Quantitative Measures of Regret and Trust in Human-Robot Collaboration Systems

Zhanrui Liao
Clemson University, zhanrul@g.clemson.edu

Follow this and additional works at: http://tigerprints.clemson.edu/all_theses

Recommended Citation
http://tigerprints.clemson.edu/all_theses/2651

This Thesis is brought to you for free and open access by the Theses at TigerPrints. It has been accepted for inclusion in All Theses by an authorized administrator of TigerPrints. For more information, please contact awesole@clemson.edu.
QUANTITATIVE MEASURES OF REGRET AND TRUST IN HUMAN-ROBOT COLLABORATION SYSTEMS

A Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
Mechanical Engineering

by
Zhanrui Liao
May 2017

Accepted by:
Dr. Yue Wang, Committee Chair
Dr. John Wagner
Dr. Yongqiang Wang
Abstract

Human-robot collaboration (HRC) systems integrate the strengths of both humans and robots to improve the joint system performance. In this thesis, we focus on social human-robot interaction (sHRI) factors and in particular regret and trust. Humans experience regret during decision-making under uncertainty when they feel that a better result could be obtained if chosen differently. A framework to quantitatively measure regret is proposed in this thesis. We embed quantitative regret analysis into Bayesian sequential decision-making (BSD) algorithms for HRC shared vision tasks in both domain search and assembly tasks. The BSD method has been used for robot decision-making tasks, which however is proved to be very different from human decision-making patterns. Instead, regret theory qualitatively models human’s rational decision-making behaviors under uncertainty. Moreover, it has been shown that joint performance of a team will improve if all members share the same decision-making logic. Trust plays a critical role in determining the level of a human’s acceptance and hence utilization of a robot. A dynamic network based trust model combing the time series trust model is first implemented in a multi-robot motion planning task with a human-in-the-loop. However, in this model, the trust estimates for each robot is independent, which fails to model the correlative trust in multi-robot collaboration. To address this issue, the above model is extended to interdependent multi-robot Dynamic Bayesian Networks.
This thesis work is dedicated to all my loved ones. Thanks for your love and support.
Acknowledgments

I would like to thank my advisor, Dr. Yue (Sophie) Wang, for offering me the opportunities to join her research group-Interdisciplinary Intelligence Research (I^2R) Lab, helping me to learn knowledge and write papers. Dr. Wang not only guide and give me support on my research, but also give me advice on future career. I appreciate all her help and support. Also I would like to thank my committee members Dr. John Wagner and Dr. Yongqiang Wang for their advice.

Also, I would like to thank all the support offered by my group colleagues in the I^2R lab, Mr. Longsheng Jiang, Mr. Huanfei Zheng, Dr. Hamed Saeidi, Mr. Behzad Sadrfaridpour, etc. They not only help me design experiments and simulations, but bring a lot of fun to our group.

Last but not least, I would like to thank my family for their unconditional love and support.
# Table of Contents

Title Page ................................................................. i

Abstract ................................................................. ii

Dedication ............................................................... iii

Acknowledgments ....................................................... iv

List of Tables .......................................................... vi

List of Figures ........................................................ vii

1 Introduction .......................................................... 1
   1.1 Introduction of HRI and HRC Systems ....................... 1
   1.2 Regret ......................................................... 3
   1.3 Trust .......................................................... 5
   1.4 Structure ..................................................... 6

2 Quantitative Regret Measurements ............................... 8
   2.1 A Quantitative Measure of Regret in Decision-Making for Human-Robot Collaborative Search Tasks ................... 8
   2.2 Regret-based Allocation of Autonomy in Shared Detection Phenomena for Human-Robot Collaborative Assembly in Manufacturing .......... 27

3 Quantitative Trust Measurements .................................. 51
   3.1 A Quantitative Trust Measure in Multi-Robot Systems with a Human-in-the-Loop ................................................. 51
   3.2 An Interdependent Multi-Robot Trust Model .................. 67

4 Conclusions .......................................................... 82

Bibliography ........................................................... 84
## List of Tables

2.1 The binary choice question with the robot and human options. . . . . . . . 11
2.2 The binary question used in the human subject tests to elicit $W(\cdot)$ iteratively. 16
2.3 The binary choice question used in human subject tests to elicit the function $Q(\cdot)$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
2.4 Parameters with chosen values to form Bayes risk curves . . . . . . . . . . 36

3.1 Initial parameters . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 76
3.2 Optimized parameters . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 77
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>An illustration of the task scenario.</td>
<td>19</td>
</tr>
<tr>
<td>2.2</td>
<td>A sample question.</td>
<td>20</td>
</tr>
<tr>
<td>2.3</td>
<td>Results window of the current choice.</td>
<td>22</td>
</tr>
<tr>
<td>2.4</td>
<td>The elicited utility function $W(\cdot)$ where $\zeta_0 = -100$: (a) Participant 1, $a_1 = 0.8450$, $R^2 = 0.9998$, (b) Participant 2, $a_1 = 0.9734$, $R^2 = 0.9973$, (c) Participant 3, $a_1 = 0.7769$, $R^2 = 0.9956$, (d) Participant 4, $a_1 = 1.015$, $R^2 = 0.9993$, (e) Participant 5, $a_1 = 1.152$, $R^2 = 0.9999$.</td>
<td>24</td>
</tr>
<tr>
<td>2.5</td>
<td>The elicited $Q(\cdot)$ function: (a) Participant 1, $b_1 = 0.09706$, $b_2 = 5.184 \times 10^{-3}$, $b_3 = 0.7099$, $R^2 = 0.9501$, (b) Participant 2, $b_1 = 0.07407$, $b_2 = 4.316 \times 10^{-3}$, $b_3 = 0.6166$, $R^2 = 0.9221$, (c) Participant 3, $b_1 = 6.669$, $b_2 = 4.018 \times 10^{-3}$, $b_3 = 0.9064$, $R^2 = 0.9993$, (d) Participant 4, $b_1 = 116.7$, $b_2 = 3.65 \times 10^{-3}$, $b_3 = -0.296$, $R^2 = 0.9623$, (e) Participant 5, $b_1 = 2.927 \times 10^{-11}$, $b_2 = 0.02381$, $b_3 = 1.146$, $R^2 = 0.9846$.</td>
<td>25</td>
</tr>
<tr>
<td>2.6</td>
<td>The hybrid cell for collaborative assembly.</td>
<td>28</td>
</tr>
<tr>
<td>2.7</td>
<td>The proposed error function $\Psi(\zeta^M_A)$ for different values of parameters in the function.</td>
<td>34</td>
</tr>
<tr>
<td>2.8</td>
<td>(a) manual mode; (b) autonomous mode; (c) mixed mode.</td>
<td>36</td>
</tr>
<tr>
<td>2.9</td>
<td>The overall implementation scheme in a flowchart.</td>
<td>46</td>
</tr>
<tr>
<td>2.10</td>
<td>The human chooses whether he/she prefers the human (himself/herself) or the robot to take additional observations during the detection.</td>
<td>47</td>
</tr>
<tr>
<td>2.11</td>
<td>Questionnaire for assessing human’s regret for a case when the robot decides to ask the human to make observations, and the human made a wrong observation.</td>
<td>47</td>
</tr>
<tr>
<td>2.12</td>
<td>Regret-based modified mixed mode.</td>
<td>48</td>
</tr>
<tr>
<td>2.13</td>
<td>Objective measure of human-robot team fluency between regret-based and Bayesian approaches. Less idle time, less functional delay time and less non-concurrent activity time indicate more team fluency.</td>
<td>49</td>
</tr>
<tr>
<td>2.14</td>
<td>Subjective measures of team fluency and human’s trust in the robot between regret-based and Bayesian approaches.</td>
<td>49</td>
</tr>
<tr>
<td>2.15</td>
<td>Cognitive workload and assembly performance (efficiency, quality) between regret-based and Bayesian approaches.</td>
<td>50</td>
</tr>
</tbody>
</table>
3.1 Multiple robots must reach a set of destinations while avoiding obstacles and collisions with other robots, taking shorter but riskier paths between obstacles with human oversight when trusted to do so. That is, when the human operator trusts a robot, he/she plans for robot motion through visual feedback from a robot’s onboard camera using HMI such as keyboard, mouse, or joystick.

3.2 A Dynamic Bayesian Network (DBN) based model for dynamic, quantitative, and probabilistic trust estimates.

3.3 Plot of human performance when collaborating with a robot i.

3.4 Example simulation environment that consists of 3 robots with marked index numbers, 6 goals marked by diamonds, and 17 obstacles marked by crosses. The robot paths are demonstrated using black bold line segments. The robot start positions are marked by rectangles and the robot current positions are marked by circles. The robot sensing and communication ranges are marked by dashed circles.

3.5 Human input devices: gamepad and mouse. The gamepad buttons are used to select and confirm a robot to collaborate with, e.g., “blue” button for Robot 1, “green” button for Robot 2, “red” button for Robot 3, and “yellow” button for releasing the human. The mouse is used to give waypoints to a robot manually.

3.6 Illustration of manual motion planning using a mouse. The human assigns waypoints starting from a, passing between two obstacles to point b, and then point c, d, and eventually reaching the goal e.

3.7 GUI designs: (a) Comparison of robots’ maximum likelihood trust and request for manual motion planning, (b) Measure of trust change $c_i$, and (c) Measure of trust feedback $f_i$.

3.8 Final paths for 3-robot system switching between manual and autonomous planning mode.

3.9 Robot 1: (a) Evolution of human performance $P_H$, robot performance $P_{R1}$, and fault $F_1$, (b) trust belief $\text{bel}(T_1(k))$, (c) human intervention $m_1$, (d) trust change $c_1$, and trust feedback $f_1$.

3.10 Robot 2: (a) Evolution of human performance $P_H$, robot performance $P_{R2}$, and fault $F_2$, (b) trust belief $\text{bel}(T_2(k))$, (c) human intervention $m_2$, (d) trust change $c_2$, and trust feedback $f_2$.

3.11 Robot 3: (a) Evolution of human performance $P_H$, robot performance $P_{R3}$, and fault $F_3$, (b) trust belief $\text{bel}(T_3(k))$, (c) human intervention $m_3$, (d) trust change $c_3$, and trust feedback $f_3$.

3.12 Mean trust evolution of all 3 robots.

3.13 Interdependent Multi-Robot (2 robots) Dynamic Bayesian Networks (DBNs) based model for dynamic, quantitative, and probabilistic trust estimates.

3.14 Robot 1: Evolution of $P_H$, $P_{R1}$, $F_1$, $\text{filter}_1(T_1(k))$, $\text{smooth}_1(T_1(k))$, $m_1$, $c_1$, and $f_1$. 

viii
3.15 Robot 2: Evolution of $P_H$, $P_{R_2}$, $F_2$, $\text{filter}_2(T_2(k))$, $\text{smooth}_2(T_2(k))$, $m_2$, $c_2$, and $f_2$. ................................................................. 80

3.16 Interdependent Multi-Robot ($n$ robots) Dynamic Bayesian Networks (DBNs) based model for dynamic, quantitative, and probabilistic trust estimates. . . 81
Chapter 1

Introduction

1.1 Introduction of HRI and HRC Systems

In recent years, although autonomy has been well developed, human intervention is still required in many complex, dynamic and uncertain autonomous systems. Such requirement calls the need of human-robot collaboration (HRC). In HRC, humans and robots work together and complement each other optimally while bypassing their disadvantages [54]. For example, in HRC domain search tasks, a robot searches for objects within the task domain autonomously by using its sensing camera. The robot asks the human collaborator for help when it is hard for the robot to determine an object, as visual observation for such tasks is highly challenging for robots [14]. An experiment carried out in 2014 reports that the computer vision algorithms have at most 33% accuracy [5]. On the other hand, humans can achieve 100% accuracy in visual observation, if given enough time (more than 2.3s, to be exact) [47]. Therefore, collaborating with a human is important as missing objects would lead to low overall task performance. However, if the human is requested too often, fatigue builds up as the workload increases, and the human will need more time for correct observations. Consequently, the overall performance degrades. Hence, there exist
trade-offs between accuracy and work cost. Such situation, if balanced well, increases the overall observation accuracy.

Another example is manufacturing assembly, which is the key contributor to high overall manufacturing performance. Humans and robots are used in assembly widely and collaboratively. However, for a long time, fully automated assembly was efficient in simple and fixed tasks while it is inflexible and too expensive to adapt to the dynamic needs in assembly. On the other hand, fully manual assembly is inefficient as humans easily became bored and fatigued. Also, maloperation of assembly machines is harmful to human’s musculoskeletal systems. Therefore, HRC in manufacturing is necessary and can make the best use of advantages of both humans and robots while avoiding their limitations [61, 32]. In human-robot collaborative manufacturing assembly, the detection of correct assembly parts in the correct order guarantees high-quality assemblies [50]. The detection of obstacles is another key component to ensure safety and efficiency [51]. Although both detection tasks can be manually performed by human workers, as the workload increases, fatigue builds up resulting in reduced efficiency. One alternative to solve this problem it to use robot sensing. But as mentioned before, visual accuracy of robots is relatively low in the manufacturing environment as well due to disturbances, such as shadows of humans, and perturbations, such as occurrences of wrong parts. Additional, robot sensing capability may be limited by sensing technologies and calibration. Such uncertainties and limitations affect the assembly process and lead to inferior assembly performance. Therefore, human-robot collaborative detection is necessary, but allocation of detection between humans and robots must be balanced optimally.

Moreover, social cognition—how humans interact with others and social situations—has been studied in many research areas. Particularly, social cognition in human-robot interaction (HRI) has been delved at length for applications in robotics. It was not until recent years that the importance of social and cognitive skills in areas where interactions between
robots and humans are necessary, has been realized. Studies on HRI show that social and cognitive behaviors have huge impact in HRI [19, 15]. Among various social cognitions, we are particularly interested in regret and trust in HRI.

