate the robot “roughly” to the designated location for localisation. An orange piece of paper is mounted at this location as the vision target to attract the robot. Once the robot recognises the orange target, it then notifies the human. The human then trades the control over to the robot to let the robot performs the localisation autonomously.
2. As shown in Figure 6.27(b), once the control is traded over to the robot, it moves towards the orange target. Once the distance as detected by the laser range finder is approximately within 0.4m, the robot stops its forward motion and steers (i.e. turn-

on-spot) to align with the target. The steering direction is based on the column position of the image. In this case, the steering angle is 26 pixels towards the right. The robot's compass is used to determine when the correct angle of rotation is reached, which is 270° with respect to the true North. However, due to the electromagnetic interferences from the environment, the actual reading is $287^\circ \pm 0.5^\circ$ with respect to the true North. As the reading from the compass cannot be measured accurately, a side guide is used to ensure that the robot can position itself perpendicular to the orange target based on the reading from the laser range finder.

3. As shown in Figure 6.27(c), once the robot aligns itself perpendicular to both the front and left objects within the preset distance of the laser range finder, the localisation process stops. The robot hands the control back to the human to reset /reinitialise the localisation system.

As simple as this task may seem, without human assistance of driving the robot to the localisation location, the robot may take a much longer time to find the localisation location or the robot may not even find it. This is because under such circumstances, the robot can only wander around the working environment to look for the designated localisation location.

6.3 Discussion on Seamless Human-Robot Interaction

To assess how seamless HRI can be achieved based on the concept of $T_{S\&T}$, the assessment is divided into two main parts. The first part presented in Section 6.1 assessed seamless HRI due to local $T_{S\&T}$ (i.e. change of human-robot roles with same task specification). The second part presented in Section 6.2 assessed seamless HRI due to global $T_{S\&T}$ (i.e. change of human-robot roles with completely different type of task specification). The concepts of local and global $T_{S\&T}$ are described in Section 3.2.5. Figure 6.28 provides an overall view of how the concept of seamless HRI was assessed in Section 6.1 and 6.2.

Figure 6.28: An overall view on the assessment of seamless HRI in Chapter 6

In assessing the concept of seamless HRI due to local $T_{S\&T}$ in Section 6.1, the results obtained for validating Hypothesis 2 (Section 6.1.1) shows that even though the human-participants were given the flexibility to change their interaction roles to attain the required degree of robot assistance they desired, the participants needed to know when to perform roles transition. For example, when making a U-turn, it was better for the participants to switch to the exclusive shared mode so as to enable the maximum robot assistance. On the other hand, the robot must constantly monitor the participant control behaviours to adapt its autonomy in response to their control changes. This implies that to achieve seamless HRI both issues discussed above must be met. The results obtained from the user's evaluation show that this can be met if the participants were given the appropriate feedback for perceiving when roles transition should be performed. For this evaluation, real-time video feedback (from the robot on-board camera) was provided to the participants to let them perform roles transition seamlessly. Real-time video feedback was required because this evaluation requires the participants to teleoperate the robot remotely (without line of sight) at a fast speed. In the context of the robot responding to the each participant's control, the results obtained show that the robot can

response to the control changes seamlessly by adjusting its autonomy (so as to attain the required degree of assistance needed by each participant) with an insignificant number of stops (an average of 2.07 stops for two trials, see Table 6.2 - *adaptive interaction modes*) during teleoperation. The stops were not caused by the robot. It was due to the uncertainty the participants have during roles transition.

In assessing the concept of seamless HRI due to global $T_{S\&T}$ in Section 6.2, two different types of experiments were conducted, namely *landmarks based navigation* (Section 6.2.2) and *human-assisted localisation* (Section 6.2.4). The experiment on landmarks based navigation illustrates the human-robot roles transition from teacher-learner (for learning of landmarks) to supervisor-subordinate (for task execution using the learned landmarks). In this context, if the robot does not have the learning capability to response to the human teaching, it is not possible to achieve seamless HRI. This is because the robot does not have the necessary learned landmark(s) for executing the navigation task. On the other hand, the experiment on human-assisted localisation illustrates the human-robot roles transition from partner-partner (for teleoperating the robot near to the localised location) to supervisor-subordinate (in which the robot performs the localisation autonomously). In addition, this experiment also illustrates an important issue that pertains to the achievement of seamless HRI, i.e., the robot must maintain a world model so as to respond to human control delegation and roles transition. To maintain a world model, the robot must constantly keep track of its own location with respect to the working environment. In this context, to achieve seamless HRI, the robot not only requires high computation for localisation (see Section 5.1.3) but also requires parallel processing for the execution of human control inputs, feedback task information to the human and maintaining the world model at the same time.

6.4 Chapter Summary

The experimental evaluation presented here for investigating the concept of $T_{S\&T}$ is divided into two main parts, namely *robot assists human* (RAH) and *human assists robot* (HAR) presented in Section 6.1 and Section 6.2 respectively. An overall view of the experimental evaluation presented in this chapter is depicted in Figure 6.1. The experimental evaluation shows that the telerobotics system implemented provided a sufficient level of robustness, utility and usability necessary for conducting the

experiments in complex environment.

The purpose of the experimental evaluation on RAH was to assess the cooperation between human and robot based on the concept of task sharing. In addition, another aim of this evaluation was to assess the performance efficiency of using different interaction roles and flexibility in roles changing during task execution. This is for illustrating the concept of seamless HRI due to change of human-robot roles with same task specification. The task of the experiment was remote navigation. Obtaining objective data for this experimental evaluation is complex. This is because, to support a useful users' evaluation, a user interface that can provide effective HRI must be provided. In addition, the robot must have sufficient level of autonomy to perform the desired task and interacts with the human-participants.

In this experiment, four performance measures, namely *task completion time*, *number of collisions*, *number of stops* and *number of turn-on-spots* were employed for assessing the three modes of control, i.e. *manual mode*, *exclusive shared mode* and *adaptive interaction modes*. The experiment was carried out based on Latin square design with two replications and the performance measures were analysed via the ANOVA technique. The ANOVA results obtained shows that the system performance is improved significantly when the robot provided appropriate assistance to the human as compared to no assistance was provided. On the other hand, the ANOVA results also show that a better performance can be achieved if human and robot can change their interaction roles dynamically during task execution as compared to the use of fixed interaction role (i.e. manual and exclusive shared modes).

The purpose of the experimental evaluation on HAR was to assess the cooperation between human and robot based on the concept of task trading. Another aim of this evaluation was to illustrate three other human-robot roles, namely *Supervisor-Subordinate*, *Teacher-Learner* and *Fully Autonomous* and the concept of seamless HRI due to change of human-robot roles with completely different type of task specification. The evaluation approach adopted assessed how human assistance was rendered to the robot using practical mobile robotics problems. The experimental studies include *waypoints navigation*, *landmarks based navigation*, *vision based tasks* and *human-assisted localisation*. The human assistance provided to the robot in these experimental

studies range from abstract goal specifications (e.g. navigation direction) and guidance to more demanding assistance such as teaching, correlation of visual scene, path planning, decision making and teleoperation. A summary of how the concepts of task trading (in the context of control trading from the human to the robot and autonomy trading from the robot to the human) and seamless HRI were demonstrated in this experimental evaluation on HAR is presented in Table 6.4.

The four experimental studies on HAR have shown that to allow human assists robot in different situations, the robot must be able to accept a full spectrum of intervention possibilities. This implies that the task trading between human and robot cannot be handled merely as augmentations to a control system; instead, opportunities for human intervention must be incorporated as an integral part of the robot's built-in intelligence. The robot must be imbued with the capability to accept different levels and frequencies of intervention. Finally, for autonomous capabilities to evolve, the robot must be able to recognise when assistance is needed from the human.

Chapter 7

Discussion and Conclusion

Achieving seamless human-robot interaction (HRI) based on the concept of *sharing and trading* in a human-robot system (HRS) has been the main focus of this dissertation. The type of HRS addressed in this thesis is a *telerobotics system*. The solution offered is, via appropriate degree of sharing and trading where human and robot can interact more effectively to ensure high-level of system performance and the satisfaction of task demands. The aim of this final chapter is to provide a road map of the thesis through the discussion of the essential points, summarisation of the contributions of this research and conclude with proposed future work.

7.1 Discussion

7.1.1 Achieving Seamless Human-Robot Interaction

In this thesis, the main concern for the design and development of a cooperative HRS is the achievement of seamless HRI. Seamless HRI implies flexibility in human control through which a human interacts with a robot in different situations, and the adaptability of robot's autonomy in response to human control. In order to give human the flexibility in controlling robot, the approach by this research is to allow for human to vary his/her degree of control involvement with robot, and to delegate control at different levels of task details to the robot during task execution. This includes allowing for human control intervention at any time. To facilitate, robot capabilities were modelled to encompass basic task-oriented behaviours for performing the delegated HRS tasks and intelligent behaviours for monitoring of human control behaviours. This capability allows for the robot to adapt its degree of autonomy in response to human control signal changes. It also allows for robot to adjust its autonomy level dynamically in order to accept a full spectrum of human control delegation.

The approach for smooth coordination of human control and robot autonomy changes (i.e. changing responsibilities) above is via a flexible communication framework. It supports both the actions of requesting and accepting a change in responsibilities invoked by both human and robot. This is to allow for human and robot to coordinate

their actions and to reduce the potential for anomalies that would lead to unexpected responses from the robot or inappropriate control responses by the human. This encompasses the facilitation of finding out what the other party is doing, its intention and the resolution of any conflicts as it arises during task execution.

Concept of Sharing and Trading

By specifying the achievement of seamless HRI in the manner above, leads to the identification of three key elements involved in the process of HRI. They were *human control*, *robot autonomy* and *information*. To describe how these elements are used in the design and development of a cooperative HRS for seamless HRI, a comprehensive formalism (i.e. a conceptual framework) is needed. In this research, the main approach for the development of this conceptual framework is based on the concept of sharing and trading. A human-robot cooperation concept, which allows human and robot to work as a team by letting them contribute according to their degree/level of expertise in different task situations and task demands.

7.1.2 The Need for a Framework of Sharing and Trading for Seamless Human-Robot Interaction

To determine if there was an existing framework of sharing and trading in robotics that could assist in the design and development of an HRS for seamless HRI, existing works from the domains of telemanipulation and teleoperation of mobile robots were reviewed. The review is based on the interaction strategies (i.e. how human and robot may work together) and requirements which were derived from the human-robot roles and relationships in different types of HRS. In this research, five human-robot roles and relationships, namely *Master-Slave*, *Supervisor-Subordinate*, *Partner-Partner*, *Teacher-Learner* and *Fully Autonomous mode by the robot* were identified.

The review shows that though there were many works applying the concept of sharing and/or trading as a basis for HRI, these applications of sharing and trading is normally applied on an ad hoc basis, without a comprehensive formalism to support the general design and development of an HRS. In particular, there is a lack of an agenda to synthesise the key elements (i.e. human control, robot autonomy and information) in the process of sharing and trading between human and robot (or HRI) in a holistic manner,

let alone a framework of sharing and trading for seamless HRI. The review suggests that there was no framework of sharing and trading to assist in the design and development of a cooperative HRS for seamless HRI in a holistic manner (Ong et al. [166]).

7.1.3 Development of the Framework of Sharing and Trading

The development of the framework of sharing and trading is divided into two stages. The first stage as presented in Chapter 2 involves the identification of the key elements and their associated features and attributes for defining what can be shared and traded between human and robot in an HRS. This provides the basic construct towards the development of the framework of sharing and trading. The key elements consist of human control, robot autonomy and information involved in the process of sharing and trading between human and robot. These elements were identified based on: (1) the command and control of robot from the perspective of human interacting with the robot; (2) the required degree of robot autonomy from the perspective of robot interacting with the human; and (3) issues concerning the communication between human and robot.

The second stage as presented in Chapter 3 is the formalisation of the concept of sharing and trading. Here, the concept of sharing and trading that was formulated is in the context of a task, denoted as *Task Sharing and Trading* ($T_{S\&T}$). The term *task* is related to the required human's and robot's functions and the goals they are attempting to accomplish. Starting from the concept of task allocation, the concept of $T_{S\&T}$ is introduced, defined and formulated. Instead of the traditional mandatory task allocation between human and robot based solely on synergistic matching of human capabilities and limitations with respect to robot capabilities and limitations, $T_{S\&T}$ provides a complementary view by looking into the possibilities of letting *robot assists human* (RAH) and *human assists robot* (HAR). The basic idea of this paradigm is to allow human and robot to work as a team, where each can actively take initiatives (via different interaction roles) to accomplish task objectives by assisting each other in different ways to fit situational needs and change of capabilities throughout a task. This serves as the effective basis in making timely and pragmatic task allocation decisions for resolving conflicts/problems between human and robot by providing insight into the design of different human-robot cooperation strategies based on mutual assistance.

A Paradigm of Robot Assists Human-Human Assists Robot (RAH-HAR)

When a human team interacts to perform a common task, a tight coordination of their actions is needed to achieve seamless cooperation. They cooperate by coordinating their actions through monitoring of each other's actions and sharing of information pertaining to the task process. These include task procedures, task assignment, task status, coordination protocol and so forth. Each team member monitors one another's action to be aware of the present state of the current task execution, and which action is necessary. Throughout the task execution, their interaction roles may change depending on individual's performances and in accordance to different situations. Appropriate actions may be provided to assist one another in correcting each other errors/mistakes, or at least alert the other member's of his/her mistakes.

In this research, the primary motivation of adopting the RAH-HAR paradigm in the sharing and trading framework is to achieve such a human-human coordination described above for seamless HRI. The guideline towards the development of this paradigm is the "Un-Fitts List" by Woods [101], which emphasises on how the competencies of human and robot can be enhanced through appropriate forms of cooperative interaction. The "Un-Fitts List" is suitable to envisage how human and robot assist each other because it does not devalue the human in order to justify the robot. It also does not criticise the robot in order to rationalise the human, but provides a complement view of exploiting the best strengths from both of them. Through this perspective, the continuum of $T_{S\&T}$ for developing a full spectrum of interaction modes ranges from "no assistance provided to the human by the robot" to "no assistance provided to the robot by the human". Within the extreme of these two interaction modes, different types of cooperation strategies between human and robot can be envisaged based on how human and robot assist each other. The results obtained from the experiments (see Section 7.1.5) show that the RAH-HAR paradigm provides a useful means of letting human and robot compensate for the unique kind of failures and limitations possessed by each other during task execution.

Different Human-Robot Roles and Roles Transition

The main corollary of the concept of $T_{S\&T}$ based on the paradigm of RAH-HAR is that human and robot need to engage in different human-robot roles to work as a team.

As a consequence, it requires the flexibility in roles transition to allow for human and robot to deal with different aspects of task situations and task demands. Here, the human-robot roles adopted for the development of the sharing and trading framework is based on the existing work of sharing and trading in robotics (see Section 7.1.2). As each human-robot roles concentrate on different aspects of $T_{S\&T}$ (or HRI), a characterisation was provided to define how human and robot might share and trade in each role. The characterisation was substantiated by illustrating with typical scenarios from different robotics applications. It was determined that to achieve seamless HRI based on the concept of $T_{S\&T}$, the framework developed must address both human-robot roles transition with same task specification (i.e. local $T_{S\&T}$) and also roles transition with completely different types of task specifications (i.e. global $T_{S\&T}$).

Formalism for $T_{S\&T}$

To formalise $T_{S\&T}$ in a holistic manner, basic task activities (e.g. desired input tasks, human tasks, robot tasks, etc.) within an HRS were defined to explain how human and robot work together. These activities were defined based on the task allocation between human and robot, task sharing and trading between them, and task reallocation. The formalism served as the basis to look into the dynamics of the $T_{S\&T}$ process for addressing the contingencies that arise when human and robot work together during task execution. As a consequence, this facilitates the achievement of seamless HRI.

Framework Formulation

To be effective and useful, the formulated framework of sharing and trading for seamless HRI was designed around general cases to accommodate different HRS configurations. To facilitate, a list of questions was considered:

1. Why should human and robot share and trade?
2. When should human and robot share and trade?
3. How does human and robot know when to share and trade?
4. How does human and robot share and trade?
5. What triggers the change from sharing to trading (or trading to sharing)?