1.2 Regret

1.2.1 Background

For a human-robot team to perform the same task, if all members have a shared mental model, the joint team performance improves and becomes predictable [30, 38, 41]. In both domain search and manufacturing assembly cases, humans and robots perform the same tasks together. Hence, it is important that humans and robots share the same decision-making logic. More specifically, the robot can be programmed to imitate the way that a human makes decisions.

The Bayesian sequential decision-making (BSD) approach [12, 59] is often used in fully automated decision-making under uncertainty. It uses the expected value criterion to evaluate the potential costs of alternatives and chooses the one with lowest costs [31]. The BSD has been be used in domain search tasks and in manufacturing assembly to seek optimal solution under uncertainty. However, the BSD approach fails to explain human decision-making behaviors [31].

Regret, on the other hand, has been shown to play a crucial role in human decision-making behaviors. In decision-making under uncertainty, regret is experienced when humans realize that they could get a better result if an alternative option is adopted [39, 2, 10]. Even during decision-making, humans always predict potential regret, and hence they prefer decisions that will help them avoid it. Avoiding regret creates bias in individuals in making decision among options. This bias is found in large number of social and economical
decision-making experiments as well as in daily life. Thus it is considered rational. Regret theory, originally developed by [2, 39], successfully models human’s rational decision-making behaviors under uncertainty and opens up the possibility to program the influence of regret in robots’ decision-making algorithms. In other words, regret-based decision-making fit the human decision-making logic, which may further improve the overall team performance. Hence, we seek to modify BSD by integrating regret such that the robot imitates human logic.

1.2.2 Quantitative Modeling of Regret

To integrate regret theory in decision-making analysis, we need a model to measure regret quantitatively. In addition, different people sometimes are highly likely to have different decisions even to the same situation. This difference can be well captured in a quantitative regret model. If the quantitative model is customized to match the decision-making of an individual in a specific working scenario, then robots can adopted the same decision-making pattern of the individual by embedding this model. A quantitative model also enables the real-time human-like decision-making in robots. To our best knowledge, there exists no such regret model which is dedicated to the decision-making in HRC search tasks. Therefore, we propose a holistic regret model based on a general quantitative measurement of regret theory [4]. Furthermore, we designed a human-in-the-loop experiment to measure regret in HRC search tasks quantitatively. The collected data points further elicit requisite functions of regret theory. Our preliminary results match all the properties of regret theory. Our proposed parametric model shows a good fit to the experimental data.
1.3 Trust

1.3.1 Background

Trust, one’s assured reliance on others, has been studied for a long time and in several areas, like sociology, psychology, philosophy, economics, and systems. Trust typically describes a situation where one human is willing to rely on the action of another. Unfortunately, a human is uncertain about the outcome of the other’s actions. Such uncertainty quantifies the risk of failure. In robotic areas, human-robot trust is an important determinant in all HRI as the increase and decrease of trust affects the overall results of the interaction. Trust in HRI has been studied regarding to robot performance [46, 8, 36]. Trust between humans and robots enables the whole team to exchange information effectively and guarantees the completion of the collaborative tasks [25, 20]. Therefore, it is critical to build trust models to measure trust between a human and a robot. There exist a number of trust models including the qualitative trust model [44, 29], the argument-based probabilistic trust model [13, 44], the time-series trust model [35, 45, 36, 21], the neural net model [18], the computational trust model[26, 42], the regression model [16], and the most recent Bayesian Dynamic Network based Online Probabilistic Trust Inference Model (OPTIMo) [63]. These trust models characterize trust from different aspects.

1.3.2 Quantitative Modeling of Trust

Of particular interest, the time-series model proposed by [35] describes the real-time correlation between human-robot trust and several factors based on robot and human performance, trust state from the past, and fault. The time-series model can be used for real-time trust analysis and prediction for allocation tasks based on its capability of modeling the real-time changes of human-robot trust. Beside that, the OPTIMo trust model proposed
in [63], represented as a Dynamic Bayesian Network (DBN), formulates the Bayesian trust beliefs over a sequential of human’s latent trust states or the degree of human’s trust in almost real-time accurately and quantitatively. It is based on the past observation, robot performance, and the human’s trust inputs through computer interfaces. However, this trust model is robot performance-centric and does not consider human-robot joint performance and it only models trust for a specific robot. In this thesis, we integrate the time-series trust model into OPTIMo and formulate a new mathematical trust model for characterizing human’s dynamic trust in a robot. It takes into account of the human-robot joint performance. We test our new quantitative model in a multi-robot system with a human-in-the-loop.

Furthermore, we seek to find correlations among the measured trust models of different robots, as the trust for each robot in the new quantitative trust model is estimated independently, which fails to model the correlative trust in multi robot systems. Since the new proposed quantitative trust model is represented by a single stream DBN, we extend it to multi-stream DBNs to describe the correlations among trust in robots. We formulate the new trust beliefs based on the multi-stream DBNs for the 2-robot case and the \(n\)-robot general case. We then test the 2-robot case in the multi-robot motion planning tasks. The new computational trust model is able to measure human’s trust in multi-robot system. Future work will focus on the optimization of the new trust model as well as simplification of multi-robot correlation networks.

### 1.4 Structure

The organization of this thesis is as follows: In Chapter 2, we first proposed a quantitative measure of regret in HRC systems [37] and then include regret analysis into a probabilistic Bayesian sequential decision-making optimal strategy for manufacturing assembly [48, 49]. In Chapter 3, we integrate time-series trust model with OPTIMo trust
model [60] and extend the new quantitative trust model into multi-stream DBNs. We test the new interdependent multi-robot trust model in a motion planning simulation with a human-in-the-loop. In Chapter 4, we provide conclusion and future works.
Chapter 2

Quantitative Regret Measurements

2.1 A Quantitative Measure of Regret in Decision-Making for Human-Robot Collaborative Search Tasks

2.1.1 Problem Formulation

Consider an HRC system, which comprises of a human collaborator and a robot, searching a domain $\mathcal{D} \subset \mathbb{R}^2$ to find $m$ objects with positions unknown beforehand. The task domain $\mathcal{D}$ is divided into $n_{\text{tot}}$ small enough cells, where $n_{\text{tot}} \geq m$, by the occupancy grid mapping method [56] such that each cell contains at most one object. At an arbitrary cell $i$, where $i = 1, 2, \ldots, n_{\text{tot}}$, there are only two possibilities: either object presence or object absence. Denote the state of object existence at cell $i$ as the random variable $X(i)$. The actual realization of $X(i)$ can be denoted by $x(i) \in \{0, 1\}$ with 0 and 1 indicates object absence and presence, respectively. Before the HRC system observes the cell $i$, the true existence of an object is unknown. However, a prior guess of the existence may be obtained with possible information from the history. We denote the prior probability as $P(X(i) = x(i))$. The robot navigates into cell $i$ based on some motion control laws [59] and decides to observe the cell..
by either using its own on-board sensing system, *i.e.* the robot option “R”, or requesting the human collaborator, *i.e.* the human option “H”. Let a random variable $Y(i)$ represents the observation result in cell $i$. The observation capability of each option is quantified by the conditional probability of making a wrong observation $P_{\Upsilon}(Y(i) \neq x(i)|X(i) = x(i))$, where the subscript $\Upsilon = R$ for the option “R” and $\Upsilon = H$ for the option “H”. Note that the probability information of robot sensing capability is often assumed known and constant.

When observing cells, both the option “R” and the option “H” have operational costs, which are expressed in terms of amounts of money [7]. Denote the operational costs of the option “R” and “H” as $c_R < 0$ and $c_H < 0$. $c_R$ is assumed to be constant. When an observation result turns out to be wrong, we further penalize it by adding the artificial cost $a_R < 0$ to $c_R$. The value of $a_R$ is related to the importance of the objects to be detected. Combining $c_R$ and $a_R$, we define the wrong observation cost made by the robot as $o_R = a_R + c_R < 0$. Note that the negative sign of costs emphasizes their nature of loss. In contrast, the human observation accuracy is assumed to be 100% and there is no need to define the human wrong-observation cost.

### 2.1.2 Integration of Regret Theory in Bayesian Sequential Decision-Making

The BSD approach can be used to find an optimal solution to the HRC decision-making problems under uncertainty [12, 59]. In Sec. 2.1.2.1, we show that BSD actually uses the expected value criterion to evaluate the potential cost and make decisions. But human decision-making behaviors violate the expected value criterion most the time [1, 2, 3, 31, 39]. Regret theory, as an alternative, better matches the evidence in human subject tests [1, 39]. Therefore, in Sec. 2.1.2.2, we modify the BSD with regret theory and introduce its properties. Since we only focus on the decision-making within one cell, the
2.1.2.1 Bayesian Sequential Decision-Making

In the BSD, which seeks to find the option with the lowest potential cost, positive expected Bayes risks are defined for each option to measure how “risky” the option is [59]. Lower “risk” corresponds to lower potential cost. However, in this paper, we consider negative costs: \( c_R < 0 \), \( c_H < 0 \) and \( o_R < 0 \), and hence we need to define the negative expected Bayes risks. The more negative the expected Bayes risk is, the higher the potential cost will be.

Same as the positive expected Bayes risk in the framework of BSD, the negative expected Bayes risk of the option “\( R \)” is,

\[
E[K_R] = k_{R0}P(X = 0) + k_{R1}P(X = 1),
\]

where the random variable \( K_R \) is the conditional Bayes risk with \( P(K_R = k_{R0}) = P(X = 0) \) and \( P(K_R = k_{R1}) = P(X = 1) \). The values \( k_{R0} \) and \( k_{R1} \) are realizations of the conditional Bayes risk and they are defined as,

\[
k_{R0} = a_RP_R(Y = 1|X = 0) + c_R, \tag{2.2}
\]

\[
k_{R1} = a_RP_R(Y = 0|X = 1) + c_R. \tag{2.3}
\]

We then define a probability \( p_R \) with

\[
p_R = 1 - P_R(Y = 1|X = 0)P(X = 0) - P_R(Y = 0|X = 1)P(X = 1). \tag{2.4}
\]

Now substitute Eqn. (2.2) and (2.3) into Eqn. (2.1), and based on Eqn. (2.4) and the
relations $P(X = 0) + P(X = 1) = 1$ and $o_R = a_R + c_R$, we have

$$E[K_R] = p_R c_R + (1 - p_R) o_R. \quad (2.5)$$

It is clear that $E[K_R]$ is actually the expected value of all possible costs of the option “R”.

To simplify the definition of $p_R$, Eqn. (2.4) reduces to,

$$p_R = P_R(Y = 1, X = 1) + P_R(Y = 0, X = 0). \quad (2.6)$$

which indicates that the probability $p_R$ is the sum of the joint probabilities of making a correct observation.

The conditional probability of human making a wrong observation $P_H(Y(i) \neq x(i)|X(i) = x(i)) = 0$. Hence, similar as Eqn. (2.4), the total probability $p_H$ for the human to make a correct observation is 1. The expected Bayes risk of the option “H” is therefore $E[K_H] = c_H$, where the random variable $K_H$ is the conditional Bayes risk of the option “H”.

The HRC decision-making problem can be formulated into a binary choice question, as shown in Table 2.1. Each option in Table 2.1 contains two results with the corresponding costs and probabilities. The costs are associated with the specific observation results. The results are mutually exclusive in the same option and independent from different options.

<table>
<thead>
<tr>
<th>Result</th>
<th>Option R</th>
<th>Option H</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cost</td>
<td>Probability</td>
</tr>
<tr>
<td>Correct</td>
<td>$c_R$</td>
<td>$p_R$</td>
</tr>
<tr>
<td>Wrong</td>
<td>$o_R$</td>
<td>$1 - p_R$</td>
</tr>
</tbody>
</table>

Table 2.1: The binary choice question with the robot and human options.

The comparison of the two options will generate three possibilities: preferring the robot option, denoted as $R \succ H$; preferring the human option, denoted as $R \prec H$; being
indifferent between the two options, denoted as $R \sim H$. The BSD compares the expected value of costs in each option in Table 2.1 to decide which one to choose:

$$E[K_R] \preceq E[K_H] \Leftrightarrow R \sim H.$$  

(2.7)

Hence, we show that BSD actually makes decisions based on the expected value criterion, which fails to capture human decision-making behaviors.

### 2.1.2.2 Integration of Regret Theory

In this section, we focus on using regret theory to model human decision-making behaviors under uncertainty and modifying the BSD with regret. Similar as the expected value criterion, regret theory models the preference over individual options with costs and probabilities. However, it defines utilities for individual results within an option. The expected utility of each option is calculated similarly as the expected cost in Eqn. (2.5) and compared to make a decision. Without loss of generality, we focus on the option “R”. The explanation on the option “H” follows the same manner.

Regret theory hypothesizes that the utility of a result in the option “R” depends not only on itself but on the result in the alternative option “H” [39]. It uses modified utility function $M(c_R, c_H)$ to model the utility of $c_R$. The cost $c_H$ in $M(c_R, c_H)$ is the one with which $c_R$ compares. The modified utility $M(c_R, c_H)$ is defined as,

$$M(c_R, c_H) = W(c_R) + T(W(c_R) - W(c_H)), \quad (2.8)$$

where $W(\cdot)$ is the choiceless utility function and $T(\cdot)$ is the regret function. $W : \mathbb{R} \mapsto \mathbb{R}$ is injective and quantifies human’s subjective estimate on the loss of a cost, given no other alternatives. The regret function $T : \mathbb{R} \mapsto \mathbb{R}$ is injective and describes the regret/rejoice a human feels by comparing the two costs. Similarly, the modified utility for $o_R$ is $M(o_R, c_H)$. 
Hence, the expected modified utility of option “R” is,

\[ E[M_R] = p_{M1}M(c_R, c_H) + p_{M2}M(o_R, c_H). \] (2.9)

where \( p_{M1} \) and \( p_{M2} \) are the probabilities of the modified utilities, which need to be defined to compute \( E[M_R] \).

The function \( M(c_R, c_H) \) contains two costs from two different options respectively and the options are independent. The estimation of \( M(c_R, c_H) \) happens if the two options both happen. Therefore, \( p_{M1} \) is the joint probability of \( c_R \) and \( c_H \), i.e. \( p_{M1} = p_R \cdot 1 \). Similarly, \( p_{M2} = (1 - p_R) \cdot 1 \). We have,

\[ E[M_R] = p_RM(c_R, c_H) + (1 - p_R)M(o_R, c_H). \] (2.10)

Similarly, the expected modified utility of the option “H” is,

\[ E[M_H] = p_RM(c_H, c_R) + (1 - p_R)M(c_H, o_R). \] (2.11)

Then the function \( Q(\cdot) \) is defined based on regret theory[39]:

\[ Q(W(o_R) - W(c_H)) = M(o_R, c_H) - M(c_H, o_R) \]
\[ = W(o_R) + T(W(o_R) - W(c_H)) - W(c_H) - T(-(W(o_R) - W(c_H))), \] (2.12)

where the variable \( \chi \) can be \( c_R \) or \( o_R \). It follows that the function \( Q(\cdot) \) is \( \mathbb{R} \mapsto \mathbb{R} \) and injective.

Regret theory then compares \( E[M_R] \) and \( E[M_H] \) to make decisions. Using the definition of \( Q(\cdot) \), we have,

\[ E[M_R] - E[M_H] = p_RQ(W(c_R) - W(c_H)) + (1 - p_R)Q(W(o_R) - W(c_H)). \] (2.13)
The decision is then made with,

$$E[M_R] - E[M_H] \geq 0 \Leftrightarrow R \succsim H.$$  \hspace{1cm} (2.14)

All the arguments and parameters in Eqn. (2.14) are known and readily listed in Table 2.1. Since the right-hand-side of Eqn. (2.13) only relies on the functions $Q(\cdot)$ and $W(\cdot)$, regret theory creates the properties for them [39]. We summarize the properties of the function $W(\cdot)$ and $Q(\cdot)$:

1. $W(\cdot)$ is continuous, linear and increasing. For any result with cost $c_u$, the choiceless utility takes the form of $W(c_u) = \alpha c_u + \beta$, where $\alpha > 0$ and $\beta$ are real constants.

2. The reference cost $c_0$, which makes $W(c_0) = 0$, can be transferred to any desired value.

3. $Q(\cdot)$ is continuous and skew symmetric. For any two results with costs $c_u$, $c_v$, the value $Q(W(c_u) - W(c_v)) = -Q(W(c_v) - W(c_u))$.

4. $Q(\cdot)$ is non-decreasing. For any four results with costs $c_u$, $c_v$, $c_s$, and $c_t$ such that $W(c_u) - W(c_v) < W(c_s) - W(c_t)$, the value $Q(W(c_u) - W(c_v)) \leq Q(W(c_s) - W(c_t))$.

5. $Q(\cdot)$ is concave with negative independent variables. For any three results with costs $c_u$, $c_v$, $c_s$ such that $W(c_u) < W(c_v) < W(c_s)$, the value $Q(W(c_u) - W(c_s)) < Q(W(c_u) - W(c_v)) + Q(W(c_v) - W(c_s))$.

Regret theory shows a good match with data collected from human decision-making tests [39]. Therefore, we can modify the decision-making strategy in BSD (Eqn. (2.7)) with the decision-making strategy from regret theory (Eqn. (2.14)), and the resulting decisions will be more like from a human.
2.1.3 Quantitative Measurement of Regret

In this section, we develop a holistic regret model for HRC search tasks. We measure regret theory quantitatively by eliciting \( W(\cdot) \) and \( Q(\cdot) \), based on their properties, instead of the modified utility function \( M(\cdot, \cdot) \). Since \( W(\cdot) \) is the argument of \( Q(\cdot) \), in Sec. 2.1.3.1, we firstly present the elicitation of \( W(\cdot) \), followed by the elicitation of \( Q(\cdot) \) in Sec. 2.1.3.2.

2.1.3.1 Elicitation of the Function \( W(\cdot) \)

Since \( W(\cdot) \) is \( \mathbb{R} \mapsto \mathbb{R} \) and injective, the shape of \( W(\cdot) \) can be sketched from the data points collected from participants. Let \( (\zeta_j, W(\zeta_j)), j = 0, 1, 2, \ldots, J \) represent the data points on the graph of \( W(\cdot) \), where \( j \) and \( J \) are the index and the total number, respectively. Then, we convert Table 2.1 to Table 2.2, by first using the property (2) to set the constant \( c_R \) as the reference cost \( \zeta_0 \) such that \( \zeta_0 = c_R \) and \( W(\zeta_0) = 0 \), as well as replacing \( c_H \) with \( \zeta_j \) and \( o_R \) with \( \zeta_j + 1 \). Next, according to Eqn. (2.13) and (2.14), we consider the indifference case \( R \sim H \) to elicit \( W(\cdot) \),

\[
p_R Q(W(\zeta_0) - W(\zeta_j)) + (1 - p_R) Q(W(\zeta_j) - W(\zeta_j + 1)) = 0. \tag{2.15}
\]

Using the property (3) and \( W(\zeta_0) = 0 \), we have

\[
p_R Q(W(\zeta_j)) = (1 - p_R) Q(W(\zeta_j + 1)) - W(\zeta_j)). \tag{2.16}
\]

Because \( Q(\cdot) \) is \( \mathbb{R} \mapsto \mathbb{R} \) and injective, if the values of \( Q(\cdot) \) are equal, their arguments also equal to each other. Therefore, to equalize the factors of \( Q(\cdot) \) on both sides of Eqn. (2.16), we choose \( p_R = 0.5 \) and hence

\[
Q(W(\zeta_j)) = Q(W(\zeta_j + 1)) - W(\zeta_j)). \tag{2.17}
\]
This results in,

\[ W(\zeta_{j+1}) = 2W(\zeta_j), \]  
(2.18)

where \( j = 1, 2, 3, \ldots, J - 1 \). Suppose now the point \((\zeta_j, W(\zeta_j))\) is known, the indifference condition \( R \sim H \) can be found by trying different values for \( \zeta_{j+1} \) in Table 2.2 until participants report indifferent preference. The final elicited value for \( \zeta_{j+1} \) is the one that \( R \sim H \) holds and the value \( W(\zeta_{j+1}) \) can be determined by (2.18). The value of \( \zeta_j \) in the current binary choice question is updated with the previous elicited value of \( \zeta_{j+1} \) iteratively to continue the elicitation.

<table>
<thead>
<tr>
<th>Which option do you prefer?</th>
<th>Option R</th>
<th>Option H</th>
</tr>
</thead>
<tbody>
<tr>
<td>Result</td>
<td>Cost</td>
<td>Probability</td>
</tr>
<tr>
<td>Correct</td>
<td>( \zeta_0 )</td>
<td>( p_R )</td>
</tr>
<tr>
<td>Wrong</td>
<td>( \zeta_{j+1} )</td>
<td>( 1 - p_R )</td>
</tr>
</tbody>
</table>

Table 2.2: The binary question used in the human subject tests to elicit \( W(\cdot) \) iteratively.

Although the point \((\zeta_0, W(\zeta_0))\) is known, we cannot use the above procedure to find \((\zeta_1, W(\zeta_1))\). In fact, the point \((\zeta_1, W(\zeta_1))\) is regarded as the starting point for the elicitation. Since \( \zeta_1 \) is the next value of \( \zeta_0 \), the reasonable choice for the value \( \zeta_1 \) is the minimum human operational cost \( c_H \). The scaling factor of \( W(\cdot) \), \( \alpha \) in property(1), is determined by the value of \( W(\zeta_1) \) and is left to be designed. In this work, we choose \( W(\zeta_1) = \tilde{w} \), where \( \tilde{w} \) is a constant. The data points \((\zeta_j, W(\zeta_j)), j = 2, 3, \ldots, J \) can be determined iteratively given that \((\zeta_1, W(\zeta_1))\) is known. Algorithm 1 summarizes the elicitation procedure for function \( W(\cdot) \).

2.1.3.2 Elicitation of the Function \( Q(\cdot) \)

Similarly, the shape of \( Q(\cdot) \) can also be sketched from the data points collected from participants to reflect the properties iteratively. Let \((\eta_k, Q(\eta_k)), k = 1, 2, \ldots, K \) represent
Algorithm 1 Elicitation of the Function $W(\cdot)$

**Input:** $\zeta_0$, $c_H$, $\tilde{w}$, $J$

**Output:** Data points $\{(\zeta_j, W(\zeta_j))\}, j = 0, 1, 2, \ldots, J$

1: $W(\zeta_0) \leftarrow 0$
2: $\zeta_1 \leftarrow c_H, W(\zeta_1) \leftarrow \tilde{w}$
3: **while** $1 \leq j < J$ **do**
4:  **while** True **do**
5:  Define a new $\zeta_{j+1}$
6:  Collect the preference of a human participant
7:  **if** $R \sim H$ **then**
8:  break
9:  **end if**
10: **end while**
11: $W(\zeta_{j+1}) \leftarrow 2W(\zeta_j)$
12: $j \leftarrow j + 1$
13: **end while**
14: Record $\zeta_j$ and $W(\zeta_j), j = 0, 1, 2, \ldots, J$

the data points on the graph of $Q(\cdot)$, where $k$ and $K$ are the index and the total number, respectively. We now convert Table 2.1 to Table 2.3, by replacing $p_R$ with a series of different constant probabilities $p_k, k \in \{1, 2, \ldots, K\}$, as well as replacing $c_H$ with a constant $c_H, c_R$ with the reference cost $\zeta_0$ such that $W(\zeta_0) = 0$ and $o_R$ with an undetermined series of costs $\{\zeta_k\}, k = 1, 2, \ldots, K$. Again, we focus on the indifference case $R \sim H$. By modifying Eqn. (2.16), we have,

$$Q(W(\zeta_k) - W(c_H)) = \frac{p_k}{1-p_k} Q(W(c_H)). \quad (2.19)$$

Then, substitute $c_H$ for $c_H$ in Eqn. (2.19) and define $\eta_k = W(\zeta_k) - W(c_H)$,

$$Q(\eta_k) = \frac{p_k}{1-p_k} Q(W(c_H)). \quad (2.20)$$

Suppose the value of $Q(W(c_H))$ is known, the condition $R \sim H$ for $p_k$ can be found by trying different values for $\zeta_k$ in Table 2.3 until participants report indifferent preference. The final elicited value of $\zeta_k$ is the one when Eqn. (2.20) holds. The value $W(c_H) = \tilde{w}$. 17
is constant. Then, the value of $Q(\eta_k)$ can be determined by Eqn. (2.20) for different $p_k$. Scaling $Q(\cdot)$ does not influence the decision-making according to Eqn. (2.14). Then, we are allowed to arbitrarily choose a scale for $Q(\cdot)$ by designing $Q(W(\zeta_H)) = \tilde{q}$.

| Which option do you prefer? | Option R | | | | Option H | | | |
|---|---|---|---|---|---|---|---|
| Result | Cost | Probability | Cost | Probability | | | |
| Correct | $\zeta_0$ | $p_k$ | $\zeta_H$ | 1 | | | |
| Wrong | $\zeta_k$ | $1 - p_k$ | | | | | |

Table 2.3: The binary choice question used in human subject tests to elicit the function $Q(\cdot)$.

Algorithm 2 summarizes the procedure of eliciting $Q(\cdot)$. Note that only property (2) and (3) are used in designing the quantitative measurement framework. Properties (4) and (5) are not trivial and can be used to bilaterally validate the measurement and regret theory.

**Algorithm 2** Elicitation of the Function $Q(\cdot)$

**Input:** $\zeta_0$, $\zeta_H$, $\{p_k\}$, $\tilde{q}$, $W(\cdot)$  
**Output:** Point coordinates $\{(\eta_k, Q(\eta_k))\}, k = 1, 2, \ldots, K$

1: while $1 \leq k \leq K$ do  
2: while True do  
3: Define a new $\zeta_k$  
4: Collect the preference of a human participant  
5: if $R \sim H$ then  
6: break  
7: end if  
8: end while  
9: $\eta_k \leftarrow W(\zeta_k) - W(\zeta_H)$  
10: Calculate $Q(\eta_k) \leftarrow \frac{p_k}{1-p_k}\tilde{q}$  
11: $k \leftarrow k + 1$  
12: end while  
13: Record $\eta_k$ and $Q(\eta_k), k = 1, 2, \ldots, K$
2.1.4 Experimental Design

2.1.4.1 Experimental Setup

In this section, we present our experiment design to measure human regret quantitatively during decision-making for an HRC collaborative search task using the methods developed in Sec. 2.1.3. The scenario we considered was a $10 \times 10$ domain containing ten objects with unknown positions. One robot searched the domain to find all objects and one human sat in front of the computer monitored the domain and robot as shown in Fig.2.1. The experiments were performed through a computer-generated interface designed by Matlab GUIDE (Graphical user interface design environment). The goal of the experiment was to collect enough data points to elicit the functions $W(\cdot)$ and $Q(\cdot)$ of each participant.

![Figure 2.1: An illustration of the task scenario.](image)

The experiment task for each participant is to allocate the observation tasks to himself/herself or the robot considering possible regret through a series of binary choice ques-
tions, like the one shown in Fig.2.2, so as to achieve the best result. There are two options, “(R)obot” or “(H)uman”, shown in Fig.2.2. The cost and probability of each possible observation result were provided as well.

![Image of a sample question](image)

Figure 2.2: A sample question.

2.1.4.2 Participants and Compensation

As a preliminary study, we recruited five participants (3 males, 2 females, average age was 24.4 years old). All of them are graduate students in engineering majors\(^1\) at Clemson University. The compensation for each participant in the experiment was $10.