6. Who is in charge of the sharing and trading process?

The first and second questions addressed the timeliness and pragmatic task allocation issues between human and robot raised by the concept of $T_{S\&T}$ for resolving arising problems during task execution. The solution provided is based on the invocation of specific task events as “conditions” to invoke the $T_{S\&T}$ process between human and robot. They are: (i) goal deviations, where the $T_{S\&T}$ process would be invoked by human intervention. This is to allow for human assistance to robot; (ii) evolving situation, in which the $T_{S\&T}$ process would be invoked by the robot to veto human commands. This is to allow for robot assistance to human; and (iii) events when both human and robot *explicitly request assistance* from each other.

The third and fourth questions addressed how the different types of cooperation strategies invoked by the concept of $T_{S\&T}$ based on RAH-HAR capabilities can be realised. The solution offers is to provide a range of flexible task interaction modes for human and robot to cooperate in different circumstances. These task interaction modes are designed in accordance to the characterisation of $T_{S\&T}$ in different human-robot roles. The four main task interaction modes proposed are *manual mode* (i.e. no assistance provide to human by robot), *exclusive shared mode* (i.e. for letting robot provides appropriate assistance to human), *exclusive traded mode* (i.e. for letting human provides appropriate assistance to robot) and *autonomous mode* (i.e. no assistance provided to robot by human). Within exclusive shared and traded modes, a range of sub-modes can be incorporated with varying degree of human control/robot autonomy for providing a finer grain of $T_{S\&T}$ (i.e. for varying the degree of human/robot assistance) as needed for particular applications. These four interaction modes have been implemented in a telerobotics system. Experiments have been conducted to show their working principals and the concept of seamless HRI due to the transition of these task interaction modes.

The fifth question addressed the types of triggers involved in the $T_{S\&T}$ process for achieving seamless HRI. They are distinguished as *mandatory* triggers for different level of tasks changes (i.e. seamless HRI under global $T_{S\&T}$) and *provisional* triggers for varying the degree of HRI within the context of a same task (i.e. seamless HRI under local $T_{S\&T}$). Finally, question six addressed the human-robot authority issues. To let robot performs a task autonomously or provide assistance to human implies that robot

may be in authority to lead certain aspect of the tasks. To ensure that human retains the overall responsibilities of the outcome of the task under such situations, the approach is to give the human the flexibility to intervene the robot operation at any circumstances.

Requirements for the Development a Shared and Traded HRS

By examining the answers to the questions above and categorising their respective contents, four essential requirements for the development of a shared and traded HRS were identified to allow for human and robot to share and trade effectively. Firstly, a shared representation between the human and the robot is required to facilitate effective communicate between them. Secondly, both the human and the robot need to monitor each other's actions and states so as to develop and update a model of each other. Third, to alter and negotiate their interaction strategies in accordance to each other's task or performance of each other, both human and robot need to learn from the interaction. Fourth, the system must be able to resolve any arising conflicts flexibly and dynamically so as to operate efficiently in response to the $T_{S\&T}$ processes and changing situations. These requirements serve as the basis for supporting the application of the framework of sharing and trading in the modelling of a telerobotics system in Chapter 4.

7.1.4 Shared and Traded Architecture

In Chapter 4, to show how the concept of $T_{S\&T}$ can be applied, it is employed in modelling of a telerobotics system, which encompasses the implementation of this system in Chapter 5 using a physical mobile robot. The overall telerobotics system architecture is complex and composed of five major subsystems/components: robot hardware (sensors and actuators), navigation, localisation, planning (including learning) and interfaces. The implemented telerobotics system is highly flexible and supports a full spectrum of interaction modes (from manual to autonomous mode) for human and robot to work together at different levels of task details. For global coordination (i.e. global $T_{S\&T}$ for different levels of tasks delegation), the implemented shared and traded architecture uses priority-based arbitration for coordinating different interaction roles. As each interaction role is designed to perform specific telerobotics task, the responsibility of choosing appropriate interaction roles for performing a particular task is determined by the human. This is because in performing a particular HRS task may

require attributes such as prior knowledge of a task, “common sense” in reasoning and perception to fulfill the overall mission. Such attributes are possessed by human but not by the robot. Therefore, human is in a better position than the robot to decide when to perform roles transition in global coordination. On the other hand, for local coordination (i.e. local $T_{S\&T}$ for varying the degree of human control and robot autonomy) within each interaction roles, the shared and traded architecture employs a hybrid approach based on priority-based arbitration and superposition-based command fusion for resolving conflicts and actions. Under this coordination scheme, both the human and robot can actively influence the coordination process in real-time.

Implicit and Explicit Communication

To facilitate the coordination of human and robot actions, both implicit and explicit communications are employed. If the states (i.e. actions, intentions and goals) of the human were known, the HRI problems would be for the robot to coordinate and adapt its autonomy to ensure satisfactory performance. However, if the states of the human are not known, the robot must logically infer the states of the human so as to coordinate and adapt its autonomy to work with the human. The implicit communication is for such situation. The purpose is to enable the robot to provide appropriate assistance to the human during task sharing. On the other hand, the purpose of explicit communication is to allow the human to provide appropriate assistance to robot (e.g. via a menu-based interface or any other explicit means) or for robot to request help from the human (e.g. by means of dialog or audio) during task trading.

7.1.5 Proof-of-Concept Experimental Evaluation

Using the implemented telerobotics system, two sets of experiments were conducted in Chapter 6 to evaluate the concepts of $T_{S\&T}$ and seamless HRI.

The first set of experiments evaluates RAH. This was required to assess the cooperation between human and robot based on the concept of task sharing. In addition, another aim of this evaluation was to assess the need of using different interaction roles (i.e. master-slave and partner-partner) and flexibility in roles changing during task execution. This is for assessing the concept of seamless HRI due to change of human-

robot roles with the same task specification. The results obtained from the experimental evaluation shows that the system performance was improved (see Section 6.1.4, Table 6.2) when the robot was allowed to provide appropriate assistance to the human. This result is supported by the work of Jacob and Goodrich [167] and Krotkov et al. [58] that compared robot-assisted teleoperation with traditional direct manual teleoperation. The results reported in both [167] and [58] shown that human users performed better (i.e. users drive faster) with robot assistance rather than without robot assistance. However, as compared to the work by Jacob and Goodrich, and Krotkov et al., the work presented here not only evaluated how RAH, but it also included the evaluation of seamless HRI by assessing whether giving human the flexibility of deciding when he/she requires robot assistance can have better performance over fixed interaction strategy. The results obtained show that a better performance can be achieved if this flexibility is given. Such objective evaluation for assessing seamless HRI has been found to be lacking in the current literature of robotics and HRI.

The second set of experiments evaluates HAR. The purpose is to assess the cooperation between human and robot based on the concept of task trading and, the concept of seamless HRI due to change of human-robot roles with completely different type of task specification. In contrast to the first set of experiments that uses a comparative study, the experiments conducted here were in the form of demonstrative examples to illustrate how human assistance was provided to the robot in a seamless way using practical telerobotics tasks. The results obtained show that:

- ❑ Human assistance is needed not only in unforeseen or novel situations that hinders the robot from performing a task or to ensure safe operation, but also in situations when human notices an opportunity to improve the robot task performance. This shows the importance and advantage of allowing human to provide appropriate assistance to robot in an HRS. Thus, there is a need to provide different types of interaction strategies for the human to assist or guide the robot in different aspects of task situations.
- ❑ To allow HAR in a seamless way, human must be given the flexibility to intervene at any instance even when the robot is competent to perform the task. Through this, if any failure occurs, human can retain the overall responsibilities of the outcome of the

tasks undertaken by the robot.

- ❑ To achieve seamless HRI, the robot must imbue with the required capability to response to the human assistance and as well as to recognise when human assistance is needed.
- ❑ The robot must maintain a world model so as to response to human control delegation and roles transition during task execution. The result obtained shows that if robot does not maintain the world model, seamless HRI cannot be achieved when robot performs task trading with human.

7.2 Contributions

This research work contributes towards the knowledge of how human and robot interacts in a human-robot configuration system by providing a holistic basis through the formalising of a framework of sharing and trading for seamless HRI. This research is important as increasing number of robots are being engaged by humans for various tasks. This implies that the opportunities of humans working with robots are increasing greatly in our current economy. Hence, this research need of formalising a sharing and trading framework for seamless HRI contributes to the understanding of how human and robot can interact more effectively.

To summarise, the contributions are:

- ❑ Development of a conceptual framework of sharing and trading to assist in the design and development of a cooperative HRS for seamless HRI. This leads to:
 - the identification of the different human-robot roles and relationships in the current literature of robotics and, as a consequence the survey of different innovative interaction strategies and interaction requirements invoked by each role for effective HRI.
 - the formalisation for sharing and trading in terms of human control, robot autonomy and information exchange in different human-robot roles. They are Master-Slave, Supervisor-Subordinate, Partner-Partner, Teacher-Learner and Fully Autonomous mode by the robot. In addition, the formalisation encompasses a comprehensive description on how to integrate the different types

- of human-robot roles under the same framework of sharing and trading and the achievement of seamless HRI due to roles transition. This serves as the basis for the integration of human control and robot autonomy at system-level.
- the invocation of a human-robot cooperation paradigm based on how robot assists human (RAH) and how human assists robot (HAR), which serves as the basis to assist in making prior timeliness and pragmatic task allocation decisions. This addresses the role of sharing and trading in allocating tasks between human and robot in the initial design stage of HRS. Such a formulation to address the role of sharing and trading in task allocation is lacking from literature of robotics adopting the concept of sharing and trading for human-robot cooperation.
- Exemplification of how the framework of sharing and trading is applied in the implementation of a telerobotics system. This included:
- the implementation of a robust and flexible telerobotics system for operating in dynamic and unpredictable environment, which serves as the research platform for conducting proof-of-concept experiments.
 - the implementation of a hybrid system architecture that shows how sharing and trading are applied in conjunction explicitly. As a consequence, the implementation of a highly flexible communication protocol that allows human and robot to coordinate their actions both implicitly (for task sharing) and explicitly (for task trading) in accordance to different task situations, task demands and their needs.
- Provided proof-of-concept experiments to show:
- that the development of a cooperative HRS not only facilitates the requirements of letting robot assists human, it must also allow for human to provide appropriate assistance to the robot, particularly in unforeseen circumstances. This shows the usefulness of the RAH-HAR paradigm invoked by the concept of sharing and trading for human-robot cooperation.
 - how the concept of seamless HRI due to both human-robot roles transition with same task specification and also roles transition with completely different type of task specifications are achieved.

7.3 Future Work

7.3.1 *Unaddressed Issues*

Although this thesis has established a comprehensive study on the design and development of a cooperative HRS based on the achievement of seamless HRI, there are other related issues that are not addressed. First, this research has identified and shown that to be an active assistant of human, a robot needs to perceive and be aware of human control behaviours (i.e. actions and intentions). However, an issue this work does not address is to what extent does the robot needs to model the human's actions and intentions so as to become an active team member of human. Further study is required to determine this issue.

Second, the approach adopted for addressing the human-robot authority issues is that it requires that neither human nor robot to be exclusively in charge of HRS task, but rather it requires human to retain as the overall responsibilities of the outcome of the tasks undertaken by the robot and retains the final authority corresponding with that responsibility. This is achieved by giving human both the flexibility in delegating control to the robot at different levels and varying his/her control involvement with the robot at varying degree of task details. However, to develop a truly cooperative HRS, it may not be the best approach, particularly in situation when human does not have the capability to delegate control to the robot or vary his/her control involvement with the robot due to insufficient/poor task feedback. If this happen, the system may be disabled and, thus affect the achievement of seamless HRI. A possible solution is to allow both human and robot to have "equal" authority to correspond to the overall responsibilities of the outcome of tasks together. This implies not just giving the robot the authority to leads certain aspect of a task, but also responsible for the task success or failure of the task. If this can be modelled, a more "synergistic" approach can be provided for addressing the human-robot authority issues. Thus, an extension of this research is to investigate this issue and how it can be incorporated into the current framework.

Third, to envision a "tighter" cooperation between human and robot just as in human-human teamwork requires not only understanding (or modelling) of each other actions and intentions, but may also depend on human's "trust" in interacting with robot. As

operations in an HRS can be complex, humans may fail to understand the mechanism of the operations. For instance, robot may operate abnormally under certain conditions. Therefore, the level of trust human has in an HRS is important. If the human trusts the system too much, he/she may become complacent about the behaviour of the robot and may not be vigilant about understanding its effects (this may lead to serious accidents). On the other hand, if the human distrusts the system, the system can be disabled. According to Barber [168], differing degree of trust impacts all interactions involving people and technology. Trust has been studied in a number of domains. For example in automation, Lee and Moray [169] and Abe et al. [170] showed that human's trust in automation is one of the major factors in the usage of automation. Following this, to make sharing and trading effective, the issue of human trusting the robot is important. Logically, trust is not acquired instantaneously; it must be built up gradually. For example, it is based on the experience of controlling the robot to know how it acts. If the action of the robot is predictable, the level of trust of the human will be high. However, if the robot acts abnormally, the level of trust of the human will be low. Although trust is an important issue to consider, it is not possible to include in this work due to the extensive evaluation and study required to model the underlying characteristics of human behaviours [169, 170]. Thus, this area requires a separate research study and it is beyond the scope of this thesis.

7.3.2 *Experimental Evaluation*

The evaluation approach adopted by this research uses the most concrete way to access how robot assists human (RAH); i.e. based on *direct comparison* of different task interaction modes. Although such an approach can provide quantitative evidence, this evidence by itself may not provide a sufficient basis to assess the full “characteristics” of the usefulness of letting RAH. This is because this approach only focuses on assessing the HRS task performance. The limitation of this approach is it cannot be used to assess *human performance* such as workload³³ which includes mental demands, physical demands, temporal demands, and effort in performing a task, to name a few. As the approach adopted here only indicates whether with robot assistance, the human can

³³ It is defined as the effort expended by a human in accomplishing the imposed task demand. Demand is determined by the goal that has to be attained by means of task performance, and, once the goal is set; external and independent of the individual (Waard de [171]).

perform better or not, it does not provide a greater understanding of which aspect of human performance is improved when appropriate robot assistance is given. To assess this, requires a “multi-dimensional assessing approach” that can derive an overall quantitative human workload measurement. However, to obtain the abovementioned workload measurements, the system must be equipped with a workload measurement technology that is capable of detecting changes in different aspects of workload levels, which is beyond the scope of this research work. Nevertheless, this research has established a robust HRS that can be used as a HRI testbed to conduct such experiments. If the HRS is able to assess the human workload objectively, this research can be extended to assess how human *learns* from robot when they share and trade, in particular how robot assistance is provided, which was identified as one of the essential requirements for achieving seamless HRI.