2.1.4.3 Experimental Procedure

Each experiment started with a description and a sample question, followed by a training session. The experiment lasted 22 minutes on average. A participant needed to choose either the “(R)obot” or “(H)uman” option that he/she thought intuitively would result in minimum negative costs. Although we could ask the participants to directly tell what costs and probabilities make the option “R” and “H” indifferent, they usually cannot estimate the values accurately. Therefore, we elicited these values through a number of

\(^1\)Note that our goal is to measure regret quantitatively for each participant to be used in the BSD framework for the robot. Therefore, in this work, we are not interested in performing statistical analysis as most human factors and economics research does. The project IRB (IRB2013-289) was approved.
binary-choice questions like the one shown in Fig. 2.2 instead. The possible observation result for choosing “(R)obot” was either correct or wrong and the observation result for choosing “(H)uman” was correct. A confirmation question was required after each choice to avoid “mis-click” errors. If the participant confirmed his/her choice, a result window of the current chosen option was provided. A particular case was shown in Fig. 2.3, where the current choice was “(R)obot” and the robot observation was correct which was indicated by the “tick” sign (if wrong, then a “cross” sign). We can see that the current cost was “-100”. Moreover, from the bar chart in Fig. 2.3, the participant got to know both the current cost (dark blue bar with red edge) and the possible cost if chose differently (light blue bar). After the participant clicked “Continue”, the next question popped up, with the same format as shown in Fig. 2.2 but with different costs and/or probabilities. Therefore, the participant made choices only based on the costs and probabilities of the current question. Based on their answers, we collected data points to elicit the functions $W(\cdot)$ and $Q(\cdot)$. The elicitation results are analyzed below.

2.1.5 Results Analysis

2.1.5.1 Experimental Results

We selected 5 data points to elicit the $W(\cdot)$ function and 9 data points to elicit the $Q(\cdot)$ function according to Algorithm 1 and Algorithm 2, respectively. More data points are used for eliciting $Q(\cdot)$ because of its nonlinearity. To elicit $W(\cdot)$, the data points are first plotted as square markers in Fig. 2.4. The curves are then fitted using the Curve Fitting Toolbox in Matlab. We initially used a second order polynomial function to fit the data. Although the data points were well fitted, the coefficient of the second-order term was strikingly small, indicating that all the elicited choiceless utility functions were linear.
Therefore, we performed linear curve fitting using the following model,

\[ W(\zeta) = a_1(\zeta - \zeta_0), \tag{2.21} \]

where \( \zeta \) is the cost, \( \zeta_0 \) is the constant reference point in Table 2.2 and 2.3 and \( a_1 > 0 \) is a constant parameter. The elicited utility functions are represented by the black solid lines in Fig.2.4. Furthermore, when performing regression analysis, the coefficient of determination, denoted as \( R^2 \), is always considered as a statistical measure of how close the data is to the fitted regression line (\( 0 < R^2 < 1 \)). The higher \( R^2 \) is, the better the regression line approximates the corresponding data points. When fitting the curves, we noticed that \( R^2 \) of all the elicited utility functions \( W(\cdot) \)s were greater than 0.99. The elicited parameter \( a_1 \)
and corresponding $R^2$ for each participant is shown in the caption of Fig.2.4.

Similarly, to elicit the $Q(\cdot)$ function, we first plot the data points as square markers with blue edge in Fig.2.5. Next, inspired by the regret function model in [11], we propose a parameterized model as the curve fitting model for $Q(\cdot)$

$$Q(\eta) = b_1 \sinh(b_2 \eta) + b_3 \eta,$$

where $\eta$ is the difference of choiceless utility functions as shown in Eqn. (2.19) and $b_1, b_2, b_3 > 0$ are constant parameters. The fitted $Q(\cdot)$ functions were shown as the solid lines in Fig.2.5. When fitting the curves, $R^2$ of all the elicited $Q(\cdot)$ functions were greater than 0.92. The elicited parameters $(b_1, b_2, b_3)$ and corresponding $R^2$ for each participant are listed in the caption of Fig.2.5.

### 2.1.6 Validity Analysis

#### 2.1.6.1 Internal Validity Analysis

To verify that our experiments had high degrees of internal validity, we included some distracting questions, with the same format as the sample question in Fig.2.2 but with random costs. In the experiment, these distracting questions can prevent the participants from observing the trend of the change of cost of each question and answering predictively. Answers of distracting questions were excluded from data collection.

#### 2.1.6.2 External Validity Analysis

We disarmed threats to external validity for high generalizability. We set different situational specifics of our experiment. For example, participants at different ages were from various fields of engineering. Moreover, they performed the experiments at different times and locations.
Figure 2.4: The elicited utility function $W(\cdot)$ where ($\zeta_0 = -100$): (a) Participant 1, $a_1 = 0.8450$, $R^2 = 0.9998$, (b) Participant 2, $a_1 = 0.9734$, $R^2 = 0.9973$, (c) Participant 3, $a_1 = 0.7769$, $R^2 = 0.9956$, (d) Participant 4, $a_1 = 1.015$, $R^2 = 0.9993$, (e) Participant 5, $a_1 = 1.152$, $R^2 = 0.9999$. 
Figure 2.5: The elicited $Q(\cdot)$ function: (a) Participant 1, $b_1 = 0.09706$, $b_2 = 5.184 \times 10^{-3}$, $b_3 = 0.7099$, $R^2 = 0.9501$, (b) Participant 2, $b_1 = 0.07407$, $b_2 = 4.316 \times 10^{-3}$, $b_3 = 0.6166$, $R^2 = 0.9221$, (c) Participant 3, $b_1 = 6.669$, $b_2 = 4.018 \times 10^{-3}$, $b_3 = 0.9064$, $R^2 = 0.9993$, (d) Participant 4, $b_1 = 116.7$, $b_2 = 3.65 \times 10^{-3}$, $b_3 = -0.296$, $R^2 = 0.9623$, (e) Participant 5, $b_1 = 2.927 \times 10^{-11}$, $b_2 = 0.02381$, $b_3 = 1.146$, $R^2 = 0.9846$. 
We provide the detailed analysis of the elicited functions $W(\cdot)$ and $Q(\cdot)$ of participant 1 based on properties summarized in Sec. 2.1.2. Analyses of participant 2-5 were similar. The elicited function $W(\cdot)$ of participant 1 is $W(\zeta) = 0.845\zeta + 84.5$ with $\zeta_0 = -100$. The elicited utility functions satisfied all the properties that regret theory requested, as $W(\zeta)$ is linear, nondecreasing (Property (1)) and $W(\zeta_0) = 0$ (Property (2)). We then analyzed the $Q(\cdot)$ function of participant 1: $Q(\eta) = 0.09706\sinh(5.184 \times 10^{-3}\eta) + 0.7099\eta$, where $Q(\eta)$ is continuous and skew symmetric (Property (3)) and non-decreasing (Property (4)). From Fig. 2.5, we also know that the $Q(\eta)$ function is concave when $\eta < 0$ (Property (5)). Therefore, the elicited $Q(\cdot)$ functions also satisfied all the properties that regret theory claimed. However, not all human decision-making behaviors can be predicted by regret theory accurately. For instance, as shown in the caption of Fig. 2.5, the extremely small parameter $b_1$ of elicited $Q(\cdot)$ functions of participant 5 indicated that the elicited $Q(\cdot)$ function almost failed in following Property (5)). Despite this random unpredictability in human behaviors, our proposed framework can quantify the decision-making behaviors of most participants. Our future work includes human subject tests with a justified sample size to further validate the preliminary results and to test the accuracy of our regret model.

For each participant, the elicited $W(\cdot)$ and $Q(\cdot)$ functions modeled his/her decision-making behavior quantitatively. With the elicited functions, we finished building the regret model of the corresponding participant, which can predict human’s current decision with given costs and properties of the present choice problem. Such regret models, if embedded in HRC search tasks as in [59], enable the robot to make decision more like human. Moreover, the implementation of human-like decision-making behaviors would ensure that humans and robots share a same mental model. The uniformity of the decision-making logics of all HRC members may help the team to reach a relatively high level of effective-
ness and correctness in search tasks. Our future work includes the implementation of the proposed regret-based decision-making framework in robotic experiments and a systematic comparison with the original BSD approach with human-in-the-loop.

2.2 Regret-based Allocation of Autonomy in Shared Detection Phenomena for Human-Robot Collaborative Assembly in Manufacturing

2.2.1 Detection Problem Formulation with Focus on the Selected Assembly Task

As shown in Fig. 2.6, the robot needs to make decisions on various detection phenomena during the collaborative assembly. We think that there may have three modes in making observations regarding the shared detection phenomena: (i) manual mode—the robot seeks help from the human, if necessary. The human then makes observations on the target part, supplies the observation information to the robot and thus helps the robot make the decision about the correctness of the part, (ii) autonomous mode—the robot makes additional observations on the target part using its vision system and makes the decision on correctness of the assembly part, and (iii) mixed mode—the robot itself makes observations on the target part and also asks the human to make observations if necessary so that the robot can use the observation information to make the decision about the correctness of the part. The selection of the mode may be real-time which depends on various dynamic parameters such as (i) probability of correctness in the observation/sensing by the human and the robot, (ii) costs involved in making observations, etc. Note that the change in the modes does not occur randomly. Instead, the changes occur when it is deemed necessary
based on the status of one or more of the dynamic parameters as mentioned above.

Figure 2.6: The hybrid cell for collaborative assembly

Denote \( \tilde{\Theta} \) as the part (object) to be observed (the targeted part). Let the binary random variable \( X(\tilde{\Theta}) \) represents state of the part \( \tilde{\Theta} \),

\[
X(\tilde{\Theta}) = \begin{cases} 
1 & \text{if the target part is } \tilde{\Theta}, \\
0 & \text{if the target part is not } \tilde{\Theta}.
\end{cases}
\]  

(2.23)

We assume that the robot can sense only one part at a given time step or at each run/trial.
Now, we define a binary observation variable \( Y(\tilde{o}, \text{run}) \) for the robot, when \( Y(\tilde{o}, \text{run}) = 1 \) and \( Y(\tilde{o}, \text{run}) = 0 \) indicate that the \( \tilde{o} \) is the correct or wrong part respectively observed by the robot at the current run. Similarly, \( Y_H(\tilde{o}, \text{run}) = 1, Y_H(\tilde{o}, \text{run}) = 0 \) indicate that \( \tilde{o} \) is the right or the wrong part respectively observed by the human. Let \( \beta_r \) and \( \beta_H \) be the sensing capability of the robot and the human respectively.

### 2.2.2 Bayesian Optimal Decision-Making Algorithm for Autonomy Allocation

A probabilistic Bayesian sequential decision-making strategy is proposed in this section to obtain the optimal choice among the three available options regarding the allocation of autonomy. We denote the decision rule as \( \Delta \), the number of observations as \( L \), and the prior probability of part being right or wrong as \( P(X(\tilde{o}) = x(\tilde{o}); \text{run}), x(\tilde{o}) = 0, 1 \).

Then, the expected Bayes risk function may be defined as follows, where \( \Delta \) is a \( 2 \times (L + 1) \) matrix. In general, \( L \geq 1 \).

\[
\text{risk}(\Delta, L, P(X(\tilde{o}) = x(\tilde{o}); \text{run}), \text{run}) = \\
R(\Delta, L | X(\tilde{o}) = 0, \text{run})P(X(\tilde{o}) = 0; \text{run}) + R(\Delta, L | X(\tilde{o}) = 1, \text{run})P(X(\tilde{o}) = 1; \text{run}).
\]

(2.24)

\( R(\Delta, L | X(\tilde{o}) = x(\tilde{o}); \text{run}) \), the conditional Bayes risk function providing state \( X(\tilde{o}) = x(\tilde{o}) \), is as follows.

For the autonomous mode,

\[
R(\Delta, L | X(\tilde{o}) = x(\tilde{o}); \text{run}) \triangleq R^A(\Delta, L | X(\tilde{o}) = x(\tilde{o}); \text{run}) = \\
\text{Prob(decide } X(\tilde{o}) \neq x(\tilde{o}) | X(\tilde{o}) = x(\tilde{o})) + c_r(\tilde{o}, \text{run})E[N(\phi)|X(\tilde{o}) = x(\tilde{o})] = \\
COST \Delta B_r^{x(\tilde{o})} + c_r(\tilde{o}, \text{run})E[N(\phi)|X(\tilde{o}) = x(\tilde{o})].
\]

(2.25)
For the manual mode,

\[ R(\Delta, L|X(\hat{o}) = x(\hat{o}); \text{run}) \triangleq R^M(\Delta, L|X(\hat{o}) = x(\hat{o}); \text{run}) = \]

\[ \text{Prob}(\text{decide } X(\hat{o}) \neq x(\hat{o})|X(\hat{o}) = x(\hat{o})) + c_H(\hat{o}, \text{run})E[N(\phi)|X(\hat{o}) = x(\hat{o})] = \]

\[ COST\Delta B^x_H + c_H(\hat{o}, \text{run})E[N(\phi)|X(\hat{o}) = x(\hat{o})]. \]

(2.26)

In the conditional Bayes risk functions (3) and (4), \( c_r(\hat{o}, \text{run}), c_H(\hat{o}, \text{run}) \) are the costs induced by making more observations by robot and human, respectively. It is assumed that human observation cost is higher than robot observation cost, i.e., \( c_H(\hat{o}, \text{run}) > c_r(\hat{o}, \text{run}) \) because human observation deviates the human from the assembly task, and also imposes cognitive workload. \( B^x_r, B^x_H \) are the columns in the conditional probability matrix of the autonomous and the manual mode respectively (see [58, 59] for details). \( N(\phi) \) is the time of stopping for making observation and arriving at a decision regarding whether the part is right or wrong. Because the observation is random, \( N(\phi) \) is a random variable. \( COST \) is the penalty of making a decision on the part status/state given the actual state. To simplify the problem, currently a uniform cost assignment (UCA) is assumed such that,

\[ COST = \begin{cases} 
0 & \text{if decision is consistent with actual state.} \\
1 & \text{otherwise.} 
\end{cases} \]