References: Chapter 1

- [1] K. W. Ong, G. Seet, and S. K. Sim. A Hierarchical Mixed Multi-Agent System, *Proceedings of the 4th IEEE International Conference on Control and Automation*, pp. 644-648, Montreal, Canada, June 10-12, 2003.
- [2] T. B. Sheridan. *Telerobotics, Automation, and Human Supervisory Control*, MIT Press, ISBN 0262193167, Cambridge, 1992.
- [3] L. Pedersen, D. Kortenkamp, D. Wettergreen and I. Nourbakhsh. A Survey of Space Robotics, *Proceedings of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, Nara, Japan, May 19-23, 2003.
- [4] G. Hirzinger. Multisensory Shared Autonomy and Tele-Sensor Programming – Key Issues in Space Robotics, *Robotics and Autonomous Systems*, Vol. 11, pp. 141-162, ISSN 0921-8890, 1993.
- [5] P. A. Hancock. On the Future of Hybrid Human-Machine Systems. In: *Verification and Validation of Complex Systems: Human Factors Issues*, J. A. Wise, V. D. Hopkin and P. Stager (Eds.), NATO ASI Series F, Vol. 110, pp. 61-85, Springer-Verlag, ISBN 3540565744, Berlin, Germany, 1992.
- [6] W. R. Hamel and M. J. Feldman. The Advancement of Remote Technology: Past Perspectives and Future plans, *Proceedings of the National Topical Meeting on Robotics and Remote Handling in Hostile Environments*, American Nuclear Society, Gatlinburg, TN, USA, 1984.
- [7] J. Casper and R. R. Murphy. Human-Robot Interactions during the Robot-Assisted Urban Search and Rescue Response at the World Trade Center, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, Vol. 33, No. 3, pp. 367-385, ISSN 1083-4419, June, 2003.
- [8] W. D. Gage. UGV History 101: A Brief History of Unmanned Ground Vehicle (UGV) development efforts, *Unmanned Systems Magazine* Vol. 13, No. 3, pp 9-16, 1995.
- [9] W. D. Gage. Current Thrusts in Ground Robotics: Programs, Systems, Technologies, Issues, *Proceedings of the 4th International Symposium on Technology and Mine Problem*, Monterey, CA, USA, March 12-16, 2000.
- [10] Q. Zhang, J. F. Reid, and N. Noguchi. Agricultural Vehicle Navigation using Multiple Guidance Sensors. *Proceedings of the International Conference on Field and Service Robotics*, Pittsburg, PA, USA, UILU-ENG-99-7013, August 29-31, 1999.
- [11] D. Carroll, G. A. Gilbreath, and H. R. Everett. Extending Mobile Security Robots to Force Protection Missions. *AUVSI Unmanned Systems*, Lake Buena Vista, FL, July 9-11, 2002.
- [12] L. Cousineau and M. Nobuyasu. *Construction Robotics: The Search for New Building Technology in Japan*, ASCE Press, ISBN 0784403171, Reston, VA, 1998.
- [13] G. Giralt, R. Chatila, and R. Alami. Remote Intervention, Robot Autonomy, and Teleprogramming: Generic Concepts and Real World Application Cases,

- Proceedings of the IEEE/RSJ International Conference on Intelligence on Intelligent Robots and Systems*, pp. 27-34, ISBN 0780308247, Yokohama, Japan, July 26-30, 1993.
- [14] F. Amigoni, V. Schiaffonati, and M. Somalvico. Processing and Interaction in Robotics, *Sensors and Actuators A: Physical*, Vol. 72, No. 1, pp. 16-26, ISSN 0924-4247, January, 1999.
 - [15] L. S. Lopes and J. H. Connell. Semisentient Robots: Routes to Integrated Intelligence, *IEEE Intelligent Systems*, Vol. 16, No. 5, pp. 10-14, ISSN 1094-7167, September/October, 2001.
 - [16] R. R. Murphy and E. Rogers. Human-Robot Interaction, *Final Report for Defense Advanced Research Projects Agency (DARPA)/National Science Foundation (NSF) Study on Human-Robot Interaction*, September 29-30, 2001.
 - [17] T. W. Fong, C. Thorpe, and C. Bauer. Collaboration, Dialogue, and Human-robot Interaction, *Proceedings of the 10th International Symposium of Robotics Research*, Lorne, Victoria, Australia, November 9-12, 2001.
 - [18] D. J. Bruemmer, J. L. Marble, D. D. Dudenhoefter, M. O. Anderson, and M. D. McKay. Mixed-Initiative Control for Remote Characterisation of Hazardous Environments, *Proceedings of the 36th IEEE Annual Hawaii International Conference on System Sciences*, pp. 127-135, ISBN 0769518745, Waikoloa Village, Hawaii, January 6-9, 2003.
 - [19] Lee, S., Intelligent Sensing and Control for Advanced Teleoperation, *IEEE Control System Magazine*, Vol. 13, No. 3, pp. 19-28, ISSN 0272-1708, 1993.
 - [20] G. Bourhis and Y. Agostini. The Vahm Robotised Wheelchair: System Architecture and Human-Machine Interaction. *Journal of Intelligent and Robotic systems*, Vol. 22, No. 1, pp. 39-50, ISSN 0921-0296, May, 1998.
 - [21] P. Hoppenot and E. Colle. Mobile Robot Command by Man-Machine Co-Operation – Application to Disabled and Elderly People Assistance, *Journal of Intelligent and Robotic Systems*, Vol. 34, pp. 235-252, ISSN 0921-0296, 2002.
 - [22] M. N. Nicolescu and M. J. Matarić. Learning and Interacting in Human-Robot Domain, *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 31, No. 5, pp.419-430, ISSN 1083-4427, 2001.
 - [23] T. Inagaki. Adaptive Automation: Sharing and Trading of Control, In: *Handbook of Cognitive Task Design*, E. Hollnagel (Ed.), Lawrence Erlbaum Associates, ISBN 0805840036, Hillsdale, New Jersey, 2003.
 - [24] H. Kawazoe, T. Murakami, O. Sadano, K. Suda, and H. Ono. Development of a Lane-Keeping Support System. *Proceedings of Intelligent Vehicle Initiative (IVI) Technology and Navigation Systems*, pp. 29-35, ISSN 1183-2770, Warrendale, PA: Society of Automotive Engineers, 2001.
 - [25] K. A. Tahboub. Natural and Manmade Shared-Control Systems: An Overview, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2655 – 2660, ISSN 1050-4729, May 21-26, 2001.
 - [26] C. E. Billings. *Aviation Automation: The Search for a Human-Centered Approach*, Lawrence Erlbaum Associates, ISBN 0805821260, Mahwah, 1997.

- [27] D. Kortenkamp, R. P. Bonasso, D. Ryan, and D. Schreckenghost. Traded Control with Autonomous Robots as Mixed Initiative Interaction, *Proceedings of the AAAI Spring Symposium on Mixed Initiative Interaction*, 1997.
- [28] M. E Agnes (Ed.). *Webster's New World College Dictionary*, 4th Edition, Webster's New World, ISBN 0028631188, 2003.
- [29] M. Haegele, J. Neugebauer, and R. D. Schraft. From Robots to Robot Assistants, *Proceedings of the 32nd International Symposium on Robotics*, pp. 404-409, ISBN 8988366042, Seoul, Korea, April 19-21, 2001.
- [30] P. Menzel and F. D'Aluisio. *Robo sapiens: Evolution of a New Species*, MIT Press, ISBN 0262133822, Cambridge, 2000.
- [31] T. Hewett, R. Baecher, S. Card, T. Carey, J. Gasen, M. Mantei, G. Perlman, G. Strong, and W. Verplank. ACM SIGCHI Curricula for Human-Computer Interaction, *Report of the ACM SIGCHI Curriculum Development Group*, 1992.
- [32] K. W. Ong, G. Seet, S. K. Sim, W. Teoh, K. H. Lim, A. N. Yow, and S. C. Low. A Testbed for Human-Robot Interactions, *Proceedings of ASME DETC'04 28th Biennial Mechanisms and Robotics Conference*, ISBN 0791837421, Salt Lake City, Utah, USA, September 28 to October 2, 2004.

References: Chapter 2

- [33] T. Thieme. *At the Controls: Columbia Presbyterian Medical Center, New York*, Popular Science, 2002.
- [34] H. T. Roman. Robotic Applications in PSEandG's Nuclear and Fossil Power Plants, *IEEE Transactions on Energy Conversion*, Vol. 8, No. 3, pp. 584-592, ISSN 0885-8969, 1993.
- [35] C. Martens, N. Ruchel, O. Lang, O. Ivlev, and A. Gräser. A Friend for Assisting Handicapped People, *IEEE Robotics and Automation Magazine*, pp. 57-65, ISSN 1070-9932, March, 2001
- [36] G. S. Wasson and J. P. Gunderson. Variable Autonomy in a Shared Control Pedestrian Mobility Aid for Elderly, *Proceedings of the IJCAI Workshop on Autonomy, Delegation and Control*, 2001.
- [37] J. Suomela and A. Halme. Novel Interactive Control Interface for Centaur-Like Service Robot, *Proceedings of the 15th IFAC World Congress on Automatic Control*, Barcelona, Spain, July 21 – 26, 2002.
- [38] W. Bluethmann, R. Ambrose, M. Diftler, E. Huber, M. Goza, C. Lovchik, and D. Magruder. Robonaut: A Robot Designed to Work with Humans in Space, *Autonomous Robots*, Vol. 14, pp. 179-207, ISSN 0929-5593, 2003.
- [39] P. Fiorini and E. Prassler. Cleaning and Household Robots: A Technology Survey, *Autonomous Robots*, Vol. 9, pp. 227-235, ISSN 09295593, Kluwer Academic Publishers, The Netherlands, 2000.
- [40] W. Burgard, A. B. Cremers, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. The Interactive Museum Tour-Guide Robot, *Proceedings of the 15th National Conference on Artificial Intelligence*, ISBN 0262510987,

- Madison, Wisconsin, AAAI Press, July 26-30, 1998.
- [41] D. A. Norman. *The Design of Everyday Things*, Basic Book, ISBN 0465067107, New York, USA, 2002.
 - [42] J. Vertut and P. Coiffet. *Teleoperation and Robotics - Evolution and Development*, Robot Technology Vol. 3A, Kogan Page, ISBN 0850385881, London, UK, 1985.
 - [43] T. W. Fong. *Collaborative Control: A Robot-Centric Model for Vehicle Teleoperation*, Ph.D. Dissertation, Robotics Institute, Carnegie Mellon University, USA, 2001.
 - [44] Y. N. Shimon (Ed.). *Handbook of Industrial Robotics*, 2nd Edition, ISBN 0471177830, John Wiley and Sons, 1999.
 - [45] R. C. Arkin. *Behavior-Based Robotics*. MIT Press, ISBN 0262011654, Cambridge, 1998.
 - [46] R. Y. Chang. *Success through Teamwork: A Practical Guide to Interpersonal Team Dynamics*, Kogan Page, ISBN 0749416610, UK, 1995.
 - [47] D. J. Bruemmer, J. L. Marble and D. D. Dudenhoeffer. Mutual Initiative in Human-Machine Teams, *Proceedings of the 7th IEEE Conference on Human Factors and Power Plants*, Scottsdale, AZ, pp. 22-30, ISBN 0780374509, September, 2002.
 - [48] E. Jackson, O. Williams, and K. Buchan. Achieving Robot Autonomy, *Proceedings of the 3rd Conference on Military Robotic Applications*, DND, Canada, September, 1991.
 - [49] S. Zhai and P. Milgram. Human Robot Synergism and Virtual Telerobotics Control, *Proceedings of the 25th Annual Conference of the Human Factors Association of Canada*, 1992.
 - [50] V. Klingspor, J. Demiris, and M. Kaiser. Human-Robot Communication and Machine Learning, *Applied Artificial Intelligence*, Vol. 11, pp.719-746, ISSN 1087-6545, 1997.
 - [51] A. Green and K. S. Eklundh. Designing for Learnability in Human-Robot Communication, *IEEE Transactions on Industrial Electronics*, Vol. 50, No. 4, ISSN 0278-0046, August, 2003.
 - [52] R. E. Curry and A. R. Ephrath. Monitoring and Control of Unreliable Systems, In: *Monitoring Behaviour and Supervisory Control*, T. B. Sheridan and G. Johannsen (Eds.), Vol. 1, pp. 193-203, Plenum Press, ISBN 030632881X, New York, USA, 1976.
 - [53] R. R. Murphy and E. Rogers. Cooperative Assistance for Remote Robot Supervision, *Presence: Teleoperators and Virtual Environments, Special Issue*, Vol. 5, No. 2, pp. 224-240, ISSN 1054-7460, 1996.
 - [54] D. R. Yoerger and J. E. Slotine. Supervisory Control Architecture for Underwater Teleoperation, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2068-2073, ISSN 1050-4729, 1987.
 - [55] Y. Yokokohji, A. Ogawa, H. Hasunuma, and T. Yoshikawa. Operation Modes

- for Cooperating With Autonomous Functions in Intelligent Teleoperation Systems, *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 3, pp. 510 – 515, ISBN 0818634502, IEEE Computer Society Press, May 2-6, 1993.
- [56] N. P. Papanikolopoulos and P. K. Khosla. Shared and Traded Telerobotics Visual Control, *Proceedings of the IEEE Conference on Robotics and Automation*, pp. 878-885, ISSN 1050-4729, 1992.
 - [57] T. W. Fong, C. Thorpe, and C. Bauer. A Safeguarded Teleoperation Controller, *Proceedings of the IEEE International Conference on Advanced Robotics*, ISBN 9637154051, Budapest, Hungary, 2001.
 - [58] E. Krotkov, R. Simmons, F. Cozman, and S. Koenig. Safeguarded Teleoperation for Lunar Rovers: From Human Factors to Field Trials, *IEEE Planetary Rover Technology and Systems Workshop*, Minneapolis, Minnesota, USA, April, 1996.
 - [59] R. C. Arkin. Reactive Control as a Substrate for Telerobotic Systems, *IEEE AES Systems Magazine*, Vol. 6, No. 6, pp. 24–31, ISSN 0885-8985, New York, USA, 1991.
 - [60] I. Lin, F. Wallner, and R. Dillmann. An Advanced Telerobotic Control System for a Mobile Robot with Multisensor Feedback, *Proceedings of Intelligent Autonomous System (IAS-4)*, U. Rembold and R. Dillmann (Eds.), pp. 365-372, IOS Press, Amsterdam, The Netherlands, 1995.
 - [61] W. R. Ferrell and T. B. Sheridan. Supervisory Control of Remote Manipulation, *IEEE Spectrum*, Vol. 4, No. 10, pp. 81-88, ISSN 0018-9235, October, 1967.
 - [62] G. Dorais and D. Kortenkamp. Designing Human Centered Autonomous Agents, *Tutorial given at the Proceedings of the 6th Pacific Rim International Conference on Artificial Intelligence*, Melbourne, Australia, August 28 - September 1, 2000.
 - [63] T. Callantine. *Tracking Operator Activities in Complex Systems*, Ph.D. Dissertation, Industrial and Systems Engineering, Georgia Institute of Technology, USA, 1996.
 - [64] A. Degani, C. M. Mitchell, and A. R. Chappell. Task models to guide analysis: Use of the operator function model to represent mode transitions, *Proceedings of the 8th International Symposium on Aviation Psychology*, R. S. Jensen (Ed.), Columbus, OH, USA, 1995.
 - [65] J. Rasmussen. Skills, Rules and Knowledge; Signals, Signs and Symbols, and other Distinctions in Human Performance Models, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 13, No. 3, pp.257-266, ISSN 1083-4427, 1983.
 - [66] T. Sawaragi, T. Shiose, and G. Akashi. Foundations for Designing an Ecological Interface for Mobile Robot Teleoperation, *Robotics and Autonomous systems*, Vol. 31, pp. 193-207, ISSN 0921-8890, 2000.
 - [67] D. Kortenkamp, R. Burridge, R. P. Bonasso, D. Schrenkenghoist, and M. B. Hudson. An Intelligent Software Architecture for Semi-autonomous Robot Control, *Proceedings of the 3rd International Conference on Autonomous Agents Workshop on Autonomy Control Software*, 1999.
 - [68] J. Brederke and A. Lankenau. A Rigorous View of Mode Confusion,

- Proceedings of SafeComp*, S. Anderson, S. Bologna, M. Felici (Eds.), pp. 19-31, ISSN 0302-9743, Springer Verlag, Catania, Italy, September 10-13, 2002.
- [69] S. Braynov and H. Hexmoor. Quantifying Relative Autonomy in Multiagent Interaction, In: *Agent Autonomy*, H. Hexmoor, C. Cristiano and R. Falcone, (Eds.), pp. 43-56, Kluwer Academic Publishers, ISBN 1402074026, Boston, USA, 2002.
- [70] J. M. Bradshaw, G. Boy, E. Durfee, M. Gruninger, H. Hexmoor, N. Suri, M. Tambe, M. Uschold, and J. Vitek (Eds.). *Software Agents for the Warfighter*, ITAC Consortium Report, AAAI Press/MIT Press, Cambridge, 2002.
- [71] T. B. Sheridan. Supervisory Control, In: *Handbook of Human Factors/Ergonomics*, G. Salvendy (Ed.), Wiley, ISBN 0471116, New York, USA, 1987.
- [72] R. A. Brooks. Elephants Don't Play Chess, *Designing Autonomous Agents, Theory and Practice from Biology to Engineering and Back*, P. Maes (Ed.), pp. 3-15, MIT Press, ISBN 0262631350, Cambridge, 1990.
- [73] R. C. Arkin and T. R. Balch. AuRA: Principles and Practice in Review, *Journal of Experimental and Theoretical Artificial Intelligence*, Vol. 9, No. 2, ISSN 1362-3079, 1997.
- [74] E. Gat. On Three-Layer Architectures, *Artificial Intelligence and Mobile Robots: Case Studies of Successful Robotics System*, Kortenkamp, D., Bonnasso, P. R., and Murphy, R. (Eds.), ISBN 0262611376, AAAI Press, 1998.
- [75] A. Saffiotti. The Uses of Fuzzy Logic in Autonomous Robot Navigation, *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, Vol. 1, No. 4, pp. 180-197, ISSN 1433-7479, 1997.
- [76] P. Pirjanian. Behavior Coordination Mechanisms - State-of-the-Art, *Tech-report IRIS-99-375*, Institute for Robotics and Intelligent Systems, School of Engineering, University of Southern California, USA, October, 1999.
- [77] R. A. Brooks. A Robust Layered Control System for a Mobile Robot, *IEEE Journal of Robotics and Automation*, Vol. RA-2, pp. 14-23, ISSN 0882-4967, April, 1986.
- [78] P. Maes. How to Do the Right Thing, *Connection Science Journal*, Vol. 1, No. 3, pp. 291-323, ISSN 1360-0494, 1990.
- [79] J. Rosenblatt and D. Payton. A Fine-Grained Alternative to the Subsumption Architecture for Mobile Robot Control, *Proceedings of the IEEE/INNS International Joint Conference on Neural Networks*, Vol. 2, pp. 317-324, ISSN 1098-7576, Washington DC, USA, June, 1989.
- [80] R. Bajcsy and J. Košecká. Discrete Event Systems for Autonomous Mobile Agents, *International Journal of Robotics and Autonomous Systems*, Vol. 12, pp. 187-198, ISSN 0921-8890, Elsevier, 1994.
- [81] R. C. Arkin and D. Mackenzie. Temporal Coordination of Perceptual Algorithms for Mobile Robot Navigation. *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 3, pp. 276-286, ISSN 1042-296X, June, 1994.