The minimum expected Bayes risk \( \text{risk}^* \) at an arbitrary run is found by searching for a pair of decision rule and observation length \( (\Delta^*, L^*) \) as follows.

\[ \text{risk}^*(\Delta^*, L^*, P(X(\hat{o}) = x(\hat{o}); \text{run}), \text{run}) = \]

\[ R(\Delta^*, L^*|X(\hat{o}) = 0, \text{run})P(X(\hat{o}) = 0; \text{run}) + R(\Delta^*, L^*|X(\hat{o}) = 1, \text{run})P(X(\hat{o}) = 1; \text{run}). \]

(2.27)
2.2.3 Inclusion of Regret Analysis in Bayesian Decision-Making for Autonomy Allocation

The expected Bayesian risk function can be translated as expected utility in expected utility theory. In behavior economics literature [39, 31, 2], it has been suggested that expected utility theory is not obeyed by a human decision-maker. People react differently to different decision-making circumstances. For instance, when humans face a choice problem of two mutually alternative options, each has the same expected utility to the other, systematically, they change their decisions from one to another depending on different probability distributions to the costs in the options. This change is recorded as the tendency towards a gain with higher certainty (low risk, risk-aversion) or a loss with more uncertainty (high risk, risk-seeking). Other violations of expected utility theory like the Allais paradox are evidenced through experiments [1]. In a word, humans do not make decisions based on the optimization of the expected utility function. It is inferred that the expected utility functions must have ignored something that is important in human decision-making. Regret theory is an alternative for expected utility theory to explain the risk-seeking/aversion behaviors. Regret theory derives from the psychological idea that people feel regret when they think better off had they chosen the alternative options. Thus the theory asserts, when making a decision, people systematically predict the regret and bias certain options more in order to avoid it. Evidence indicates that most people indeed act as predicted by regret theory [39, 31]. What is more, in the study of teamwork, the overall performance of a team has been shown to be better if all the team members share similar mental models [41]. In a human-robot collaboration team, if the robot makes decisions exactly like the human worker does, the collaboration is more predictable for the human. The human, thus, is more satisfied and the overall performance increases. Adopting regret in the standard Bayesian sequential strategy for a robot is a natural ways to make the robot
decision-making more human-like. We propose a method to integrate regret analysis into
the optimal strategy.

If the manual (autonomous) mode is chosen when the actual state is \( X(\tilde{\phi}) = 1 \), see
Equations (2.25) and (2.26), then the conditional Bayes risk is given by \( R^A(\Delta, L|X(\tilde{\phi}) = 1, \text{run}) \) and \( R^M(\Delta, L|X(\tilde{\phi}) = 1, \text{run}) \), respectively. When a robot takes more observations,
the mutually alternative options are the manual mode and the autonomous mode. Regret
is predicted when actually choosing the manual (autonomous) mode and giving up the au-
tonomous (manual) mode. In other words, the regret associating with choosing the manual
(autonomous) mode depends on both the manual mode and the autonomous mode. There-
fore, the modified conditional Bayes risks are,

\[
\tilde{R}^A(\Delta, L|X(\tilde{\phi}) = 1, \text{run}) = R^A_1 + RGT(R^A_1 - R^M_1),
\]

\[
\tilde{R}^M(\Delta, L|X(\tilde{\phi}) = 1, \text{run}) = R^M_1 + RGT(R^M_1 - R^A_1),
\]

where, \( RGT(\cdot) \) is the regret function, \( R^A_1 \) and \( R^M_1 \) are the short notations for \( R^A(\Delta, L|X(\tilde{\phi}) = 1, \text{run}) \) and \( R^M(\Delta, L|X(\tilde{\phi}) = 1, \text{run}) \), respectively, with the understanding that the discussion
is restricted to the same \( \Delta, L \) and \( \text{run} \).

Due to the symmetry of the structure, without losing generality, we restrict the dis-
cussion to \( \tilde{R}^A_1 \), the short notation of \( \tilde{R}^A(\Delta, L|X(\tilde{\phi}) = 1, \text{run}) \), when exploring the properties
of the regret function. Indicated in [39], the regret function candidates must enable several
properties of the modified conditional Bayes risk function \( \tilde{R}^A_1 \). Firstly, \( \partial \tilde{R}^A_1 / \partial R^A_1 > 0 \), in-
dicating that \( \tilde{R}^A \) is increasingly biased if \( R^A \) increases. Secondly, \( \partial \tilde{R}^A_1 / \partial R^M_1 < 0 \), meaning
the increase of the rival conditional Bayes risk \( R^M_1 \) leads to the decrease of the bias towards
\( \tilde{R}^A_1 \). What is more, \( RGT(\cdot) \) is non-decreasing and \( RGT(0) = 0 \); The regret is 0 when the
disparity of \( R^A_1 \) and \( R^M_1 \) is 0, and not decreasing when \( R^A_1 \) gets larger than \( R^M_1 \). Based on the
knowledge, the following exponential function is a good candidate for the regret function [9],

\[ RGT(\xi) = \frac{g_2}{2} (1 - e^{-g_1 \xi}) \] (2.30)

where, \( \xi \) is the independent for the function, \( g_1 > 0 \) and \( g_2 > 0 \) are constant parameters.

The error function of \( \tilde{R}^A(\Delta, L | \mathbf{x}(\mathbf{o})) = 1, \text{run} \) and \( \tilde{R}^M(\Delta, L | \mathbf{x}(\mathbf{o})) = 1, \text{run} \) is defined as follows,

\[ \Psi(R_A^1 - R_M^1) = \tilde{R}^A(\Delta, L | \mathbf{x}(\mathbf{o}) = 1, \text{run}) - \tilde{R}^M(\Delta, L | \mathbf{x}(\mathbf{o}) = 1, \text{run}) \] (2.31)

Substitute Equations (2.28), (2.29) and (2.30) into Equation (2.31) to get,

\[ \Psi(R_A^1 - R_M^1) = R_A^1 - R_M^1 + g_2 \sinh(g_1 (R_A^1 - R_M^1)) \] (2.32)

Let \( \xi_A^M = R_A^1 - R_M^1 \), equation 2.32 becomes,

\[ \Psi(\xi_A^M) = \xi_A^M + g_2 \sinh(g_1 \xi_A^M) \] (2.33)

According to regret theory, \( \Psi(\xi_A^M) \) has to be monotonically increasing and convex when \( \xi_A^M \geq 0 \). Also it has to be skew-symmetric, \( \Psi(-\xi_A^M) = -\Psi(\xi_A^M) \). Figure 2.7 contains the plots of Equation (2.33) for different valued parameters \( g_1 \) and \( g_2 \). We can observe from the plots that the proposed function \( \Psi(\xi_A^M) \) possesses all the properties required by regret theory. Furthermore, the value of parameter \( g_1 \) and \( g_2 \) affects the convexity of function \( \Psi(\xi_A^M) \), implying different risk-seeking/risk-aversion attitudes of humans. This function also matches with the quantitative measurements of regret theory by [4], indicating the proposed regret function can be used.

We adopt this specific regret function. The explicit form of the modified conditional Bayes risk function is,

\[ \tilde{R}^A(\Delta, L | \mathbf{x}(\mathbf{o}) = 1, \text{run}) = R_A^1 + \frac{g_2}{2} (1 - e^{-g_1 (R_A^1 - R_M^1)}) \]. (2.34)
Figure 2.7: The proposed error function $\Psi(\xi^M)$ for different values of parameters in the function

Likewise,

$$
\tilde{R}^M(\Delta, L | X(\tilde{\omega}) = 1, \text{run}) = R^M_1 + \frac{g_2}{2} (1 - e^{-g_1(R^M_1 - R^A_1)}).
$$

Similarly, given that $X(\tilde{\omega}) = 0$ we have the modified conditional Bayes risks $\tilde{R}^A(\Delta, L | X(\tilde{\omega}) = 0, \text{run})$ and $\tilde{R}^M(\Delta, L | X(\tilde{\omega}) = 0, \text{run})$. The values of $g_1$ and $g_2$ are tuned through experiments. In this work, for simplification, $g_2$ is fixed as a constant, $g_1$ is regarded as the regret intensity factor. Qualitative measurement of human worker’s regret is conducted during the experiment to tune the value of $g_2$.

In the special case when no observation is taken, the modified expected Bayes risk function, like the original expected Bayes risk function, is given by Eqn. (2.27). Otherwise, the modified Bayes risk function is,

$$
\tilde{\text{risk}}(\Delta, L, P(X(\tilde{\omega}) = x(\tilde{\omega}); k), \text{run}) = 
\tilde{R}(\Delta, L | X(\tilde{\omega}) = 0, \text{run}) P(X(\tilde{\omega}) = 0; k) + \tilde{R}(\Delta, L | X(\tilde{\omega}) = 1, \text{run}) P(X(\tilde{\omega}) = 1; k).
$$

(2.36)
where $R = R^M$ for the manual mode and $R = R^A$ for the autonomous mode.

### 2.2.3.1 Illustration of Bayesian Decision-Making Approach

We choose different values of $\beta_r, \beta_H, c_r, c_H$ as listed in Table I to obtain several relative positions in the Bayes risk curves as in Figs.2-4. These modes without regret are the optimal decision-making schemes. Certain intersections of different decision curves represent the threshold probabilities $\pi_{m,L}, \pi_{m,U}, \pi_{a,L}, \pi_{a,U}$ respectively that divide the neighboring regions. Different decision regions are determined by the minimum Bayes risk curve and different threshold probabilities. An optimal decision about the observation mode is made based on which decision region the probability value of the part being right determined by the robot lies in.

Line 1 and line 2 are the risk functions for directly deciding the part as wrong and right respectively without any more observation. Line 3 and Line 6 are the risk functions for the robot and the human respectively making one more observation. Lines 4 and 5, and Lines 7 and 8 are the risk functions for the robot and the human respectively making two more observations. Based on these risk function curves, we find the minimum risk function represented by the red bold line. If the probability value of a part being right determined by the robot lies in region $a$, the final decision is that the part is wrong. If the probability lies in region $b$, the final decision is that the part is right. If the probability lies in region $c_1$ or $c_2$, the decision is that the robot makes one or two more observations respectively to make the final decision. If the probability lies in region $d_1$ or $d_2$, the decision is that the human makes one or two more observations respectively to help the robot with the observation information to make the final decision.
### Table 2.4: Parameters with chosen values to form Bayes risk curves

<table>
<thead>
<tr>
<th>Mode</th>
<th>$\beta_r$</th>
<th>$\beta_H$</th>
<th>$c_r$</th>
<th>$c_H$</th>
<th>$\pi_{m,l_2}$</th>
<th>$\pi_{m,l_1}$</th>
<th>$\pi_{m,l_1}$</th>
<th>$\pi_{a,l_2}$</th>
<th>$\pi_{a,l_1}$</th>
<th>$\pi_{a,l_1}$</th>
<th>$\pi_{a,l_2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual</td>
<td>0.50</td>
<td>0.78</td>
<td>0.02</td>
<td>0.04</td>
<td>0.195</td>
<td>0.383</td>
<td>0.616</td>
<td>0.804</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Autonomous</td>
<td>0.81</td>
<td>0.78</td>
<td>0.02</td>
<td>0.07</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0.110</td>
<td>0.435</td>
<td>0.565</td>
<td>0.890</td>
</tr>
<tr>
<td>Mixed</td>
<td>0.72</td>
<td>0.78</td>
<td>0.02</td>
<td>0.05</td>
<td>-</td>
<td>0.376</td>
<td>0.624</td>
<td>-</td>
<td>0.198</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**2.2.3.2 Illustration of Regret-Based Decision-Making Approach**

We here provide an example of regret analysis in the mixed mode, and similar illustrations are straightforward for autonomous and manual modes. Inclusion of regret in Bayes risk curve for the mixed mode may also result in the mixed mode (modified mixed mode), but with different probability regions. As illustrated in Fig.2.12 for the case when more manual observations are made in the mixed mode, the region $c_2$ in the modified mixed mode is smaller and the region $d_1$ is larger compared to that in the Bayesian optimal mixed mode. Similarly, we may illustrate the case in the mixed mode when more autonomous observations are made.

![Figure 2.8: (a) manual mode; (b) autonomous mode; (c) mixed mode](image-url)
2.2.4 Implementation Scheme of the Regret-Based Bayesian Decision Making Approach for the Assembly Task

2.2.4.1 The overall scheme in a flowchart

The overall regret-based Bayesian decision making scheme for the assembly task is shown in Fig. 2.9 as a flowchart.

As shown in the flowchart, the robot sensor at first detects whether the type of the input assembly part is correct or not, and displays the robot’s observation in its face screen. The probability of correctness in the robot’s sensing in the current time step is also displayed in the robot face screen. Then, the system constructs the Bayes risk curve using the current values of (i) detection probability of the robot sensor, (ii) detection probability of the human, (iii) cost induced by taking more observations under the autonomous mode, and (iv) cost induced by taking more observations under the manual mode. Then, depending on the minimum intersection points (Bayes risk, probability) between line 1 and other lines (3, 4, 5, 6), one of the three modes (manual mode, autonomous mode, mixed mode) of allocation of autonomy in the detection for the assembly task is selected. If the manual mode is selected, then depending on the detection probability of the robot sensor, three cases may happen as follows: (i) the robot decides the input part as wrong, (ii) the robot decides the input part as right, (iii) the robot asks the human to take one or two more observations and share the experiences with the robot to help the robot make the final decision about the correctness of the input assembly part. Similar phenomena happen if the autonomous mode is selected such as depending on the detection probability of the robot sensor, three cases may happen as follows: (i) the robot decides the input part as wrong, (ii) the robot decides the input part as right, (iii) the robot takes one or two more observations to make the final decision about the correctness of the input assembly part. If the mixed mode is selected, then depending on the detection probability of the robot sensor, the following
cases may happen: (i) the robot decides the input part as wrong, (ii) the robot decides the input part as right, (iii) the robot takes more one or two observations to make final decision about the correctness of the input part, and (iv) the robot asks the human to take one or two more observations and share the experiences with the robot to help the robot make the final decision about the correctness of the input assembly part.