- [82] S. Kristensen. Sensor Planning with Bayesian Decision Theory, *International Journal of Robotics and Autonomous Systems*, Vol. 19, No. 3-4, pp. 273-286, ISSN 0921-8890, Elsevier, March, 1997.
- [83] L. P. Kaelbling, M. L. Littman and A. W. Moore. Reinforcement Learning: A Survey, *Journal of Artificial Intelligence Research*, Vol. 4, ISSN 1076-9757, AAAI Press, 1996.
- [84] O. Khatib. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *International Journal of Robotics Research*, Vol. 5, No. 1, pp. 90-98, ISSN 0278-3649, MIT Press, 1986.
- [85] R. C. Arkin. Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behaviour, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 264-271, ISSN 1050-4729, Raleigh, NC, USA, 1987.
- [86] J. K. Rosenblatt. Utility Fusion: Map-Based Planning in a Behavior-Based System, *Proceedings of the 3rd International Conference on Field and Service Robotics*, A. Zelinsky (Ed.), pp. 411-418, ISBN 1852330392, Springer-Verlag, 1998.
- [87] P. Pirjanian. *Multiple Objective Action Selection and Behaviour Fusion Using Voting*, Ph.D. Dissertation, Department of Medical Informatics and Image Analysis, Institute of Electronic Systems, Aalborg University, Fredrik Bajers Vej 7, DK-9220 Aalborg, Denmark, August, 1998.
- [88] P. Pirjanian and M. J. Matarić. A Decision-Theoretic Approach to Fuzzy Behaviour Coordination, *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, ISBN 0780358066, IEEE Press, November, 1999.
- [89] A. R. Graves and C. Czarnecki. Design Patterns for Behaviour-based Robotics, *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 30, No. 1, pp.36-41, ISSN 1083-4427, 2000.
- [90] T. W. Fong, F. Conti, S. Grange, and C. Baur. Novel Interfaces for Remote Driving: Gesture, Haptic and PDA, *SPIE Telemanipulator and Telepresence Technologies VII*, Boston, MA, USA, November, 2000.
- [91] K. Myers and D. Morley. Human Directability of Agents, *Proceedings of the 1st International Conference on Knowledge Capture*, ISBN 1581133804, ACM Press, 2001.
- [92] J. Scholtz. Evaluation Methods for Human-System Performance of Intelligent Systems, *Proceedings of the Performance Metrics for Intelligence System (PerMIS) Workshop*, E. R. Messina and A. M. Meystel (Eds.), August 13-15, 2002.

References: Chapter 3

- [93] T. B. Sheridan. Task Analysis, Task Allocation and Supervisory Control, In: *Handbook of Human-Computer Interaction (2nd, Completely Revised Edition)*, M. Helander, T. K. Landauer and P. Prabhu (Eds.), pp. 87-105, Elsevier, ISBN

- 0444818262, Amsterdam, New York, USA, 1997.
- [94] S-L. Hwang, W. Barfield, T-C. Chang, and G. Salvendy. Integration of Humans and Computers in the Operation and Control of Flexible Manufacturing Systems, *International Journal of Production Research*, Vol. 22, pp. 841-851, ISSN 0020-7543, Taylor and Francis, 1984.
 - [95] P. M. Fitts. *Human Engineering for an Effective Air Navigation and Traffic Control System*, National Research Council, Washington, D.C, USA, 1951.
 - [96] G. A. Bekey. The Human Operator in Control Systems. In: *Systems Psychology*, K. B. De Greene (Ed.), pp. 248-277, McGraw-Hill, ISBN 0070162387, New York, USA, 1970.
 - [97] D. Meister. *Behavioral Analysis and Measurement Methods*, Wiley, ASIN 0471896403, New York, USA, 1982.
 - [98] N. Jordan. Allocation of Function between Man and Machines in Automated System, *Journal of Applied Psychology*, Vol. 47, pp. 161-165, ISSN 0021-9010, 1963.
 - [99] J. M. Bradshaw, M. Sierhuis, A. Acquisti, P. Feltovich, R. Hoffman, R. Jeffers, D. Prescott, N. Suri, A. Uszok, and R. V. Hoof. Adjustable Autonomy and Human-Agent Teamwork in Practice: An Interim Report on Space Applications, In: *Agent Autonomy*, H. Hexmoor, C. Cristiano and R. Falcone, (Eds.), pp. 191-220, Kluwer Academic Publishers, ISBN 1402074026, Boston, USA, 2002.
 - [100] T. B. Sheridan. Telerobotics, *Automatica*, Vol. 25, No. 4, pp. 487-507, ISSN 0005-1098, 1989.
 - [101] D. D. Woods. Steering the Reverberations of Technology Change on Fields of Practice: Laws That Govern Cognitive Work, *Proceedings of the 24th Ann. Meeting Cognitive Science Soc.*, Erlbaum, L., Mahwah, New Jersey, USA, 2002.
 - [102] R. Hoffman, P. Feltovich, K. M. Ford, D. D. Woods, G. Klein, and A. Feltovich. A Rose by Any Other Name... Would probably be Given an Acronym, *IEEE Intelligent Systems*, pp. 72-80, ISSN 1541-1672, 2002.
 - [103] W. B. Rouse. Twenty Years of Adaptive Aiding: Origins of the Concept and Lessons Learned. In: *Human Performance in Automated Systems: Current Research and Trends*, M. Mouloua and R. Parasuraman (Eds.), pp. 28-32, Lawrence Erlbaum Associates, ISBN 080581616X, Hillsdale, New Jersey, USA, 1994.
 - [104] P. A. Hancock and S. F. Scallen. Allocating Functions in Human-Machine Systems. In: *Viewing Psychology as a Whole*, R. Hoffman, M. F. Sherrick, and J. S. Warm (Eds.), pp. 509-540, American Psychological Association, ISBN 1557984743, Washington, D.C, USA, 1998.

References: Chapter 4

- [105] E. Messina, A. Meystel, and L. Reeker. Measuring Performance and Intelligence of Intelligent systems, *Proceedings of the Performance Metrics for Intelligence System (PerMIS) Workshop*, Mexico City, Mexico, September 4, 2001.

- [106] B. Chandrasekaran, J. R. Josephson, and V. R. Benjamins. What are Ontologies and Why Do We Need Them?, *IEEE Intelligent Systems*, Vol. 14, No. 1, pp. 20-26, ISSN 1541-1672, January/February, 1999.
- [107] N. Guarino and P. Giaretta. Ontologies and Knowledge Bases – Towards a Terminological Clarification, In: *Towards Very Large Knowledge Bases - Knowledge Building and Knowledge Sharing*, N. J. I. Mars (Ed.), pp. 25-32., IOS Press, ISBN 9051992173, Amsterdam, The Netherlands, 1995.
- [108] H. Geffner, and J. Wainer. Modelling Action, Knowledge and Control, *Proceedings of the 13th European Conference on Artificial Intelligence*, pp. 532-536, ISSN 0922-6389, 1998.
- [109] G. E. Hovland and B. J. McCarragher. Hidden Markov Models as a process Monitor in Robotic Assembly, *Modelling Identification and Control*, Vol. 20, No. 4, pp. 201-223, 1999.
- [110] M. C. Nechyba and Y. Xu. Human Control Strategy: Abstraction Verification and Replication, *IEEE Control Systems Magazine*, Vol. 17, No. 5, pp. 48-61, ISSN 0272-1708, 1997.
- [111] E. Demeester, M. Nuttin, D. Vanhooy-donck, and V. H. Brussel. Assessing the User's Intent using Bayes' Rule: Application to Wheelchair Control, *Proceedings of the 1st International Workshop on Advance in Service Robotics*, pp. 117-124, Bardolino, Italy, March 13-15, 2003.
- [112] M. N. Nicolescu. *A Framework for Learning from Demonstration, Generalization and Practice in Human-Robot Domains*, Ph.D. Dissertation, University of Southern California, USA, 2003.
- [113] Y. Kuniyoshi, M. Inaba, and H. Inoue. Learning by Watching: Extracting Reusable Task Knowledge from Visual Observation of Human Performance, *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 6, pp. 799-822, ISSN 1042-296X, 1994.
- [114] P. Jian. *Extraction of Salient Features from Sensory-Motor Sequences for Mobile Robot Navigation*, Ph.D. Dissertation, University of Vanderbilt, USA, 2004.
- [115] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, ISBN 0131038052, New Jersey, 1995.
- [116] S. K. Sim, K. W. Ong, and G. Seet. A Foundation of Robot Learning, *Proceedings of the 4th IEEE International Conference on Control and Automation*, pp. 649-653, ISBN 078037777X, Montreal, Canada, June 10-12, 2003.

References: Chapter 5

- [117] H. F. Durrant-Whyte. A Critical Review of the State-of-the-Art in Autonomous Land Vehicle Systems and Technology, *Sandia Report, SAND2001-3685*, Sandia National Laboratories, USA, November, 2001.
- [118] J. Borenstein, H. R. Everett and L. Feng. Where am I? Sensors and Methods for Mobile Robot Positioning. *Robotics Technology Development Program*,

- University of Michigan, USA, April, 1996.
- [119] S. Suksakulchai, S. Thongchai, D. M. Wikes, and N. Sarkar. Mobile Robot Localization using an Electronic Compass for Corridor Environment, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, ISSN 0360-8913, Nashville, Tennessee, USA, October 8-11, 2000.
 - [120] N. Manouk. IEEE 802.11 Communications Test for Robotic Systems, *AUVSI Unmanned Systems*, Lake Buena Vista, FL, July 9-11, 2002.
 - [121] BreezeCom. BreezeNet PRO.11 Series User's Guide. *BreezeCOM Ltd.*, Cat No. 213083, 1999.
 - [122] G. N. DeSouza and A. C. Kak. Vision for Mobile Robot Navigation: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 2, pp. 237-267, ISSN 0018-9340, February, 2002.
 - [123] P. Gaussier, C. Joulain, S. Zrehen, J. P. Banquet, and A. Revel. Visual Navigation in an Open Environment without Map, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 545-550, ISBN 0780341201, Grenoble, France, September, 1997.
 - [124] D. Lambrinos, M. Maris, H. Kobayashi, T. Labhart, R. Pfeifer, and R. Wehner. Navigating with a Polarized Light Compass, *Proceedings of the IEEE on Self-Learning Robots II: Bio-robotics*, pp.7/1 - 7/4, February 2, 1998.
 - [125] J. J. Leonard and H. F. Durrant-Whyte. Simultaneous Map Building and Localisation for an Autonomous Mobile Robot, *Proceedings of the IEEE/RSJ International Workshop on Intelligent Robots and Systems*, Vol. 3, pp. 1442-1447, 1991.
 - [126] E. Nebot. Simultaneous Localisation and Mapping, *Summer School (Version 0.9)*, Australian Centre for Field Robotics, University of Sydney NSW 2006, Australia, July 31, 2002.
 - [127] S. Thrun. Robotic mapping: A survey, G. Lakemeyer and B. Nebel (Eds.), *Exploring Artificial Intelligence in the New Millennium*, Chapter 1, Morgan Kaufmann, ISBN 1558608117, San Mateo, California, USA, 2002.
 - [128] J. Borenstein and L. Feng. Gyrodometry: A New Method for Combining Data from Gyros and Odometry in Mobile robots, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 423-428, ISSN 1050-4729, Minneapolis, Minnesota, USA, April 22-28, 1996.
 - [129] B. Barshan and H. F. Durrant-Whyte. Inertial Navigation Systems for Mobile Robots, *IEEE Transactions on Robotics and Automation*, Vol. 11, No. 3, pp. 328-342, ISSN 1042-296X, June, 1995.
 - [130] D. Fox. KLD-Sampling: Adaptive Particle Filters, T. G. Dietterich; S. Becker and Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems*, Vo. 14, pp.713-720, MIT Press, ISBN 1558602224, Cambridge, December 3-8, 2001.
 - [131] J. Latombe. *Robot Motion Planning*, Kluwer Academic Publishers, ISBN 0792391292, The Netherlands, 1990.

-
- [132] S. Healy. *Spatial Representation in Animals*, Oxford University Press, ISBN 0198500076, UK, 1997.
 - [133] O. Wijk and H. Christensen. Extraction Of Natural Landmarks And Localisation Using Sonars, *Proceedings of the International Symposium on Intelligent Robotic Systems*, Edinburgh, Scotland, July, 1998.
 - [134] L. A. Zadeh. Fuzzy Logic, Neural Networks, and Soft Computing, *Communications of the ACM*, Vol. 37, No. 3, pp. 77-84, ISSN 0001-0782, March, 1994.
 - [135] L.-X. Wang and J.M. Mendel. Generating Fuzzy Rules by Learning from Examples, *Proceedings of the IEEE International Symposium on Intelligent Control*, pp. 263 – 268, August 13-15, 1991.
 - [136] S. Thongchai. Behavior-Based Learning Fuzzy Rules for Mobile Robots, *Proceedings of the American Control Conference*, ISSN 0743-1619, Anchorage, Alaska, USA, May 8-10, 2002.
 - [137] F. Hoffmann. Fuzzy Behavior Coordination for Robot Learning from Demonstration, *Proceedings of the 23rd International Conference of the North American Fuzzy Information Processing Society*, Banff, Canada, June, 2004.
 - [138] G. e. P. Box and G. M. Jenkin. *Time Series Analysis: Forecasting and Control*, Holden-Day Inc., ISBN 0816211043, 1976.
 - [139] L. Itti, C. Koch, and E. Niebur. A Model of Saliency-Based Visual Attention for Scene Analysis, *IEEE Transactions on Pattern analysis and Machine Intelligence*, Vol. 20, No. 11, pp. 1254-1259, ISSN 0018-9340, November, 1998.
 - [140] M.-S. Kim and K. R. Cave. Spatial Attention in Visual Search for Features and Feature conjunctions, *Psychological Science*, Vol. 6, pp. 376-380, ISSN 0956-7976, 1995.
 - [141] M. Jagersand. Saliency Maps and Attention Selection in Scale and Spatial Coordinates: An Information Theoretic Approach, *Proceedings of the 5th IEEE International Conference on Computer Vision*, pp. 195-202, ISBN 0818670428, MIT, Cambridge, June 20-23, 1995.
 - [142] L. Itti. The Beobot Platform for Embedded Real-Time Neuromorphic Vision, In: *Advances in Neural Information Processing Systems, Vol. 15, Hardware Demo Track*, T. G. Dietterich, S. Becker, Z. Ghahramani (Eds.), MIT Press, ISBN 0262025507, Cambridge, 2003.
 - [143] B. J. W. Waarsing, M. Nuttin, and H. Van Brussel. A Software Framework for Control of Multi-Sensor, Multi-Actuator System, *Proceedings of the 11th IEEE International Conference on Advanced Robotics*, pp. 41 - 46, ISBN 9729688990, Coimbra, Portugal, June 30 - July 3, 2003.
 - [144] J. Lipton and R. A. Jarvis. Madron - Multi-Agent Distributed Robot Navigation, *Proceedings of the 8th Australian Joint Conference on Artificial Intelligence*, Canberra, Australia, November, 1995.
 - [145] P. Giorgini, M. Kolp and J. Mylopoulos. Socio-Intentional Architectures for Multi-Agent Systems: the Mobile Robot Control Case, *Proceedings of the 4th International Bi-Conference Workshop on Agent-Oriented Information Systems*,

- ISSN 1613-0073, Toronto, Ontario, Canada, May 27-28, 2002.
- [146] A. H. Bond and L. Gasser. *Readings in Distributed Artificial Intelligence*, Morgan Kaufmann, ISBN 093461363X, San Mateo, California, USA, 1988.
 - [147] J. Gowdy. A Qualitative Comparison of Interprocess Communications Toolkits for Robotics. *Technical Report CMU-RI-TR-00-16*, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, 2000.
 - [148] E., Petroustos. *Mastering Visual Basic 6*, Sybex, Inc., ISBN 0782122728, San Francisco, USA, 1998.
 - [149] IS Robotics, Inc., Real World Interface Division, www.isr.com, 32 Fitzgerald Dr., Jaffrey, NH 03452, USA. *Mobility Robot Integration Software User's Guide*, Part Number: 2841; Rev. 5, 2002.
 - [150] M. Henning and S. Vinoski. *Advanced CORBA Programming with C++*, Addison Wesley Longman, ISBN 0201379279, Massachusetts, USA, 1999.
 - [151] Object Management Group. *CORBA services: Common Object Services Specification*, [ftp://ftp.omg.org/pub/docs/formal/98-07-05.pdf](http://ftp.omg.org/pub/docs/formal/98-07-05.pdf), Object Management Group, Framingham, MA, USA, 1997.

References: Chapter 6

- [152] J. Borenstein and Y. Koren. Real-time Obstacle Avoidance for Fast Mobile Robots, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 5, pp. 1179-1187, ISSN 1083-4427, September/October, 1989.
- [153] R. A. Cooke. Microcomputer Control of Free Ranging Robots, *Proceedings of the 13th International Symposium on Industrial Robots and Robots*, pp. 13.109-13.120, Chicago, Ill., USA, April, 1983.
- [154] R. Chattergy. Some Heuristics for the Navigation of a Robot, *International Journal of Robotics Research*, Vol. 4, No. 1, pp. 59-66, ISSN 0278-3649, MIT Press, 1985.
- [155] A. Elfes. Sonar-Based Mapping and Navigation System, *Technical Report*, Carnegie-Mellon University, The Robotics Institute, pp. 25-30, USA, 1985.
- [156] H. P. Moravec. Sensor Fusion in Certainty Grids for Mobile Robots, *AI Magazine*, pp. 61-74, ISSN 0738-4602, 1988.
- [157] J. Borenstein and Y. Koren. The Vector Field Histogram - Fast Obstacle-Avoidance for Mobile Robots, *IEEE Journal of Robotics and Automation*, Vol. 7, No. 3, pp. 278-288, ISSN 0882-4967, June, 1991.
- [158] I. Ulrich and J. Borenstein. VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1572-1577, ISSN 1050-4729, Leuven, Belgium, May 16-21, 1998.
- [159] I. Ulrich and J. Borenstein. VFH*: Local Obstacle Avoidance with Look-ahead Verification, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2505-2511, ISSN 1050-4729, San Francisco, CA, April 24-28, 2000.

- [160] R. E. Kirk. *Experimental Design: Procedures for the Behavioral Sciences*, Brooks/Cole, ISBN 0534250920, 1995.
- [161] J. Scholtz and S. Bahrami. Human-Robot Interaction: Development of an Evaluation Methodology for the Bystander Role of Interaction, *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, Vol. 4, pp. 3212 – 3217, ISSN 062-922X, October 5-8, 2003.
- [162] J. Matsumura. *Exploring Advanced Technologies For The Future Combat Systems Program*. Rand, ISBN 0833030264, Santa Monica, USA, 2002.
- [163] R. Jain, R. Kasturi, and B.G. Schunck. *Machine Vision*, McGraw-Hill Inc., ISBN 0070320187, New York, 1995.
- [164] B. Jung and S. Gaurav. Detecting Moving Objects using a Single Camera on a Mobile Robot in an Outdoor Environment, *Proceedings of the 8th International Conference on Intelligent Autonomous Systems*, pp. 980-987, Amsterdam, The Netherlands, March, 2004.
- [165] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, W. H. Freeman and Company, ISBN 0716715678, San Francisco, USA, 1982.

References: Chapter 7

- [166] K. W. Ong, G. Seet, and S. K. Sim. Sharing and Trading in a Human-Robot System, V. Kordic, A. Lazinica and M. Merdan (Eds.), *Cutting Edge Robotics, Chapter 2 – Human Robot Interaction*, pp. 467-499, ISBN 3-86611-038-3, 2005.
- [167] W. C. Jacob and M. A. Goodrich. Characterizing Efficiency of Human-Robot Interaction: A Case Study of Shared-Control Teleoperation, *Proceedings of the International Conference on Intelligent Robots and Systems*, pp. 1290-1295, ISBN 0780373987, EPFL, Lausanne, Switzerland, October, 2002.
- [168] B. Barber. *The Logic and Limits of Trust*, Rutgers University Press, ISBN 0813509580, New Brunswick, New Jersey, USA, 1983.
- [169] J. Lee and N. Moray. Trust, Self-Confidence and Supervisory Control in a Process Control Simulation, *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 1, pp. 291 -295, ISSN 0884-3619, October 13-16, 1991.
- [170] G. Abe, M. Itoh, and K. Tanaka. Trust in and Use of Automation: Their Dependence on Occurrence Patterns of Malfunctions Systems, *Proceedings of IEEE International Conference on System, Man, and Cybernetics*, Vol. 3, pp. 715 -720, ISSN 0884-3619, 1999.
- [171] D. Waard de. *The Measurement of Drivers' Mental Workload*, Ph.D. Dissertation, Centre for Environmental and Traffic Psychology, University of Groningen, The Netherlands, 1996.

APPENDICES

- Appendix A: Robot Hardware
- Appendix B: Implementation Details
- Appendix C: Questionnaire
- Appendix D: Experimental Results
- Appendix E: Computer Vision Libraries

Appendix A: Robot Hardware

This research uses numerous ready off-the-shelves toolsets to build up the basic infrastructure presented in Chapter 5. In order to provide a basis, this appendix gives a description of the mobile robot and its hardware toolsets employed:

- A1. All-Terrain Mobile Robot (ATRV-Jr™)
- A2. Polaroid Sonar Sensors
- A3. Sick Laser Measurement System LMS 200-30106
- A4. Sony EVI-D30 Video Camera
- A5. KVH-C100 Compass
- A6. Odometry
- A7. Crossbow Attitude & Heading Reference System (AHRS) 400CA
- A8. BreezeNET PRO.11 Wireless Modems
- A9. Premier Wireless Video CS-220

A1 All-Terrain Mobile Robot (ATRV-Jr™)

The ATRV-Jr™ manufactured by iRobot Corporation [1] is for operating in both outdoor and indoor. It is equipped with a set of 17 Polaroid sonar sensors. This robot is powered by two 12V gel-cell batteries. It has a differential drive system, which uses two electrical DC brush motor. The two wheels on either side are driven together and can be controlled independently (skid steering, discuss below) with a maximum speed of 1.7 m/s, which enables the robot to turn in every direction on the spot. Therefore the robot's movement is comparable to the movement of a track vehicle. Inside the robot works an Intel Pentium III™ double processor PC system at a clock rate of 800 MHz. The motherboard is mounted in a special aluminum enclosure with a shock-mounted 40GB IDE hard-disk drive. The robot is currently running on RedHat 6.2, a Linux operating system and is configured to the MOBILITY Robot Integration Software [2].

A1.1 Characteristics of Skid Steering Drive System

Skid steering drive system can be compact, light, require few parts, and exhibit agility from point turning to line driving using only the motions, components, and swept volume needed for straight driving. Skid steering is achieved by creating a differential thrust between the left and right sides of the robot thus causing a change in heading. This is an effective and easy solution to steer the robot. However, it is not as accurate as other steering methods; certain characteristics including friction, wheel slippage and other unpredictable attributes can cause problems. This steering configuration is a special case where the bisectors of the wheels do not intersect and wheel slip is exploited to cause the robot to rotate. In addition, skidding causes unpredictable power requirements because of terrain irregularities and non-linear tire-ground interaction. In short, the advantages of this drive system include mechanical/control simplicity and high maneuverability. The disadvantage is it requires high steering power. The equations of motion for controlling such a drive system are discussed below.

A1.2 Equations of Motion of ATRV-Jr™

The following assumptions are made for formulating the free body diagram of the ATRV-Jr™ presented in Figure A.1:

- Robot is moving on the horizontal plane
- Robot speed is very low
- Longitudinal slippage neglected
- Lateral force of the tire is directly proportional to its vertical load
- Wheel actuation is equal on each side to reduce longitudinal slip

As shown in Figure A1, (X, Y) defines a fixed reference frame, (x, y) is a moving frame attached to the robot body with origin at the centre of mass; i.e. the robot's Cartesian position. The centre of mass is located at distances " a " and " b " from front and rear wheels respectively. Wheelbase is " $2d$ ". " θ " is the heading (orientation) = Angle of x axis – with X -axis. Following this, the rotation matrix relating the

coordinate frames is given by:

$$R(\theta) = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \quad (\text{A.1})$$

Figure A1: Free body diagram of ATRV-Jr™

Given the robot position and orientation (called *location*, i.e. $(x(t), y(t), \theta(t))$), the dynamics model of this kind of mobile robot can be described by the following differential equations (Kanayama et al. [3]):

$$\begin{aligned} \dot{x}(t) &= v(t) \cos \theta(t) \\ \dot{y}(t) &= v(t) \sin \theta(t) \\ \dot{\theta}(t) &= \omega(t) \end{aligned} \quad (\text{A.2})$$

Where $v(t)$ is the linear velocity and $\omega(t)$ is the angular velocity of the mobile robot, and they are the control variables. Robot's motion is controlled by giving the correct linear and angular profile. This type of control method is employed in this implementation for both manual and exclusive shared control. In addition, the robot can also reach a specified location (x, y, θ) by controlling the linear and angular profile. This type of control method is employed in this implementation for both exclusive shared and autonomous control.

The above described the robot motion in local frame f . In the fixed frame F , the absolute velocities are:

$$\begin{pmatrix} \dot{X}(t) \\ \dot{Y}(t) \end{pmatrix} = \begin{pmatrix} \cos \theta(t) & \sin \theta(t) \\ -\sin \theta(t) & \cos \theta(t) \end{pmatrix} \begin{pmatrix} \dot{x}(t) \\ \dot{y}(t) \end{pmatrix} \quad (\text{A.3})$$

Differentiating equation (A.3) with respect to time gives the acceleration:

$$\begin{pmatrix} \ddot{X}(t) \\ \ddot{Y}(t) \end{pmatrix} = R(\theta(t)) \begin{pmatrix} \ddot{x}(t) + \dot{x}(t)\dot{\theta}(t) \\ \ddot{y}(t) - \dot{y}(t)\dot{\theta}(t) \end{pmatrix} = R(\theta(t)) \begin{pmatrix} a_x \\ a_y \end{pmatrix} \quad (\text{A.4})$$

Where the longitudinal velocity $\dot{x}_i$ and lateral velocity $\dot{y}_i$ are given by:

$$\begin{aligned} \dot{x}_1(t) = \dot{x}_4(t) &= \dot{x}(t) - d\dot{\theta}(t) && (\text{left}) \\ \dot{x}_2(t) = \dot{x}_3(t) &= \dot{x}(t) + d\dot{\theta}(t) && (\text{right}) \\ \dot{y}_1(t) = \dot{y}_2(t) &= \dot{y}(t) + a\dot{\theta}(t) && (\text{front}) \\ \dot{y}_3(t) = \dot{y}_4(t) &= \dot{y}(t) - b\dot{\theta}(t) && (\text{rear}) \end{aligned} \quad (\text{A.5})$$

The free body diagram of forces and velocities is shown in Figure A.1. As discussed above, the robot has velocities $\dot{x}$, $\dot{y}$ and $\dot{\theta}$. The four wheels of the robot develop tractive forces F_{xi} and are subject to rolling resistance R_{xi} where $i = 1, 2, 3, 4$. As wheel actuation is equal, $F_{x1} = F_{x4}$ and $F_{x2} = F_{x3}$. Lateral forces F_{yi} acts against lateral skidding and there is a resistance moment M_r about the centre of mass due to F_{yi} and R_{xi} . For the robot of mass m and moment of inertia I , the equations of motion in frame f can be written as follows:

$$\begin{aligned} ma_x &= 2F_{x1} + 2F_{x2} - R_x \\ ma_y &= -F_y \\ I\ddot{\theta} &= 2d(F_{x1} - F_{x2}) - M_r \end{aligned} \quad (\text{A.6})$$

A2 Polaroid Sonar Sensors

Sonar is a form of active sensing. It operates on the time-of-flight principle. A high frequency click of sound is emitted that reflects off a nearby surface and returns back. The distance to the surface that reflected the sound is half of the speed of the sound multiplies the delay time. The range of sonar sensor is from 0.5 to 4 metres. The main advantage of sonar sensor is its low cost. However, due to the nature of sonar sensors (e.g. crosstalk, specular reflection, etc.), the readings from these sensors are neither reliable nor accurate.

A3 Sick Laser Measurement System LMS 200-30106

Similar to sonar, laser is also a form active sensing based on time-of-flight principle. The LMS 200 proximity laser scanner produced by Sick Optic Electronic Inc. is a measurement system that works without contact by scanning its surroundings in two dimensions (laser radar) up to 180°. As it is a scanning system, neither reflectors nor positional markers are required. The LMS 200 high-resolution laser measurement system solves measurement functions, which have, up to now, been impossible or could only be achieved with great difficulty. The system carries out measurement functions for measuring objects, determining position and monitoring areas. The followings provide a brief description on its

operating principle, functionality and versatility.

A3.1 Operating Principle

The LMS 200 operates by measuring the time of flight of light pulses. A pulsed laser beam is transmitted. If it meets an object it is reflected and the reflection is registered in the scanner's receiver. The time between emission and reception of the impulse is directly proportional to the distance between the scanner and the object (time of flight). An internal rotating mirror deflects the pulsed laser beam so that a fan-shaped scan is made of the surrounding area. The contour of the target area is determined from the sequence of impulses received. The measurement data is available for further evaluation in real time via a serial interface.

A3.2 Functionality

The method of measurement together with its high degree of functionality makes the LMS 200 enormously practical. Its principle characteristics and advantages include:

- High measurement resolution (10 mm resolution)
- Contact-free measurement
- Active system, no illumination of target objects necessary
- Target objects require no reflectors or markings
- Target objects require no special reflective properties
- Range: Max. 80m, configurable
- Maximum scanning angle: 180°, configurable
- Angular resolution: 0.5°
- High scanning frequency (up to 75 Hz)
- Transfer of measurement data in real time (13ms response time)
- Laser protection class: 1 (eye safe)
- System error: typical $\pm 15\text{mm}$
- Interface serial: RS-422 or RS-232 switchable
- Dimension: 155 x 185 x 156 mm (W x H x D)
- Operating temperature: 0°C to 50°C
- Power consumption: Max. 17.5W
- Weight: 4.5kg

A3.3 Versatility

A decisive advantage of this method of measurement is that any objects, regardless of their shape, colour or surface structure, can be measured without any contact. Even if target objects have different reflective properties, up to a range of 20m this can be ignored.