If the input part is finally decided as the right, the robot picks it and manipulates it to the human, and the assembly task moves forward. However, if the input part is finally decided as the wrong, the robot does not manipulate it to the human. Instead, it issues a warning message to display on its face screen to make the human aware of the status of the input part. If the human replaces the wrong part by a right part, the robot picks it and manipulates it to the human, and the assembly task moves forward. While the assembly is going on, the experimenter verifies the decisions about the correctness of the input part. If the decisions are found correct, there is no action to take. However, if any decision is identified as wrong, the experimenter inputs this to the system, the assembly task stops temporarily, a window appears in the computer screen asking the human co-worker to measure his/her regret intensity for the wrong decisions. If the regret intensity is below a threshold, no additional action is required. However, if the regret intensity is above a threshold, another window appears in the computer screen to ask the human to know whether he/she is risk seeking or risk averse in the forthcoming assembly tasks. The regret intensity may be positive or negative based on the human’s risk seeking or risk aversive attitude. Then, based on the algorithms, Bayes risk is adjusted with the regret, new Bayes risk curve is constructed, and a different mode of allocation of autonomy in the detection may be selected based on the newly constructed Bayes curve characteristics (Bayes risk, probability).
2.2.4.2 Measurement of sensing probability and observation cost

The probability of correctness in the robot’s sensing ($\beta_i$) can be measured following Eqn. (2.37), where $n_w(k-1)$ is the number of wrong decision and $n_t(k-1)$ is the number of total decision made by robot $i$ in time step $k-1$.

$$\beta_i(k) = (1 - \frac{n_w(k-1)}{n_t(k-1)}) \times 100\% \quad (2.37)$$

Similarly, the probability of correctness in the human sensing ($\beta_H$) can be measured following Eqn. (2.38), where $n_{w0}(k-1)$ is the number of wrong observation and $n_{t0}(k-1)$ is the number of total observation made by the human in time step $k-1$.

$$\beta_H(k) = (1 - \frac{n_{w0}(k-1)}{n_{t0}(k-1)}) \times 100\% \quad (2.38)$$

The observation cost $c_H$ can be measured following Eqn. (2.39), where $T_H(k-1)$ is the total time (in seconds) needed by the human to make observations.

$$c_H(k) = \begin{cases} 0, & T_H(k-1) = 0s \\ 0.1, & T_H(k-1) = 10s \\ 0.2, & T_H(k-1) = 20s \\ \vdots & \vdots \\ 1.0, & T_H(k-1) \geq 100s \end{cases} \quad (2.39)$$

Similarly, the observation cost $c_I$ can be measured following Eqn. (2.40), where
\( T_i(k - 1) \) is the total time (in seconds) needed by the robot to make observations.

\[
c_i(k) = \begin{cases} 
0, & T_i(k - 1) = 0s \\
0.1, & T_i(k - 1) = 10s \\
0.2, & T_i(k - 1) = 20s \\
\vdots & \vdots \\
1.0, & T_i(k - 1) \geq 100s 
\end{cases} \tag{2.40}
\]

### 2.2.4.3 Measurement method for human regret intensity

Regret is the feeling of repentance over the decision about the detection the human prefers the robot to make. Regret in current time step has great impact on human’s preference/choice in the next time step—both positively and negatively. Regret happens when the human makes a preference about who should take additional observations during the detection if needed, but experiences negative feelings when he/she knows that the results might be better if he/she could prefer differently [39, 2]. In this subsection, we present a software-based approach of human regret measurement [4, 40], as follows:

In our scenario, regret measurement is event-based because regret measurement is needless when no mistake in the decision making is identified. Once a mistake is identified (by the experimenter in our case), the regret measurement window appears in the computer screen. The window as shown in Fig. 2.10 at first asks the human to choose whether he/she prefers the human (himself/herself) or the robot to prefer to take additional observations during the detection phenomena in usual condition. If the human chooses “prefer human”, a questionnaire regarding the human performance in additional observations during the detection appears as in Fig. 2.11. If the human chooses “prefer robot”, a similar questionnaire regarding the robot performance in additional observations during the detection appears. Then, the human responds the questionnaire in the window, and then the regret is auto-
matically calculated and normalized between 0 and 1. Here, if the human chooses “prefer human”, it indicates the human’s risk-averse tendency. Similarly, if the human chooses “prefer robot”, it indicates the human’s risk-seeking tendency.

Actually, the questionnaires are “Likert-Type Scales” [6, 50, 51] with 7 questions to measure regret. In this decision making approach regarding who will make additional observations if needed, we analyze regret along with disappointment—the feeling of sadness caused by the nonfulfillment of the human’s expectations in the decision making [40]. The human may experience disappointment when the human is overloaded and the robot decides about the detection with a low performance. In this case, the human does not feel regret about the choice due to the workload but is not satisfied with the results, which may affect the trust level of the human in the robot [28, 50].

In each case, the questionnaire contains 7 items. The items are explained below as an example when the human chooses “prefer human” (risk-averse case).

- Item 1 (I am not happy about the human’s performance in additional observations) is an “Affective reaction”.

- Item 2 (I wish I have chosen the robot to make the additional observations) is a “Regret counterfactual”. The human might have higher value on this item when the human feels like that the robot could make a better preference if it was chosen. In other words, the human is not satisfied with the outcomes of the detection, and he/she will experience regret about the preference.

- Item 3 (I wish the human has done the additional observations better) is a “Disappointment counterfactual”. The human might have higher value on this item due to low trust in the robot. In other words, although the human is not satisfied with the
outcomes, the human does not feel regret about the choice “prefer human” as the human has a relatively low trust in the robot.

- Item 4 (I feel responsible for my (human) choice/preference) is an “Internal attribution”.

- Item 5 (The human’s inattention or poor capability causes the mistake or low performance in the detection) is an “External attribution”

- Item 6 (I am satisfied with the overall detection performance) is a “Control item”.

If the preference result had higher values on this control item than on the affective reaction item (item 1, “I am not happy about the human’s performance in additional observations”), this questionnaire needs to be excluded from the subsequent analysis.

- Item 7 (Outcomes would have gone better if) is a “Choice between counterfactuals”.

Similar approach is followed to measure human regret intensity when/if the human chooses “prefer robot”.

2.2.5 Experimental Evaluation of the Regret-Based Bayesian Autonomy Allocation

2.2.5.1 Experimental Objective and Hypothesis

The objective is to evaluate the efficacy of the regret-based Bayesian decision-making approach. The adopted hypothesis is that the regret-based Bayesian suboptimal decision-making approach is more human-like than the Bayesian optimal decision approach, which may result in better HRI and collaborative assembly performance.
2.2.5.2 The Evaluation Scheme

We evaluate HRI and assembly performance for the collaborative assembly. Human-robot team fluency, human’s cognitive workload, and human’s trust in the robot are considered as the HRI criteria. Team fluency is assessed by the human co-worker subjectly following a Likert-type rating scale (1-least fluency level, 5-most fluency level) [50]. Team fluency is also measured objectively through functional delay, nonconcurrent activity time and idle time of robot and human as a percentage of total task time during the collaborative task. Trust is also assessed using the Likert-type rating scale (1-least trust level, 5-most trust level) [50]. Cognitive workload is measured following NASA TLX [51]. The assembly performance is expressed through efficiency and quality. Assembly efficiency ($\varepsilon$) is expressed in Eqn. (2.41), where $T_r$ is the actual assembly time and $T_s$ is the standard assembly time for a run (trial). Assembly quality ($\lambda$) is expressed in Eqn. (2.42), where $n_i$ is the total number of finished assembly with incorrect part and $n_f$ is the total number of finished assembly in a specified period of time.

$$\varepsilon\text{(run)} = \left( \frac{T_s}{T_r} \right) \times 100\%.$$  \hspace{1cm} (2.41)

$$\lambda\text{(period)} = \left( 1 - \frac{n_i}{n_f} \right) \times 100\%.$$  \hspace{1cm} (2.42)

2.2.5.3 Subjects

Twenty mechanical engineering students are recruited to participate in the experiments voluntarily. Ten subjects (Group I) participate in the collaborative assembly under regret-based Bayesian decision-making approach (“regret-based approach”). The remaining ten subjects (Group II) participate in the collaborative assembly under the Bayesian decision-making approach (without consideration of regret), which we call “Bayesian ap-
2.2.5.4 Experimental Procedures

In the first phase, each subject conducts some mock practices of the assembly with the robot. Then, in the second phase, the formal collaborative assembly is conducted by each subject as shown in Fig.1(a) following the collaboration scheme. Each subject in Group II and Group I separately performs the collaborative assembly for Bayesian approach and regret-based approach, respectively. The HRI and assembly performance are evaluated for these two approaches for each subject separately following the evaluation scheme.

2.2.6 EXPERIMENTAL RESULTS

Fig.2.13 shows that human-robot collaborative fluency significantly improves due to inclusion of regret in the decision-making. Fig.2.14 shows that both team fluency and human’s trust in the robot increase for the regret-based approach compared to the Bayesian approach. Inclusion of regret allocates new modes that reflect similar mental models of the robot and the human. In such case, coordination and synchronization between human and robot activities increase [41], which might have increased the team fluency.

Fig.2.15 shows that mean cognitive workload significantly reduces for the regret-based approach. We believe that similar mental model between human and robot [41], and allocation of observation mode that fits human psychology have reduced the cognitive workload. In the regret-based approach, the allocation of observations is more effective to make correct decision, which might have increased the quality. The regret assessment and regret-based reallocation of mode may take some additional time that may reduce the efficiency. However, the overall improvement in HRI especially reduction in functional delay and idle time of human and robot (Fig.2.13) and improvement in quality might have made
up the efficiency drop due to time loss during regret assessment and enhanced the overall efficiency. We believe that utilizing the favorable cognitive workload, team fluency and quality, the collaborative system may keep increasing the efficiency, which may be more visible for real applications when the assembly is continued for long hours. Increment in efficiency also proves better team fluency for the regret-based approach. As Fig.2.14 shows, good team fluency, favorable workload and high assembly quality in the collaboration for the regret-based approach might have contributed to enhance the human’s own trust in the robot.

Analyses of Variance (ANOVA) shows that variations in HRI (subjective and objective team fluency, trust, cognitive workload) and assembly performance (efficiency, quality) between the subjects are not statistically significant (p > 0.05 at each case), which indicate the generality of the results.
Figure 2.9: The overall implementation scheme in a flowchart
Figure 2.10: The human chooses whether he/she prefers the human (himself/herself) or the robot to take additional observations during the detection.

Figure 2.11: Questionnaire for assessing human’s regret for a case when the robot decides to ask the human to make observations, and the human made a wrong observation.
Figure 2.12: Regret-based modified mixed mode
Figure 2.13: Objective measure of human-robot team fluency between regret-based and Bayesian approaches. Less idle time, less functional delay time and less non-concurrent activity time indicate more team fluency.

Figure 2.14: Subjective measures of team fluency and human’s trust in the robot between regret-based and Bayesian approaches.
Figure 2.15: Cognitive workload and assembly performance (efficiency, quality) between regret-based and Bayesian approaches.
Chapter 3

Quantitative Trust Measurements

3.1 A Quantitative Trust Measure in Multi-Robot Systems with a Human-in-the-Loop

3.1.1 Problem Setup

An intelligence, surveillance, and reconnaissance (ISR) scenario is considered in which a team of \( N \) robots represented by an index set \( I_R = \{1, \ldots, N\} \subset \mathbb{Z} \), supervised by a human operator, must reach a set of \( M \) goal destinations represented by the set \( Goals = \{1, \ldots, M\} \subset \mathbb{Z} \) while avoiding collisions with stationary obstacles and with each other (i.e., mobile obstacles), as shown in Fig. 3.1. As is standard in robot motion planning, the workspace is discretized into \( Q \) regions or states represented by the set \( W = \{w_1, w_2, \ldots, w_Q\} \), which are labeled with relevant properties (e.g., whether they contain an obstacle or goal). A combination of human judgment, assistance, and permission might be needed in order to make the attempt, motivating the need for a human operator, as significant computational resources are needed for computation of paths even the region is discretized well.
Figure 3.1: Multiple robots must reach a set of destinations while avoiding obstacles and collisions with other robots, taking shorter but riskier paths between obstacles with human oversight when trusted to do so. That is, when the human operator trusts a robot, he/she plans for robot motion through visual feedback from a robot’s onboard camera using HMI such as keyboard, mouse, or joystick.

With respect to human interaction, a quantitative, probabilistic, and dynamic trust model based on robot performance, human performance, joint human-robot fault, human intervention, and feedback evaluation is used to estimate human trust in each of the robots throughout the scenario. This estimate of trust affects the specification decomposition, with more trusted robots assigned more destinations. Trust is also used to determine when the robot should suggest navigating between obstacles, as this requires real-time switching between manual and autonomous motion planning. Human consent for this switching is assumed to depend on the change of trust as well as whether or not the human is currently occupied with other tasks. Under manual motion planning, when the human trusts a robot, he/she will assign waypoints. In that case, rather than moving between cell centroids, the robot will navigate autonomously between successive waypoints. In our system, the human can see trust level, visual information, and support robot requests via the robot’s onboard camera, a HMI and/or graphical user interface (GUI) as depicted in Fig. 3.1. More details regarding the HMI and GUI designs will be provided in the simulation section 3.1.4.