A4 Sony EVI-D30 Video Camera

EVI-D30 is a pan/tilt/zoom (PTZ) video camera with highly sophisticated image processing technology, which enables target objects to be recognised through its auto tracking and motion detection features. The EVI-D30 offers crisp image quality, high speed, wide range PTZ, auto-focus and 12X zoom, remote commander controller and AC adapter. The Sony EVI-D30 is used together with the BT848 framegrabber (discuss below) for visual surveillance and colour segmentation for intrusion detection and intruder pursuit. The specifications of this camera are:

- Image Sensor: 1/3" Color CCD
- Effective Pixels: 768 [Horizontal (H)] x 492 [Vertical (V)]
- Resolution: 460TV lines (H) and 350TV lines (V)
- Lens: X12 power zoom, $\sim f=5.4$ to 64.8mm, F1.8 to F2.7
- H. angle of view: 4.4 ° (telephoto end) $\sim$ 48.8 ° (wide angle end)
- V. angle of view: 3.2 ° (telephoto end) $\sim$ 37.6 ° (wide angle end)
- Pan/Tilt: H. $\pm 100^\circ$ (Max. speed 80°/sec), V. $\pm 25^\circ$ (Max. speed 50°/sec)
- Video output: RCA pin jack, 1Vp-p, 75ohm unbalanced

- Control terminal: RS232C, 8 pin mini DIN, 9600bps, Data 8 bit, Stop 1 bit
- Dimension: 142mm x 109mm x 164mm (W x H x D)
- Operating temperature: 0°C to 40°C
- Power consumption: 10.5w
- Weight: 1200g

A4.1 BT848 framegrabber

The framegrabber is produced by Hauppauge, and is based on Brook Tree BT848 chip. The v4l (video - for (4) - Linux) protocol is used to fetch the digitized image from the framegrabber to the robot computer memory. Due to the computation complexity of image processing, and real-time navigation requirement, the digitized image size was set to 160x120 in this implementation. The processing rate is 3-4 frames/second, depending on the complexity of the image processing.

A5 KVH-C100 Compass

The KVH C100 produced by KVH Industries Inc. is a complete stand-alone sensor subsystem that outputs extremely accurate heading data in any of six user-selectable digital or analog formats. The C100 microprocessor-controlled fluxgate compass consists of a detachable toroidal fluxgate sensor element and a small electronics board. The ring core is housed in a Lexan cylinder and free floating in an inert fluid to keep it horizontal with respect to the earth. Windings surround the Lexan cylindrical housing, electrically driving the ring core into saturation and measuring the amplitude of induced pulses which are proportional to the earth's magnetic field. This data is then sent to the microprocessor which compensates for the hard and soft iron magnetic interference of the host platform. The resulting output is translated into extremely accurate heading data. The specifications of this compass are:

- Accuracy: $\pm 0.5^\circ$
- Repeatability: $\pm 0.2^\circ$
- Resolution: 0.1°
- Response Time: 0.1 to 24 seconds (user selectable)
- Interface serial: RS-232
- Dimension: 46 x 28 x 114 mm (W x H x D)
- Operating temperature: -40°C to $+65^\circ\text{C}$
- Weight: 64g

A6 Odometry

The drive shafts of ATRV-JrTM are equipped with optical encoders. Given initial position and heading, the robot's current position and heading can be calculated by integrating the optical encoders' readings. This technique is usually called dead reckoning. For short distances, the odometry reading is reasonably accurate. However, due to the slippage between the wheel and the ground, the odometer's error accumulates over long distance and this error is unbounded.

A7 Crossbow Attitude & Heading Reference System (AHRS) 400CA

The AHRS is a nine-axis measurement system that combines linear accelerometers, rotational rate sensors, and magnetometers. The AHRS uses the 3-axis accelerometer and 3-axis rate sensor to make a complete measurement of the dynamics of a system (i.e. the ATRV-JrTM). The addition of a 3-axis magnetometer also allows the AHRS to make a true measurement of magnetic heading. The AHRS is the solid-state equivalent of a vertical gyro/artificial horizon display combined with a directional gyro. For this research, the AHRS is only employed for calculating the ATRV-JrTM orientation (via integrating the yaw rate) and use the calculated orientation for reducing the (x, y) position errors derived from the odometry discuss above. The specifications of this sensor are:

- Update Rate: $>60\text{Hz}$
- Heading: $\pm 180^\circ$ - Resolution: $<0.1^\circ$ rms
- Attitude: $\pm 180^\circ$ (Roll), $\pm 90^\circ$ (Pitch) - Resolution: $<0.1^\circ$
- Angular rate: $\pm 100^\circ/\text{sec}$ (Roll, Pitch, Yaw) - Resolution: $<0.0025^\circ/\text{sec}$

- Acceleration: $\pm 2G$ - Resolution: 0.25mg
- Operating temperature: -40°C to $+71^{\circ}\text{C}$
- Interface serial: RS-232
- Dimension: 9.53 x 10.41 x 7.62cm (W x H x D)
- Power consumption: 4w
- Weight: 0.64kg

A8 BreezeNET PRO.11 Wireless Modems

The BreezeNET PRO.11 modem produced by BreezeCOM Ltd. is an IEEE 802.11b-compliant wireless networking product designed to provide high performance wireless point-to-point and point-to-multipoint links. The products are designed for maximum performance and reliability. According to the IEEE 802.11b protocol standard, the data rates can vary up to 2 Megabits per second (Mbps). However BreezeCOM has incorporated a proprietary data rate of up to 3 Mbps in the BreezeNET PRO.11 modems. As the unit provides a connection for a detached antenna, the optimal antenna may be selected per installation. Two types of BreezeNET PRO.11 modems are used in this research infrastructure; an Access Point (AP-10) and a Station Adapter (SA-10). An AP-10 is a wireless hub that connects a wireless network to a wired network and establishes a wireless communication for the SA-10 (the SA-10 is mounted on the ATRV-Jr™). The range is approximately 1km for outdoor and 150m for office environment.

A9 Premier Wireless Video CS-220

Premier's wireless video CS-220 systems produced by Premier Wireless Inc. are designed to provide wireless distribution of full-motion color video signals. Standard NTSC or PAL video/audio signals can be transmitted on any one of four user selectable channels. The system operates in the 2400 to 2483 MHz frequency band and is FCC certified requiring no user license. The maximum transmission range (~300m) will be achieved when there is a clear line of sight between receive and transmit antennas.

References: Appendix A

- [1] IS Robotics, Inc., Real World Interface Division, www.isr.com, 32 Fitzgerald Dr., Jaffrey, NH 03452, USA. *ATRV-Jr All-Terrain Mobile Robot User's Guide*, Part Number: 3698; Rev. 5, 1999.
- [2] IS Robotics, Inc., Real World Interface Division, www.isr.com, 32 Fitzgerald Dr., Jaffrey, NH 03452, USA. *Mobility Robot Integration Software User's Guide*, Part Number: 2841; Rev. 5, 2002.
- [3] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi. A Stable Tracking Control Method for Autonomous Mobile Robot, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 384-389, ISSN 1050-4729. 1990.

Appendix B: Implementation Details

The following topics are discussed in this appendix:

- B1. Landmark learning
- B2. Visual attention operator
- B3. Monitoring of human control behaviours
- B4. Real-time obstacle avoidance scheme

B1 Landmark Learning

In this implementation, a learning system for a mobile robot is designed based on fuzzy control technique [1] to let the robot learn specified landmark. The landmarks in this context are any object in a specific location which can be detected by the robot sensors. Compass, laser range finder readings and image (via the vision system) are used as the fuzzy inputs. The learning system generates a set of fuzzy rules based on these sensory inputs for landmark detection. The inputs and output used are defined as follows. The input provided by the compass is the robot heading (in degree). Next, three types of inputs are provided by the laser range finder. They are average obstacle distance (in metre), closest obstacle distance (in metre), and direction of the closest obstacle (in degree). The average obstacle distance is computed from all the detected obstacles which relate to the robot's position. The distance and the direction of the closest obstacle give important features that the robot can recognise as a landmark for each location. The input provided by the vision system is the attention focus of the robot represented by a point (column, row) in an image (i.e. 160x120 resolutions). The approach used to compute the robot visual attention is discussed in Section B.2. Finally, landmarks are defined as fuzzy outputs using a sequence of number (1, 2, 3 ... n). The three main processes during the learning phase are *fuzzification*, *rules generation* and *defuzzification* described in the following sub-sections.

B1.1 Fuzzification

Divides the inputs and outputs spaces of the numerical data into fuzzy regions as follows:

The fuzzy inputs are as follows:

- Robot heading from compass in degree – 0° to 360°, using 9 Memberships Functions (MF)
- Laser average obstacle distance in metre – 0 to 6m, using 12 MF
- Laser closest obstacle distance in metre – 0 to 6m, using 12 MF
- Laser direction of the closest obstacle in degree – 0° to 180° , using 9 MF
- Column image of the robot attention focus in pixel – 0 to 160, using 8 MF
- Row image of the robot attention focus in pixel – 0 to 120, using 6 MF

The fuzzification (based on Gaussian function) of the six sensory inputs above are presented in Figure B1. As for the output, no need in this case as it is just “number”. Examples of output fuzzification of ATRV-Jr™ are the linear and angular speeds.

B1.2 Rules Generation

Based on the inputs, fuzzy rules are generated. The implementation is based on Wang & Mendel [2] approach of rules generation from numerical data. Using this approach, to resolve conflict among the generated rules, a degree is assigned to each generated rule. The rules will define the input-output pairs (stated control) which give the specified values of the inputs and the corresponding successful outputs. Here, the rules generated are “AND” rules, i.e., rules in which the conditions of the IF part must be met simultaneously in order for the result of the THEN part to occur. For the problems considered, i.e., generating fuzzy rules from the six numerical data above, only “AND” rules are required since the antecedents are different components of single input vectors.

B1.3 Defuzzification

This process determines a functional mapping from input space to output space based on the generated

rules. As there is only a single output landmark number (e.g. 1, 2, 3, ... n), the defuzzification is done using rule matching, i.e. while the robot is moving, a match fuzzy rule gives the output as the number which is represented as the landmark. Notes, defuzzification methods such as centroid, bisector of area, maximum defuzzification, etc. are not needed for this case.

Figure B1: Sensory inputs membership functions for landmark learning

B2 Visual Attention Operator

The visual attention operator used in this research is a saliency-based visual attention method proposed by Laurent Itti [3]. This model (Figure B2) is built on a biologically plausible architecture, proposed by Koch & Ullman [4] and at the basis of several models Milanese et al. [5] and Baluja & Pomerleau [6]. It is related to the so-called “feature integration theory,” explaining human visual search strategies. Visual input is first decomposed into a set of topographic feature maps. Different spatial locations then compete for saliency within each map, such that only locations which locally stand out from their surround can persist. All feature maps feed, in a purely bottom-up manner, into a master “saliency map,” which topographically codes for local conspicuity over the entire visual scene. In primates, such a map is believed to be located in the posterior parietal cortex (Leventhal [8]) as well as in the various visual maps in the pulvinar nuclei of the thalamus (Arkin [9]). The model’s saliency map is endowed with internal dynamics which generate attentional shifts. This model consequently represents a complete account of bottom-up saliency and does not require any top-down guidance to shift attention. This provides a massively parallel method for the fast selection of a small number of interesting image locations to be analysed by more complex and time-consuming object-recognition processes.

Figure B2: General architecture of the visual attention model (adapted from Itti et al. [3])

As shown in Figure B2, the input for this model is provided in the form of static color images (for this research the input image is digitized at 160x120 resolutions). Nine spatial scales are created using dyadic Gaussian pyramids (Greenspan et al. [7]). Each feature is computed by a set of linear “center-surround” operations akin to visual receptive fields (Figure B2): Typical visual neurons are most sensitive in a small region of the visual space (the center), while stimuli presented in a broader, weaker antagonistic region concentric with the center (the surround) inhibit the neuronal response. Such architecture that is sensitive to local spatial discontinuities is particularly well-suited to detecting locations which stand out from their surround and is a general computational principle in the retina, lateral geniculate nucleus, and primary visual cortex (Leventhal [8]).

Center-surround is implemented in the model as the difference between fine and coarse scales: The center is a pixel at scale $c \in \{2, 3, 4\}$, and the surround is the corresponding pixel at scale $s = c + d$, with $d \in \{3, 4\}$. The across-scale difference between two maps is obtained by interpolation to the finer scale and point-by-point subtraction. Using several scales not only for c but also for $d = s - c$ yields truly multi-scale feature extraction, by including different size ratios between the center and surround regions. In total, there are 42 feature maps are computed (Figure B.2): 6 for intensity, 12 for colour, and 24 for orientation. These feature maps are combined into three “conspicuity maps”, intensity, colour, and orientation the last scale of the saliency map. A winner-take-all neural network finds the maximum of the saliency map. This maximum defines the most salient image location, to which the focus of attention should be directed. This salient image location is used as the input for landmark learning described in Section B1.

B3 Monitoring of Human Control Behaviours

Robot must monitor human control behaviours so as to provide appropriate assistance to his/her during teleoperation. In this implementation, for real-time continuous human control input, a joystick is employed. The following sub-section first provides a discussion of how this joystick can be used to control the ATRV-Jr™ follow by discussing how robot can provide appropriate assistance to the human based on the control input from the joystick.

B3.1 Human Control Input: Joystick

The joystick is an ancillary input device for applications that provide alternatives to using the keyboard and mouse. The joystick provides positional information within a coordinate system that has absolute maximum and minimum values in each axis of movement. In this research, a simple 2-axis joystick with *force feedback* is used to control the robot (Figure B3). The three main measurements are the x -position (for rotation, in rad/s), y -position (for translation, in m/s) and buttons (for control execution and interaction mode changing).

Figure B3: The region mapping of a joystick in controlling the ATRV-Jr™.

As shown in Figure B3, the boundary region that is accessible by joystick is partitioned as distinct region dedicated to specific motion control. For x -position, 0 unit represents extreme left (-1.7 rad/s) and 65535 is extreme right (1.7 rad/s) and likewise for y -position, but it represents forward (1.7 m/s) and backward (-1.7 m/s). The exact centre position is 32767 units for x -position and y -position (i.e. 0 rad/s, 0 m/s). The centre position of the joystick generates a stop command to the robot. When the joystick is

released or not being used, it is not really stand still at the exact centre position but off-centre most of the time. To eradicate this error, a buffer region is created as a tolerance area for joystick resting position. In this implementation, the buffer region is set in range of ± 3500 unit from the exact centre position. This mean about 11% of the total useable data will be voided as buffer area. The basis for this is through trial and error experiments with users, where users actually treat their hand resting position (while gripping on the joystick) as the centre position of the joystick instead of the actual centre position of the joystick. Thus, a larger buffer area needs to accommodate this communication gap between the human and the control interface. It is also impractical and less effective to correct or rectify this error because the natural hand resting position of their hand is a subconscious action.

As the joystick is implemented for rate control, to obtain the direction, fuzzy logic is employed to process the translation and rotation speed (i.e. input). The purpose is for tracking human joystick control behaviours so as to let robot provides appropriate assistance to human. The fuzzy output is the desired joystick direction. The fuzzification (based on Gaussian function) of the fuzzy inputs - translation (y-axis of the joystick) and rotation (x-axis of the joystick) speed are presented in Figure B4 and the output - direction is presented in Figure B5. Both axes used 35 MF and the joystick direction used 9 MF. As in landmark learning (Section B1), the rules employed are “AND” rules. For example, “IF y-axis = 0.7 AND x-axis = 0 THEN direction = 0°”. The defuzzification is based on Center of Gravity (COG) method [1].

Figure B4: Joystick x and y axes membership functions (input)

Figure B5: Joystick direction membership functions (output)

B3.2 Robot Behaviour Encoding

A functional mapping is required to encode human control behavioural response (i.e. the joystick control direction discussed above) from the stimulus plane (i.e. the joystick plane as shown in Figure B.3 and the robot perception of the task environment) to the robot motor plane (i.e. drive system, see Appendix A). The robot motor plane needs two parameters, the strength (magnitude of the response, i.e. the speed) and the orientation (direction of the action). In accordance to Arkin [9], robot behaviour can be expressed as (S, R, β) , where:

- S : Denotes the stimulus domain of all interpretable stimuli. A stimuli s (where $s \in S$) is represented by (p, λ) , where p is the class of robot perception (in this implementation, this including both perception of the human control behaviours and the robot perception of its task environment) with a threshold value, τ , above which a response is generated and λ is its strength.
- R : Denotes the range of possible responses. A response, r (where $r \in R$) can be represented by a six-dimensional vector $[x, y, z, \theta, \phi, \psi]$, where (x, y, z) are the three translation degrees of freedom (DOF) and $(\theta$ for roll, ϕ for pitch, ψ for yaw) are the three rotational DOF. In this implementation, as the mobile robot only move on horizontal plane, there are only three DOF, i.e. x, y and ψ .
- β : Denotes the mapping of the stimulus domain to the response domain ($\beta(s) \rightarrow r$). The mapping function generates response only when $\lambda > \tau$. The behavioural mapping, β , of stimuli onto responses fall into three general categories:
 - *Null*: The stimulus produces no motor response. This may due to human stop sending command to the robot (buffer region, Figure B.3) or robot the stop the operation (e.g. *emergency stop*) when its sense that there are danger (e.g. colliding) with obstacles in its surrounding.
 - *Discrete*: The stimulus produces a response from an enumerable set of prescribed choices (i.e. turn-right, go-forward, stop etc.). Normally, discrete encoding involves the use of “rule-based system”, where β is represented as a collection of IF-THEN rules. I.e. IF *Obstacles-ahead* (stimuli) THEN *decelerate* (response). This form of mapping is used for implementing the collision prevention behaviour. This is discussed in Section B3.3.
 - *Continuous*: The stimulus domain produces a motor response that is continuous over R 's range (i.e. specific stimuli s is mapped into an infinite set of response encoding by β . This mapping method is used in the implementation of the real-time obstacle avoidance scheme for the correction of human control behaviours in exclusive shared control modes described in Section B4.

B3.3 Collision Prevention

The collision prevention behaviour is designed for safe deceleration when a collision is about to occur. The profile (polynomial) for safe deceleration is presented in Figure B6. This behaviour is employed in both manual and exclusive shared mode. This behaviour employed a discrete encoding scheme that involves the use of a rule-based system for decision making as depicted in Table B1.

Figure B6: Deceleration profile for collision prevention behaviour.

Table B1: Rules for collision prevention behaviour

<u>Robot Sensors States</u>	<u>Human Control Behaviours</u>	<u>Allowable Output States</u>
<p>Sonar and laser are grouped as follows for accessing the human control behaviour.</p> <p> S_1 – Front S_2 – Front-Left S_3 – Left S_4 – Back S_5 – Right S_6 – Front-Right </p> <p>State 0: OFF State 1: ON</p>	<p> SP – Stop F – Forward Translation FL – Combination of Forward Translation and Left Rotation L – Left Rotation BL – Combination of Backward Translation and Left Rotation B – Backward Translation BR – Combination of Backward Translation and Right Rotation R – Right Rotation FR – Combination of Forward Translation and Right Rotation </p>	
$S_10, S_20, S_30, S_40, S_50, S_60 \rightarrow A_0$	SP, F, FL, L, BL, B, BR, R, FR	SP, F, FL, L, BL, B, BR, R, FR
$S_10, S_20, S_30, S_40, S_50, S_61 \rightarrow A_1$	SP, F, FL, L, BL, B, BR, R, FR	SP, F, FL, L, BL, B, BR, R, FR
$S_10, S_20, S_30, S_40, S_51, S_60 \rightarrow A_2$	SP, F, FL, L, BL, B, BR, R, FR	SP, F, FL, L, BL, B, BR, FR
$S_10, S_20, S_30, S_40, S_51, S_61 \rightarrow A_3$	SP, F, FL, L, BL, B, BR, R, FR	SP, F, FL, L, BL, B, BR
.....		
$S_11, S_21, S_31, S_41, S_51, S_61 \rightarrow A_{63}$	SP, F, FL, L, BL, B, BR, R, FR	SP

B.4 Real-Time Obstacle Avoidance Scheme

To achieve safe and seamless navigation, the adopted obstacle avoidance approach must not only facilitate the detection of an obstacle but also must allow the robot to negotiate with the obstacle, which requires some kind of quantitative measurements concerning the obstacle's dimensions (Borenstein & Koren [10]). Some of the well-known real-time obstacle avoidance approaches to this includes, *edge or boundary detection*, *certainty grid*, *artificial potential field* and *vector field histogram*. They are described in Section B4.1. The obstacle avoidance scheme employed in this research is described in Section B4.2.

B4.1 Obstacle Avoidance Approaches

B4.1.1 Edge or Boundary Detection

A classical obstacle avoidance approach employ in many mobile robot applications (Cooke [11], Chatterg [12]). Here, the algorithm tries to determine the position of the vertical edges of the obstacle and consequently attempts to steer the robot around either edge. The line connecting the two edges is considered to represent one of the obstacle's boundaries. A disadvantage of this approach is the need of the robot to stop in front of an obstacle in order to allow for a more accurate measurement. Furthermore, this approach is sensitivity to sensor accuracy. Unfortunately, ultrasonic sensors, which are mostly used in mobile robot applications, offer many shortcomings in this respect. This includes [10]: poor directionality to determine the position of the obstacle edge; frequent miss-readings causes by ultrasonic noise; and specular reflections. To overcome these effects, one widely used approach is to use certainty grid, proposed by Elfes [13].

B4.1.2 Certainty Grid

This approach uses a probabilistic representation of obstacles in a 2D grid-type world model, called certainty grid (or occupancy grid) (Elfes [13]). With the certainty grid world model, the robot's work area is represented by a 2D array of square elements, denoted as cells. Each cell contains a *certainty value* (CV) that indicates the measure of confidence that an obstacle exists within the cell area. Consequently, CVs are updated by a probabilistic function, which is empirically formulated to take into account the characteristics of a given type of sensor. Apart from the accommodation of inaccurate sensor data such as from ultrasonic sensor (above), another advantage of this approach is it allows adding and retrieving data on the fly and enables easy integration of multiple sensors (Moravec [14]). However, a disadvantage with this approach is the need of the robot to stop for scanning when it navigates to a new location so as to update the probabilistic function.

B4.1.3 Artificial Potential Field

This approach is based on idea of imaginary forces acting on the robot. It is first proposed by Andrews and Hogan [15] and Khatib [16] for robotics manipulator obstacle avoidance. The working principles are as follows: obstacle exerts repulsive forces onto the robot, while the target applies an attraction force to the robot. Following this, the sum of all forces determines the subsequent direction and speed of travel. According to Koren and Borenstein [17], the main advantage of this approach is it is simple and can be implement quickly with acceptable results, as compared to edge detection and certainty grid approaches. Consequently, this approach is also adopted by Brooks [18] and Arkin [19] in mobile robotics for obstacle avoidance. Although their implementations do not require the robot to stop either for accurate measurement or scanning but it only guaranteed collisions avoidance with static obstacle (Newman & Hogan [20]). In addition, Koren and Borenstein [17] found that using this approach for mobile robot obstacle avoidance has four severe limitations:

- (i) Trap situations due to local minima when the robot runs into a dead end (e.g., inside a U-shaped obstacle);
- (ii) No passage between closely spaced obstacles;
- (iii) Unstable motion in the presence of obstacles; and
- (iv) Oscillations in narrow passages cause by repulsive forces simultaneously from opposite sides.

B4.1.4 Vector Field Histogram (VFH)

This approach proposed by Borenstein and Koren [21], is developed based on the concept of the artificial potential field and certainty grid. This combination produces a powerful and robust obstacle avoidance scheme for mobile robots. For example, Borenstein and Koren [22] integrated this approach into a mobile robot with 24 sonar sensors onboard, and could perform real-time obstacle avoidance with an average speed of 0.54 m/s without stopping. The VFH utilises a two-dimensional Cartesian Histogram Grid, derived from the concept of certainty grid. Like the certainty grid, each cell in the histogram grid holds a CV that represents the confidence in the existence of an obstacle at that location. However, unlike the certainty grid approach that requires the robot to stop and update every single cell that are affected by a range reading, the histogram grid only update one cell for each range reading while the robot is moving. Therefore, an accurate obstacle representation can obtain without require the robot to stop for scanning as compare to using certainty grid. Given this, the two-dimensional Histogram Grid is converted into a one-dimensional Polar Histogram, which represents the polar obstacle density around the robot. The robot then performs obstacle avoidance by moving towards the sector that contains low obstacle density. This is similar to that of the artificial potential field concept. However, this approach allows a robot to navigate through a narrow passage or close to dynamic obstacles without oscillations as compared to the artificial potential field approach.

Over the years, the VFH approach is further refined by Borenstein's research group (e.g. [23, 24]). This includes the reduction of some of the parameter tuning of the original VFH approach by explicitly compensating for the robot width, provides a better approximation of the mobile robot trajectory, to name a few. Due to the ability to perform robust real-time obstacle avoidance as compared to the other three approaches discussed above, this approach is adopted. This is further described in Section B4.2.

B4.2 Implementation of the Obstacle Avoidance

The obstacle avoidance algorithm implemented here is based on the concept of VFH (described in Section B4.1.4) but augmented with the capability of decision making using the immediate sensor readings. One of the reasons for using this approach is to improve the reliability of the obstacle avoidance algorithm through the use of "redundancy". This is necessary because the use of occupancy grids requires an accurate localisation system to maintain the world model (Borenstein & Koren [22]). If it is inaccurate, the occupancy grids may give wrong representation of obstacles position and thus incorrect calculation of the obstacles forces. To overcome this, current sensor readings are used to verify whether the calculation of the obstacle forces from occupancy grids representation (Figure B7(a)) is correct. To facilitate this, the sonar and laser sensors are grouped in accordance to Figure B7(b). Each group of sensors produces a vector that determines the relative distance of obstacles to the centre of the robot.

The verification of the sensor readings is only activated if any of the current sensor readings in each of the sensor groups is lower than one metre. If their calculations do not tally (i.e. different steering direction) with each other, the following steps will be performed:

1. Stop the robot.
2. Recompute the occupancy grids based on the current robot position and orientation.
3. Calculate the new heading of the robot. The calculation is based on the obstacle forces from the occupancy grids and the desired heading by the human.
4. Robot turns according to the calculate heading and proceeds at a lower speed.
5. Increase to a higher speed when all the obstacles are clear.

The advantage of using sensor groupings is that it allows for a much quicker reaction time to objects in the environment and requires no previous knowledge of the environment. However, as compared to representation using occupancy grids, this sensors representation is oversimplified and has difficulty in performing obstacle avoidance in a cluttered environment such as the Robotic Research Centre (RRC). This is because this cluttered environment contained different types of obstacles (e.g. chairs and table of different size, cables from the computer work bench, sharp edges, etc.). When the robot reacts to these obstacles, unstable and unpredictable motion is produced. Hence, it is not employed as the main obstacle avoidance routine.

Figure B7: Robot front perception representation and sensors grouping

Another reason of using the current sensor readings is to facilitate coordination. This is important because the VFH approach is not a reactive methodology as it requires maintaining a world model. Therefore, it is difficult to integrate with other behavioural units (e.g. collision prevention, stay on path, etc.) which use reactive control that is “representation-free” (Section 2.3.2). However, the advantage of using the occupancy grid representation is that it facilitates map-building. Current implementation does not build a global map (Section 5.1.3), but rather it maintains a short-term memory of the robot front perception (6m x 3m occupancy grids) based on the sonar and laser sensors (Figure B7(a)). The purpose of this local map is to facilitate reactive path planning (Section 5.1.4).

References: Appendix B

- [1] L. A. Zadeh. Fuzzy Logic, Neural Networks, and Soft Computing, *Communications of the ACM*, Vol. 37, No. 3, pp. 77-84, ISSN 0001-0782, March, 1994.
- [2] L.-X. Wang and J.M. Mendel. Generating Fuzzy Rules by Learning from Examples, *Proceedings of the IEEE International Symposium on Intelligent Control*, pp. 263 – 268, August 13-15, 1991.
- [3] L. Itti, C. Koch, and E. Niebur. A Model of Saliency-Based Visual Attention for Scene Analysis, *IEEE Transactions on Pattern analysis and Machine Intelligence*, Vol. 20, No. 11, pp. 1254-1259, ISSN 0018-9340, November, 1998.
- [4] C. Koch and S. Ullman. Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry, *Human Neurobiology*, Vol. 4, pp. 219–227, ISSN 0721-9075, 1985.
- [5] R. Milanese, S. Gil, and T. Pun. Attentive Mechanisms for Dynamic and Static Scene Analysis, *Optical Engineering*, Vol. 34, No. 8, pp. 2,428–2,434, ISSN 0892-354X, August, 1995
- [6] S. Baluja and D. A. Pomerleau. Expectation-Based Selective Attention for Visual Monitoring and Control of a Robot Vehicle, *Robotics and Autonomous Systems*, Vol. 22, No. 3-4, pp. 329–344, ISSN 0921-8890, December, 1997.
- [7] H. Greenspan, S. Belongie, R. Goodman, P. Perona, S. Rakshit, and C. H. Anderson. Overcomplete Steerable Pyramid Filters and Rotation Invariance,” *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 222-228, ISBN 0818656751, Seattle, Wash., USA, June 21-23, 1994.
- [8] A. G. Leventhal. *The Neural Basis of Visual Function: Vision and Visual Dysfunction*, Vol. 4, CRC Press, ISBN 0849375045, Boca Raton, Fla, USA, 1991.
- [9] R. C. Arkin. *Behavior-Based Robotics*. MIT Press, ISBN 0262011654, Cambridge, 1998.
- [10] J. Borenstein and Y. Koren. Real-time Obstacle Avoidance for Fast Mobile Robots, *IEEE*

- Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 5, pp. 1179-1187, ISSN 1083-4427, September/October, 1989.
- [11] R. A. Cooke. Microcomputer Control of Free Ranging Robots. *Proceedings of the 13th International Symposium on Industrial Robots and Robots*, pp. 13.109-13.120, Chicago, Ill., USA, April, 1983.
 - [12] R. Chattergy. Some Heuristics for the Navigation of a Robot, *International Journal of Robotics Research*, Vol. 4, No. 1, pp. 59-66, ISSN 0278-3649, MIT Press, 1985.
 - [13] A. Elfes. Sonar-Based Mapping and Navigation System, *Technical Report*, Carnegie-Mellon University, The Robotics Institute, pp. 25-30, USA, 1985.
 - [14] H. P. Moravec. Sensor Fusion in Certainty Grids for Mobile Robots, *AI Magazine*, pp. 61-74, ISSN 0738-4602, 1988.
 - [15] J. R. Andrews and N. Hogan. Impedance Control as a Framework for Implementing Obstacle Avoidance in a Manipulator, *In Control of Manufacturing Processes and Robotic Systems*, Hardt, D. E. & Book, W. (Eds.), pp. 243-251, ISBN 0854961461, ASME, Boston, USA, 1983.
 - [16] O. Khatib. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *The International Journal of Robotics Research*, Vol. 5, No. 1, pp. 90-98, ISSN 0278-3649, MIT Press, 1986.
 - [17] Y. Koren and J. Borenstein. Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1398-1404, ISSN 1050-4729, 1991.
 - [18] R. A. Brooks. A Robust Layered Control System for a Mobile Robot, *IEEE Journal of Robotics and Automation*, Vol. RA-2, pp. 14-23, ISSN 0882-4967, April 1986.
 - [19] R. C. Arkin. Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behaviour, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 264-271, ISSN 1050-4729, Raleigh, NC, USA, 1987.
 - [20] W. S. Newman and N. Hogan. High Speed Robot Control and Obstacle Avoidance Using Dynamic Potential Functions, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 12-24, ISSN 1050-4729, 1987.
 - [21] J. Borenstein and Y. Koren. Real-Time Obstacle Avoidance for Fast Mobile Robots in Cluttered Environments, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 572-577, ISSN 1050-4729, 1990.
 - [22] J. Borenstein and Y. Koren. The Vector Field Histogram - Fast Obstacle-Avoidance for Mobile Robots, *IEEE Journal of Robotics and Automation*, Vol. 7, No. 3, pp. 278-288, ISSN 0882-4967, June, 1991.
 - [23] I. Ulrich and J. Borenstein. VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1572-1577, ISSN 1050-4729, Leuven, Belgium, May 16-21, 1998.
 - [24] I. Ulrich and J. Borenstein. VFH*: Local Obstacle Avoidance with Look-ahead Verification, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2505-2511, ISSN 1050-4729, San Francisco, CA, April 24-28, 2000.