Robots are good at local tasks given their limited computing, sensing, and communication capabilities. we allow the human to intervene if required to increase task efficiency. This combination of autonomous and manual motion planning allows the joint system to achieve the task efficiently but without overloading the human operator. As planning and
execution proceeds, human trust in each robot evolves dynamically, with the human choosing to collaborate more often with trusted robots. To reduce computational complexity, more trusted robots assigned more tasks and vice versa. To further integrate human intelligence while guaranteeing task safety, a trust-based real-time switching framework is developed, and an autonomous decision-making aid is designed. By default, the autonomous and safe motion planning is implemented to guarantee task completion based on an over-approximation of the task domain, ensuring that obstacles are avoided. However, if the human trusts a robot and is not overloaded, manual planning may be requested for more efficient but riskier solutions, e.g., moving between close obstacles. In such a case, the robot behavior under manual planning needs to be monitored so that if any event that will violate the task is detected, the autonomous planning will be activated again. The robot and human performance measurements, joint human-robot fault measurements, direct human intervention, as well as trust evaluation are used as feedback to update the trust model.
3.1.2 Computational Trust Model

Since human behaviors are notoriously difficult to model, predict, and verify and the task environment is usually complex and uncertain, probabilistic analysis must be utilized to capture these uncertainties in trust estimates. Our proposed integrated trust model is shown in Fig. 3.2. First, let $T_i(k)$ denote human trust in robot $i$, $i = 1, 2, \cdots, N$, which is a hidden random variable taking values from 0 to 1. This assumption is made because trust is difficult to measure directly in real-time and is usually measured subjectively after each experiment session. We use solid green ellipses to represent $T_i$ in the figure, indicating that this is a process evolving in real-time. The discrete trust state results in a standard DBN model. The actual realization and the sequence of the process, i.e., dynamic evolution of trust over time, is hidden. Based on our previous works involving creation of a time-series trust model [57, 52, 53], we identify three major factors impacting trust, i.e., robot performance $P_{R,i}$, human performance $P_{H}$, and joint human-robot system fault $F_i$. These factors are shown in solid yellow ellipses in the figure. Following the OPTIMo model, we also have the human inputs $m_i(k), c_i(k), f_i(k)$ represented by solid and dashed blue ellipses in the figure, with dashed ellipses indicating intermittent observations. This is because it might not be practical to have human inputs all the time. The term $m_i(k) \in \{0, 1\}$ represents human intervention (i.e., switches between manual and autonomous modes) in motion planning, and its default value of zero indicates no intervention. Hence, $m_i(k)$ can be measured and updated in real-time. The term $c_i \in \{-1, 0, +1\}$ represents change in trust as reported by the human, with -1 indicating a decrease in trust, 0 indicating no change, and +1 indicating an increase in trust. The term $f_i \in (0, 1)$ represents subjective trust feedback, which is a continuous value between 0 and 1. Both $c_i$ and $f_i$ only require occasional observations. That is, the participants will only be asked to provide trust change $c_i(k)$ and trust feedback $f(k)$ periodically. This ensures there is not much additional cognitive
workload for the human operator during multi-robot cooperative tasks.

The conditional probability distribution (CPD) of human trust in robot $i$ at time $k$ based on the previous trust value $T_i(k-1)$ and the above causal factors can be expressed as a Gaussian distribution with mean value $\mu_1(k)$ and covariance $\sigma_1(k)$:

$$p(T_i(k)|T_i(k-1), P_{R,i}(k), P_{R,i}(k-1), P_H(k), P_H(k-1), F_i(k), F_i(k-1)) = \mathcal{N}(T_i(k); \mu_1(k), \sigma_1(k))$$

(3.1)

where

$$\mu_1(k) = T_i(k-1) + B(P_{R,i}(k) - P_{R,i}(k-1)) + C(P_H(k) - P_H(k-1)) + D(F_i(k) - F_i(k-1))$$

(3.2)

Here, $\mu_1(k) \in (0,1)$ represents the mean value of human trust in a robot $i$ at time $k$, $P_{R,i} \in (0,1)$ represents performance of robot $i$, $P_H \in (0,1)$ represents human performance, $F_i \in (0,1)$ represents faults made in the joint human-robot system and $\sigma_1$ reflects the variance in each individual’s trust update. The coefficients $B, C, D$ can be determined by data collected from human subject tests (see the author’s previous work [52] for more details). Here, these coefficients are further scaled such that $\bar{T}_i$ is normalized. See Fig. 3.12 for an illustration of the dynamics of mean trust $\bar{T}_i$.

In this scenario, robot performance $P_{R,i}$ is modeled as a function of “rewards” the robot receives when it identifies an obstacle or reaches a goal destination:

$$P_{R,i}(k) = C_O \frac{N_{O_i}(k)}{\sum_{i=1}^{N} N_{O_i}(k)} + C_G \frac{N_{G_i}(k)}{\sum_{i=1}^{N} N_{G_i}(k)}$$

(3.3)

where $N_{O_i}$ and $N_{G_i}$ are the number of obstacles detected and goals reached by the robot $i$ up to time $k$, and $C_O \in (0,1)$ and $C_G = 1 - C_O \in (0,1)$ are corresponding positive rewards chosen such that $P_{R,i}(k)$ is normalized. This allows the robot to earn trust as it learns details of the environment.

Human performance is calculated based on workload and the complexity of the
environment surrounding the robot with which the human is currently collaborating. The concept of utilization ratio, $\gamma$, is used to measure workload [55]

$$
\gamma(k) = \gamma(k-1) + \frac{\sum_{i=1}^{N} m_i(k) - \gamma(k-1)}{\tau}
$$

(3.4)

where $m_i(k) = 1$ if the human is collaborating with robot $i$ and 0 otherwise, and $\tau$ can be thought of as the sensitivity of the operator. Assuming a human can only collaborate with one robot at a time, i.e., manually assigning paths through obstacles for the chosen robot, (3.4) allows workload to grow or decay between 0 and 1. Complexity of the environment is based on the number of obstacles that lie within sensing range $r_i$ of collaborating robot $i$ at time $k$. The human’s superior capability in creating more detailed paths will be enhanced in more complex environments, leading to increased performance in the presence of more obstacles. On the other hand, human performance decreases with respect to workload. Therefore, $P_H(k)$ can be modeled as follows:

$$
P_H(k) = \begin{cases} 
1 - \gamma(k)^{S_{oi}(k)+1} & \text{if } m_i(k) = 1 \\
1 - \gamma(k) & \text{if } m_i(k) = 0 
\end{cases}
$$

(3.5)

where $S_{oi}$ is the number of obstacles within sensing range of collaborating robot $i$, reflecting the environmental complexity. Fig. 3.3 shows the change of human performance with respect to workload $\gamma$ and environmental complexity $S_{oi}$.

Faults in the system are modeled as the “penalty” the robot receives when it enters an obstacle region or detects an obstacle on its planned path: $F_i(k) = - \frac{N_{Hi}(k)}{N_{Oi}(k)}$ where $N_{Hi}(k)$ is the total number of obstacle regions robot $i$ has entered before sensing the corresponding obstacle up to time $k$. Note that faults can originate from both the robot and the human, i.e., human trust in a robot will decrease even if the robot enters an obstacle region under manual motion planning.

Based on the DBN, we can quickly establish the filtered belief update of trust, i.e.,
\[ \text{filter}(T_i(k)) = p(T_i(k)|P_{R,i}(1:k), P_H(1:k), F_i(1:k), m_i(1:k), c_i(1:k), f_i(1:k), T_i(0)), \]

using the forward algorithm by applying the principle of dynamic programming to avoid incurring exponential computation time due to the increase of \( k \). We can first compute

\[
\overline{\text{bel}}(T_i(k), T_i(k - 1)) = p(m_i(k)|T_i(k), T_i(k - 1))p(c_i(k)|T_i(k), T_i(k - 1))p(f_i(k)|T_i(k)) \cdot \\
p(T_i(k)|T_i(k - 1), P_{R,i}(k), P_{R,i}(k - 1), P_H(k), P_H(k - 1), F_i(k), F_i(k - 1))\text{filter}(T_i(k - 1)),
\]

(3.6)

where \( p(m_i(k)|T_i(k), T_i(k - 1)) \) is the probability of human intervention, \( p(c_i(k)|T_i(k), T_i(k - 1)) \) is the probability of a trust change given current and prior trust, and \( p(f_i(k)|T_i(k)) \) is the probability of subjective trust evaluation, respectively, which can follow a similar sigmoid distribution as in [63]. The CPD of human intervention based on trust can be modeled as
latent trust where \( \sigma \) e \( \omega \) where follows

\[
p(m_i(k) = 1|T_i(k), T_i(k-1)) = \frac{1}{1 + \exp(-{(\omega_1 + \omega_2 T_i(k) + \omega_3 (T_i(k) - T_i(k-1)))})},
\]

\[
p(m_i(k) = 0|T_i(k), T_i(k-1)) = 1 - \frac{1}{1 + \exp(-{(\omega_1 + \omega_2 T_i(k) + \omega_3 (T_i(k) - T_i(k-1)))})},
\]

(3.7)

where \( \omega_1 \) and \( \omega_2 \) are positive weights and this CPD indicates higher willingness to collaborate with a robot (intervention in path planning) when the human trust is higher.

The CPD of trust change based on trust can be modeled as follows

\[
p(c_i(k) = 1|T_i(k), T_i(k-1)) = e_{ib} + (1 - 3e_{ib}) \frac{1}{1 + \exp(-{(v((T_i(k) - T_i(k-1)) - o)))}},
\]

\[
p(c_i(k) = -1|T_i(k), T_i(k-1)) = e_{ib} + (1 - 3e_{ib}) \frac{1}{1 + \exp(-{(v(-(T_i(k) - T_i(k-1)) - o)))}},
\]

\[
p(c_i(k) = 0|T_i(k), T_i(k-1)) = 1 - p(c_i(k) = 1|T_i(k), T_i(k-1)) - p(c_i(k) = -1|T_i(k), T_i(k-1)),
\]

(3.8)

where the offset in a change to latent trust \((T_i(k) - T_i(k-1))\), \(o\), the variability \(v\) and the idling bias error term \(e_{ib}\).

The CPD of subjective trust evaluation based on trust can be modeled as follows

\[
p(f_i(k)|T_i(k)) = N(f_i(k); T_i(k), \sigma_f(k)),
\]

(3.9)

where \( \sigma_f \) is the uncertainty in human’s subjective trust evaluation \( f_i(k) \) with respect to latent trust \( T_i(k) \).

Given all the past data, the filtered trust belief follows that

\[
\text{filter}(T_i(k)) = \frac{\int \text{bel}(T_i(k), T_i(k-1))dT_i(k-1)}{\int \int \text{bel}(T_i(k), T_i(k-1))dT_i(k-1)dT_i(k)}.
\]

(3.10)

The derivation can be found in Sec3.2.1. The network parameters for the DBN
such as the state transition probabilities and output probabilities can be learned by the well-known expectation maximization (EM) algorithm [43]. A separate trust model should be trained based on each user’s experience.

3.1.3 Real-Time Trust-Based Switching Between Manual and Autonomous Motion Planning

In this section, we utilize trust analysis in a real-time switching framework to enable switches between manual and autonomous motion planning. Although the autonomous motion planning is guaranteed to be correct, it is usually conservative due to overapproximation of the environment. So while the autonomous motion planning is safe, more efficient but riskier paths – in this case, paths between obstacles in adjacent regions – may exist. If the human trusts a robot’s ability in navigating between two obstacles, the human can choose to construct a more efficient path between the obstacles based on, e.g., sensory information about the obstacles supplied by the robot.

We define events according to the condition: \( \|x_i - x_o^j\| \leq r_o \), where \( x_i \) is the robot position, \( x_o^j \) is the obstacle position, and \( r_o \) is some minimally acceptable distance between the robot and the obstacle. If the robot come within this distance, an event will be detected. According to the mean trust equation (3.2), this event detection also leads to fault penalty and hence lowers trust in the robot, leading to a re-evaluation of the assigned tasks. The result is that other more trusted robots may be re-assigned some of the destinations that were originally assigned to the robot that generated the fault. Once this re-evaluation is performed, the operator is free to continue working with the same or another robot depending on the change of levels of trust. Once the event recognizer detects an event, it will send the information to the checker module. The checker checks whether or not the current execution of the system meets the specification. Based on the information received
from the checker, the decision module determines under which mode the system should run for motion planning. More specifically, the trust-based decision module first computes the trust belief distribution based on Equation (3.10) for each robot $i$, then the corresponding trust value that yields the maximum likelihood is obtained. Next, the current maximum likelihood trust is compared with the maximum likelihood trust at the previous time step and the change of trust value can be calculated. A request then sends to the human and he/she then collaborates with the robot that has the highest trust increase beyond a certain threshold. In case multiple robots have the same highest trust change, some priority criterion can be used to choose an individual robot. See Fig. 3.7 for an example of the GUI design used in our simulation for a robot requesting manual motion planning based on trust comparison.

### 3.1.4 Simulation

In this section, a set of simulations of the ISR scenario is used to demonstrate our methods. The simulation is conducted in Matlab with model checking performed using NuSMV. An example environment is shown in Fig. 3.4. A human subject can choose to collaborate with a robot using a gamepad (see Fig. 3.5) and replan a robot path using a mouse (see Fig. 3.6).

The obstacles in the environment are initially unknown by the robots until they are gradually sensed. The sensor range $r_i$ of a robot is marked by a dashed circle around it. In our simulations, we set the range in a way such that a robot can always observe the 8 neighboring cells around it. Once an obstacle is sensed, its position becomes known to that individual robot. The communication range of a robot is set to be the same as the sensing range. The obstacle information, planned path, and goal assignment information are communicated with other robots when they come within communication range. The goal of the
human-robot team is to successfully reach each goal while avoiding all collisions. For this
scenario, trust levels are assumed to be equal at the start of the simulation (equal to zero),
leading each robot to be assigned an equal number of goals.