Appendix C: Questionnaire

For the experimental evaluation on robot assists human, a subjective rating is employed to determine how participants feel about the evaluation, i.e. user satisfaction (Scholtz & Bahrami [1]). This is important because it allows participants to submit their analysis of the control interface and interaction modes, as well as their opinions. A widely adopted approach of measuring user satisfaction is the use of the Likert scale (Nielsen [2]), a one dimensional scaling method. In this method, participants are asked to express whether they agree or disagree to the set of questions in a “point” scale after they have completed the evaluation. There is a variety of possible response scales (e.g., 0-to-4, 1-to-5, 1-to-7, 1-to-9, etc.). All of these odd-numbered scales have a middle value labeled as *Neutral* or *Undecided*. Here, a 1-to-9 response scales is employed, with 1 for the worst and 9, the best. In this context, the participant is asked to decide whether they choose more towards the worst end of the scale or best end of the scale for each question. Each degree of agreement is given a numerical value ranging from 1 to 9. Thus, a total numerical value can be calculated from all the responses. Although it is a subjective rating of the participant, an average of the total numerical value provides a reasonable quantitative measure of the usability of the control interface and each interaction modes.

The design of the questionnaire for this experiment comprises of 14 questions and is divided into four groups:

1. *Overall reaction to the control interface*: This measure the general response of the participant to the generic human control interface employed throughout the evaluation. This section contains four sets of questions listed in Table C1. The purpose is to access whether the control interface used in the experiment is sufficient to provide effective HRI.

Table C1: Overall reaction to the control interface questions

Questions	Bad Rating	Rating	Good Rating
1. How easy was it to use?	Difficult	1-to-9	Easy
2. Information organisation?	Confusing	1-to-9	Very clear
3. Was the control functions easily remembered?	Difficult	1-to-9	Easy
4. Was the interface responsive?	Slow	1-to-9	Fast

2. *Overall reaction to the manual mode* (i.e. without robot assistance): This measures the response of driving the robot via the manual mode. This section contains two sets of questions listed in Table C2.

Table C2: Overall reaction to manual mode questions

Questions	Bad Rating	Rating	Good Rating
1. How easy was it to drive the robot?	Difficult	1-to-9	Easy
2. Learning to operate?	Difficult	1-to-9	Easy

3. *Overall reaction to the exclusive shared mode* (i.e. with robot assistance): This measures the response of driving the robot via the exclusive shared mode. This section contains four sets of questions listed in Table C3. The first and second questions are for comparing with the manual mode and the adaptive interaction modes. The third and fourth questions are for determining whether the assistance provided by the robot is useful.

Table C3: Overall reaction to the exclusive shared mode questions

Questions	Bad Rating	Rating	Good Rating
1. How easy was it to drive the robot?	Difficult	1-to-9	Easy
2. Learning to operate?	Difficult	1-to-9	Easy
3. Did you understand how the robot assists you?	Confusing	1-to-9	Clear
4. Was the robot assistive?	Not assistive	1-to-9	Assistive

4. *Overall reaction to the adaptive interaction modes:* This measures the response of driving the robot through the adaptive interaction modes. This section contains four sets of questions listed in Table C4. The first and second questions are for comparing with the manual and the exclusive shared modes. The third and fourth questions are for determining the usability of the adaptive interaction modes.

Table C4: Overall reaction to the adaptive interaction modes varying questions

Questions	Bad Rating	Rating	Good Rating
1. How easy was it to drive the robot?	Difficult	1-9	Easy
2. Learning to operate?	Difficult	1-9	Easy
3. How easy was it to perform the mode transition?	Difficult	1-9	Easy
4. Did the robot react to the mode transition seamlessly?	Poor	1-9	Good

The questionnaire also allows the participants to comments on the most negative and positive aspects of each interaction mode. The actual questionnaire forms can be found in page C3 to C5.

References: Appendix C

- [1] J. Scholtz and S. Bahrami. Human-Robot Interaction: Development of an Evaluation Methodology for the Bystander Role of Interaction, *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, Vol. 4, pp. 3212 – 3217, October 5-8, 2003.
- [2] J. Nielsen. *Usability Engineering*, AP Professional Press, ISBN 012184050, Boston, USA, 1993.

Nanyang Technological University
Robotics Research Centre

Pre-Test Questionnaire/Survey

Ref: _____

General

Age: _____ Gender: ☐ Male ☐ Female

Highest Education Level

☐ Primary Sch ☐ Secondary Sch ☐ Pre-University
☐ Bachelor's ☐ Post Graduate

Driving experience

Do you have a driving license? ☐ Yes ☐ No

If yes,

for how long have you had the license: _____

how often do you drive?

☐ Daily ☐ Weekly ☐ Monthly ☐ Rarely ☐ Never

Video/computer Game Experience

How often do you play video/computer games?

☐ Daily ☐ Weekly ☐ Monthly ☐ Rarely ☐ Never

If you do play games, how often do you play games or driving simulation games using joystick?

☐ Daily ☐ Weekly ☐ Monthly ☐ Rarely ☐ Never

Remote Driving Experience

Have you ever remotely driven a vehicle (e.g. remote control car, robot)?

☐ Yes ☐ No If yes, describe: _____

If yes, how many times? ☐ Once ☐ Several ☐ Often

Nanyang Technological University
Robotics Research Centre

Post-Test Questionnaire/Survey

Ref: _____

Overall reaction to the control interface		1	2	3	4	5	6	7	8	9	
5. How easy was it to use?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
6. Information organisation?	Confusing	☐	☐	☐	☐	☐	☐	☐	☐	☐	Very clear
7. Was the control functions easily remembered?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
8. Was the interface responsive?	Slow	☐	☐	☐	☐	☐	☐	☐	☐	☐	Fast

Overall reaction to the manual mode		1	2	3	4	5	6	7	8	9	
3. How easy was it to drive the robot?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
4. Learning to operate?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy

Comments on the most **negative** aspect(s) of using **manual mode**:

Comments on the most **positive** aspect(s) of using **manual mode**:

Overall reaction to the robot-assisted mode		1	2	3	4	5	6	7	8	9	
5. How easy was it to drive the robot?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
6. Learning to operate?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
7. Did you understand how the robot assists you?	Confusing	☐	☐	☐	☐	☐	☐	☐	☐	☐	Clear
8. Was the robot assistive?	Not assistive	☐	☐	☐	☐	☐	☐	☐	☐	☐	Assistive

Nanyang Technological University
Robotics Research Centre

Post-Test Questionnaire/Survey (Continue)

Ref: _____

Comments on the most **negative** aspect(s) of using **robot-assisted mode**:

Comments on the most **positive** aspect(s) of using **robot-assisted mode**:

Overall reaction to the adjustable mode		1	2	3	4	5	6	7	8	9	
5. How easy was it to drive the robot?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
6. Learning to operate?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
7. How easy was it to perform the mode transition?	Difficult	☐	☐	☐	☐	☐	☐	☐	☐	☐	Easy
8. Did the robot react to the mode transition seamlessly?	Poor	☐	☐	☐	☐	☐	☐	☐	☐	☐	Good

Comments on the most **negative** aspect(s) of using **adjustable mode**:

Comments on the most **positive** aspect(s) of using **adjustable mode**:

Thank you for your participation!

Appendix D: Experimental Results

Table D1: Performance measure: Task Completion Time (Seconds)

Participants	Experimental Conditions (A – Manual mode, B – Exclusive shared mode & C – Adaptive interaction modes)					
	Replicate I (Trial I)			Replicate II (Trial II)		
	1	2	3	1	2	3
1	A (222)	B (192)	C (167)	C (170)	B (185)	A (208)
2	C (141)	A (205)	B (188)	A (212)	C (132)	B (163)
3	B (185)	C (151)	A (259)	B (187)	A (210)	C (160)
4	C (127)	B (182)	A (185)	A (176)	B (161)	C (133)
5	A (182)	C (112)	B (170)	C (126)	A (179)	B (172)
6	B (176)	A (184)	C (158)	B (180)	C (162)	A (192)

Table D2: Performance measure: Number of Collisions

Participants	Experimental Conditions (A – Manual mode, B – Exclusive shared mode & C – Adaptive interaction modes)					
	Replicate I (Trial I)			Replicate II (Trial II)		
	1	2	3	1	2	3
1	A (5)	B (0)	C (3)	C (1)	B (0)	A (6)
2	C (2)	A (6)	B (0)	A (4)	C (0)	B (0)
3	B (0)	C (1)	A (2)	B (0)	A (4)	C (2)
4	C (1)	B (0)	A (2)	A (3)	B (0)	C (0)
5	A (3)	C (1)	B (0)	C (0)	A (1)	B (0)
6	B (0)	A (4)	C (1)	B (0)	C (1)	A (5)

Table D3: Performance measure: Number of Stops

Participants	Experimental Conditions (A – Manual mode, B – Exclusive shared mode & C – Adaptive interaction modes)					
	Replicate I (Trial I)			Replicate II (Trial II)		
	1	2	3	1	2	3
1	A (21)	B (0)	C (5)	C (0)	B (0)	A (22)
2	C (2)	A (22)	B (1)	A (26)	C (0)	B (0)
3	B (0)	C (3)	A (34)	B (0)	A (25)	C (5)
4	C (4)	B (0)	A (22)	A (17)	B (0)	C (5)
5	A (18)	C (0)	B (0)	C (1)	A (16)	B (0)
6	B (0)	A (19)	C (0)	B (0)	C (0)	A (21)

Table D4: Performance measure: Number of turn-on-spots

Participants	Experimental Conditions (A – Manual mode, B – Exclusive shared mode & C – Adaptive interaction modes)					
	Replicate I (Trial I)			Replicate II (Trial II)		
	1	2	3	1	2	3
1	A (19)	B (0)	C (0)	C (0)	B (0)	A (15)
2	C (0)	A (9)	B (0)	A (12)	C (0)	B (0)
3	B (0)	C (0)	A (17)	B (0)	A (18)	C (0)
4	C (0)	B (0)	A (14)	A (11)	B (0)	C (0)
5	A (10)	C (0)	B (0)	C (0)	A (9)	B (0)
6	B (0)	A (13)	C (0)	B (0)	C (0)	A (16)

Table D5: ANOVA table for number of collisions

Sources of Variation	Sum of Squares	Degree of Freedom	Mean Square	F	F _{0.05}
Types of interaction modes	89.39	2	44.70	46.56	3.37
Participants	11.89	5	2.38	2.48	2.59
Variation in run to run	0.39	2	0.20	0.21	3.37
Error	24.89	26	0.96		
Total	126.56	35			

Table D6: ANOVA table for number of stops

Sources of Variation	Sum of Squares	Degree of Freedom	Mean Square	F	F _{0.05}
Types of interaction modes	3496.22	2	1748.11	273.60	3.37
Participants	100.47	5	20.09	3.14	2.59
Variation in run to run	44.22	2	22.11	3.46	3.37
Error	166.06	26	6.39		
Total	3806.97	35			

Table D7: ANOVA table for number of turn-on-spots

Sources of Variation	Sum of Squares	Degree of Freedom	Mean Square	F	F _{0.05}
Types of interaction modes	1577.39	2	788.70	219.08	3.37
Participants	34.56	5	6.91	1.92	2.59
Variation in run to run	7.72	2	3.86	1.07	3.37
Error	93.56	26	3.60		
Total	1713.23	35			

Experimental Results

Appendix D

Table D8: Overall reaction to the control interface rating (with '1' the worst and '9' the best) obtained via the post-evaluation questionnaire

Questions	Participants						
	1	2	3	4	5	6	Average
1. How easy was it to use?	8	6	7	7	8	8	7.33
2. Information organisation?	7	9	7	7	7	8	7.50
3. Was the control functions easily remembered?	8	9	7	9	9	9	8.50
4. Was the interface responsive?	7	9	4	8	8	8	7.33

Table D9: Overall reaction to the manual mode rating (with '1' the worst and '9' the best) obtained via the post-evaluation questionnaire

Questions	Participants						
	1	2	3	4	5	6	Average
1. How easy was it to drive the robot?	6	7	3	8	2	5	5.17
2. Learning to operate?	6	7	4	8	4	7	6.00

Table D10: Overall reaction to the exclusive shared mode rating (with '1' the worst and '9' the best) obtained via the post-evaluation questionnaire

Questions	Participants						
	1	2	3	4	5	6	Average
1. How easy was it to drive the robot?	8	9	7	8	7	9	8.00
2. Learning to operate?	7	9	7	8	6	9	7.67
3. Did you understand how the robot assists you?	7	9	8	7	7	8	7.67
4. Was the robot assistive?	8	9	7	8	8	9	8.17

Table D11: Overall reaction to the adaptive interaction modes rating (with '1' the worst and '9' the best) obtained via the post-evaluation questionnaire

Questions	Participants						
	1	2	3	4	5	6	Average
1. How easy was it to drive the robot?	8	7	7	7	5	8	7.00
2. Learning to operate?	8	9	7	7	4	9	7.33
3. How easy was it to perform the mode transition?	8	6	7	8	3	9	6.83
4. Did the robot react to the mode transition seamlessly?	8	8	8	8	6	9	7.83

Table D12: ANOVA table for "How easy was it to drive the robot?" - For the comparison between the manual modes, exclusive shared mode & adaptive interaction modes

Sources of Variation	Sum of Squares	Degree of Freedom	Mean Square	F	F _{0.05}
Between Groups	24.78	2	12.39	5.04	3.68
Error	36.83	15	2.46		
Total	61.61	17			

Table D13: ANOVA table for “Learning to operate?” - For the comparison between the manual modes, exclusive shared mode & adaptive interaction modes

Sources of Variation	Sum of Squares	Degree of Freedom	Mean Square	F	F _{0.05}
Between Groups	9.33	2	4.67	1.81	3.68
Within-groups (error)	38.67	15	2.58		
Total	48	17			

Appendix E: Computer Vision Libraries

This appendix provides an overview of the two computer vision libraries employed in the development of the vision system. In this implementation, visual task such as object following and tracking are developed using a combination of colour segmentation and edge detection. They are selected due to the simplicity and suitability for real-time implementation. The colour segmentation and edge detection are implemented using Colour Machine Vision (CMVision) and Open Source Computer Vision (OpenCV) libraries described in Section E1 and E2 respectively.

E1 Colour Machine Vision Library

Colour Machine Vision (CMVision in short) library was written by James Bruce at CMU and is freely available under the GNU GPL: <http://www-2.cs.cmu.edu/~jbruce/cmvision>. It is a fast colour segmentation (or blob-finding) software library. This library uses YUV or fractional YRGB as robust color spaces for thresholding. This aim of this library is to accelerate low level vision for use in real-time applications where hardware acceleration is either too expensive or unavailable. It has been tested on a system that can perform bounded computation, full frame processing at camera frame rates. The system can track up to 32 colors at resolutions up to 640x480 and rates at 30 or 60Hz without specialised hardware. In addition, successful applications have been made using this library in the Sony Quadrupeds, the Minnow robots, the RoboCup F180 League, and tracking animals in real-time. As this library has proven its usefulness in real-time, it is selected to develop the vision system for the mobile robot.

E2 Open Source Computer Vision Library

Open Source Computer Vision (OpenCV in short) library is initiated by Intel® to provide a collection of C/C++ implementation of image processing and computer vision algorithms. To date, it is one of the most widely used libraries in computer vision. OpenCV is free for both non-commercial and commercial use and can be obtained from <http://www.sourceforge.net/projects/opencvlibrary>. This library is selected because it implements a wide variety of tools for image interpretation. It is compatible with Intel® Image Processing Library (IPL) that implements low-level operations on digital images. In spite of primitives such as binarisation, filtering, image statistics, pyramids, OpenCV is mostly a high-level library implementing algorithms for calibration techniques (Camera Calibration), feature detection (Feature) and tracking (Optical Flow), shape analysis (Geometry, Contour Processing), motion analysis (Motion Templates, Estimators), 3D reconstruction (View Morphing), object segmentation and recognition (Histogram, Embedded Hidden Markov Models, Eigen Objects). The essential feature of the library along with functionality and quality is performance. The algorithms are based on highly flexible data structures (Dynamic Data Structures) coupled with IPL data structures; more than a half of the functions have been assembler-optimized taking advantage of Intel® Architecture (Pentium® MMX™, Pentium® Pro, Pentium® III, Pentium® 4).