Fig. 3.7 shows the GUI designs used in the simulation for human intervention and
collaboration. Fig. 3.7(a) shows the dynamic evolution of the maximum likelihood trust
for all robots and compares the change of trust with a preset threshold. Once the change
of maximum likelihood trust exceeds the threshold, the robot with the largest trust increase
will request manual motion planning (e.g., Robot 3 requests that the human chooses a path
for it as shown in the figure where the threshold is set as 0.07). Fig. 3.7(b) shows the
GUI for measuring the trust change $c_i$ to be used in the calculation of trust belief (3.10) for all robots where “Lose” corresponds to “-1”, “Unchange” corresponds to “0”, and “Gain” corresponds to “+1”. This GUI measure is shown to the human operator every 60 time steps in the simulation. Fig. 3.7(c) shows the GUI for measuring the trust feedback $f_i$, where “full distrust”, “medium trust”, “neutral”, “medium trust”, and “full trust” span the spectrum from 0 to 1 and is a continuous scale. This GUI measure is also shown to the human operator every 200 time steps. We finally show the simulation results of 3-robot system with a human-in-the-loop. Fig. 3.8 shows the final paths traveled by all robots under the trust-based switching framework. Figs. 3.9-3.11 show the human performance $P_H$, robot performance $P_{R_i}$, fault $F_i$, trust belief $bel(T_i(k))$, human intervention $m_i$, trust change $c_i$, and trust feedback $f_i$ for Robot 1-3. Note that human performance $P_H$ is the same for all robots. Fig. 3.12 shows the evolution of mean trust distribution over time.

At the beginning, Robot 1 senses an obstacle on its path and Robot 2 senses an ob-
Figure 3.6: Illustration of manual motion planning using a mouse. The human assigns waypoints starting from a, passing between two obstacles to point b, and then point c, d, and eventually reaching the goal e.

 obstacle on its left side. Therefore, in the first trust change question, Robot 1’s trust decreases by 1 and Robot 2’s trust increases by 1. That is to say, in the trust question, when a robot senses an obstacle on its path, the human subject should choose “Gain”. When a robot senses an obstacle on its path or meet another robot and change its path, one should choose “Lose”. Otherwise, one should keep choosing “Unchange”.

 When all three robots detect their first obstacles, the trust gain of Robot 3 goes over the threshold. Hence, it requests the human to intervene. The human then plans a new path for Robot 3. However, there is an unknown obstacle in the path. Once this unknown obstacle is sensed, a fault occurs and Robot 3’s trust drops. Robot 3 then gives its current goal to Robot 1 (whose trust value is the highest at this moment). Robot 3 continues to its last left goal.
Figure 3.7: GUI designs: (a) Comparison of robots’ maximum likelihood trust and request for manual motion planning, (b) Measure of trust change $c_i$, and (c) Measure of trust feedback $f_i$. 
Figure 3.8: Final paths for 3-robot system switching between manual and autonomous planning mode.

Figure 3.9: Robot 1: (a) Evolution of human performance \( P_H \), robot performance \( P_R \), and fault \( F_1 \), (b) trust belief \( bel(T_1(k)) \), (c) human intervention \( m_1 \), (d) trust change \( c_1 \), and trust feedback \( f_1 \).
Figure 3.10: Robot 2: (a) Evolution of human performance $P_H$, robot performance $P_R$, and fault $F_2$, (b) trust belief $bel(T_2(k))$, (c) human intervention $m_2$, (d) trust change $c_2$, and trust feedback $f_2$.

Figure 3.11: Robot 3: (a) Evolution of human performance $P_H$, robot performance $P_R$, and fault $F_3$, (b) trust belief $bel(T_3(k))$, (c) human intervention $m_3$, (d) trust change $c_3$, and trust feedback $f_3$. 

66
For Robot 1, at around the time step 120, $P_{R_1}$ increases and then $P_H$ drops with the trust belief increases and drops correspondingly. At around time step 540, the human intervenes and $P_H$ drops because of overload with corresponding trust decrease. At time step 690, $P_{R_1}$ increases, leading to trust increases as well. For Robot 3, at time step 245, a fault occurs and trust drops.

### 3.2 An Interdependent Multi-Robot Trust Model

The trust model presented in Sec. 3.1.2 is a Dynamic Bayesian Network (DBN) model for a specific robot, independent of other robots in the system. Therefore, in the
previous implementation, the proposed trust model is used to measure trust for each robot individually. In a multi-robot system, the evolvement of trust for each robot may be inter-dependent [33]. In this section, we further seek to find correlations among trust modeled for each individual robot.

Dynamic Bayesian Networks is used to infer beliefs about a sequence of latent trust states, based on the history of observations and human interventions and trust inputs, shown in Fig.3.2. We further extended it to multi-robot systems by modeling a interdependent multi-robot trust model, represented by a multi-stream DBNs. Firstly, we consider the 2-robot correlation case. The interdependent multi-robot trust model for 2 correlative robots is represented as a new multi-stream DBNs in Fig.3.13. The latent trust state of robot 1, $T_1(k)$, is not only related to the history of observations and human interventions and trust inputs of robot 1, but also related to the trust state of the robot 2 at last time step, i.e. $T_2(k - 1)$. Correspondingly, the latent trust state of robot 2 at time step $k$, $T_2(k)$, is also relate to the trust state of the robot 1 at last time step $T_1(k - 1)$. We only model the new trust model for robot 1 here. The modeling process for robot 2 is similar.

3.2.1 Derivation of Filtered Trust Beliefs

To find the form of the new filtered belief at the current time $k$, let $\text{filter}_1(T_i(k)) = p(T_1(k)|T_2(1:k-1), P_{R,1}(1:k), P_{H}(1:k), F_1(1:k), m_1(1:k), c_1(1:k), f_1(1:k), T_1(0))$.

We used the 2-Step Temporal Bayesian Networks (2TBN) Derivation approach in [62]. The Bayes’ rule, variable marginalization and variable independence properties in [34] are also used in the Derivation. Let $\text{Rule}_{B}(X,Y)$ represent the act between two condition probability distributions $p(X,Y|Z)$ and $p(X|Y,Z)$, following the Bayes’ rule, where $X,Y$ and $Z$ may refer one or multiple variables:

$$p(X,Y|Z) = \frac{\text{Rule}_{B}(X,Y)}{p(X|Y,Z)p(Y|Z)}.$$ (3.11)
Figure 3.13: Interdependent Multi-Robot (2 robots) Dynamic Bayesian Networks (DBNs) based model for dynamic, quantitative, and probabilistic trust estimates.
Let \( \text{Rule}_M(A) \) denote the variable marginalization rule of \( X \) into an arbitrary condition probability distribution \( p(Y|Z) \):

\[
p(Y|Z) \xrightarrow{\text{Rule}_M(X)} \int p(X|Y,Z)p(Y|Z)dX.
\] (3.12)

Also, let \( \text{Rule}_I(X \perp Y|Z) \) refers the variable independence rule of a conditional distribution \( p(X|Y,Z) \), where \( X \perp Y|Z \):

\[
p(X|Y,Z) \xrightarrow{\text{Rule}_I(X \perp Y|Z)} p(X|Z).
\] (3.13)

Let \( \text{Per}_1(1:k) \) represents the performances from the past till the current time step \( k \) \( P_R(1:k), P_H(1:k), F_1(1:k) \) and \( \text{Hum}_1(1:k) \) denotes the human inputs of robot 1 also from the past till the current time step \( k \) \( m_1(1:k), c_1(1:k), f_1(1:k) \). Although all beliefs are under the condition of a prior distribution over the latent \( T_1(0) \), the conditioning on prior distribution initial point is ignored generally in [34, 62]. Therefore \( T_1(0) \) is ignored.
in this work as well.

\[
\begin{align*}
\text{filter}_k(T_1(k)) &= p(T_1(k)|T_2(1:k), P_{R1}(1:k), P_{RH}(1:k), F_1(1:k), m_1(1:k), c_1(1:k), f_1(1:k), T_1(0)) \\
&= p(T_1(k)|T_2(1:k), \text{Per}_1(1:k), \text{Hum}_1(1:k))
\end{align*}
\]

\[
\begin{array}{l}
\text{Rule}_{(T_1(k), \text{Hum}_1(1:k))} \\
p(T_1(k), \text{Hum}_1(1:k) | \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1)) \\
p(T_1(k), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1)) \\
\end{array}
\]

\[
\begin{array}{l}
\text{Rule}_{(T_1(k-1))} \\
f \left( \int p(T_1(k-1|k), \text{Hum}_1(1:k) | \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1)) \right) dT_1(k-1) \\
\end{array}
\]

\[
\begin{array}{l}
\text{Rule}_{(\text{Hum}_1(1:k), T_1(k-1:k))} \\
f \left( \int \frac{p(T_1(k|k-1), T_1(k), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))}{p(T_1(k-1), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))} \right) dT_1(k-1) \\
\end{array}
\]

\[
\begin{array}{l}
\text{Rule}_{(\text{Per}_1(1:k), \text{Hum}_1(1:k-1))} \\
f \left( \int \frac{p(T_1(k|k-1), T_1(k), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))}{p(T_1(k-1), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))} \right) dT_1(k-1) \\
\end{array}
\]

\[
\begin{array}{l}
\text{Rule}_{(T_1(k-1):\text{Per}_1(1:k), \text{Hum}_1(1:k-1))} \\
f \left( \int \frac{p(T_1(k|k-1), T_1(k), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))}{p(T_1(k-1), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))} \right) dT_1(k-1) \\
\end{array}
\]

\[
\begin{array}{l}
\text{Rule}_{(\text{Hum}_1(1:k), T_1(1:k-1):T_1(1:k))} \\
f \left( \int \frac{p(T_1(k|k-1), T_1(k), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))}{p(T_1(k-1), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))} \right) dT_1(k-1) \\
\end{array}
\]

\[
\begin{array}{l}
\text{Rule}_{(T_1(1:k-1):T_2(1:k-1))} \\
f \left( \int \frac{p(T_1(k|k-1), T_1(k), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))}{p(T_1(k-1), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))} \right) dT_1(k-1) \\
\end{array}
\]

\[
\begin{array}{l}
\text{Rule}_{(T_1(1:k-1):T_2(1:k-1):\text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_1(1:k-2))} \\
f \left( \int \frac{p(T_1(k|k-1), T_1(k), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))}{p(T_1(k-1), \text{Per}_1(1:k), \text{Hum}_1(1:k-1), T_2(1:k-1))} \right) dT_1(k-1) \\
\end{array}
\]

71
where the integrand of the denominator of Eqn. (3.14) is equal to the numerator of the 2nd equation. Following the same derivation steps, we have

\[
\text{filter}_1(T_1(k)) = \frac{\int p(Hum_1(k)|T_1(k-1 : k))p(T_1(k)|Per_1(k-1 : k), T_2(k-1))\text{filter}_1(T_1(k-1))dT_1(k-1)}{\int \int p(Hum_1(k)|T_1(k-1 : k))p(T_1(k)|Per_1(k-1 : k), T_2(k-1))\text{filter}_1(T_1(k-1))dT_1(k-1)dt_1(k)},
\]

(3.15)

where the term \(p(T_1(k)|Per_1(k-1 : k), T_2(k-1))\) can be converted by using Rule_M:

\[
p(T_1(k)|Per_1(k-1 : k), T_2(k-1)) \xrightarrow{\text{Rule}_M(T_2(k-1))} \int p(T_1(k), T_2(k-1)|Per_1(k-1 : k), T_2(k-1))dT_2(k-1)
\]

\[
= \int p(T_1(k), T_2(k-1)|Per_1(k-1 : k))dT_2(k-1).
\]

(3.16)

The integrand of Eqn. (3.16) can be expressed as a bivariate Gaussian CPD:

\[
p(T_1(k)|T_1(k-1), P_h(k-1 : k), P_f(k-1 : k), T_2(k-1)) = N(T_1(k), T_2(k-1); \mu_1(k), \sigma_1^2)
\]

\[
= \frac{1}{2\pi\sigma_1^2\sqrt{1-\rho_{12}^2}} \exp \left( -\frac{(T_1(k) - \mu_1(k))^2 - 2\rho_{12}(T_1(k) - \mu_1(k))(T_2(k-1) - \mu_1(k)) + (T_2(k-1) - \mu_1(k))^2}{2(1-\rho_{12}^2)\sigma_1^2} \right),
\]

(3.17)

with mean \(\mu_1(k) = AT_1(k-1) + B_1P_h(k-1) - B_2P_h(k-1) + C_1P_h(k) - C_2P_h(k-1) +

72
\[ D_1 F_1(k) - D_2 F_1(k - 1) \], standard deviation \( \sigma_1 \) and correlation \( \rho_{12} \) between \( T_1(k) \) and \( T_2(k - 1) \).

Substituting Eqn. (3.16) into Eqn. (3.15), we have

\[
\text{filter}_1(T_1(k)) = \frac{\int \left( \frac{p(Hum_1(k)|T_1(k-1:k)) \cdot \int p(T_1(k), T_2(k-1)|Per_1(k-1:k))dT_2(k-1) \cdot \text{filter}_1(T_1(k-1))}{\int \int \left( \frac{p(Hum_1(k)|T_1(k-1:k)) \cdot \int p(T_1(k), T_2(k-1)|Per_1(k-1:k))dT_2(k-1) \cdot \text{filter}_1(T_1(k-1))}{dT_1(k-1)dT_1(k)} \right) \right) dT_1(k-1)}{\int \int \left( \frac{p(Hum_1(k)|T_1(k-1:k)) \cdot \int p(T_1(k), T_2(k-1)|Per_1(k-1:k))dT_2(k-1) \cdot \text{filter}_1(T_1(k-1))}{dT_1(k-1)dT_1(k)} \right) dT_1(k-1)dT_1(k)}. \tag{3.18}
\]

The joint belief over sequential latent states can be represented by

\[
\text{belief}_1(T_1(k-1:k)) = p(Hum_1(k)|T_1(k-1:k)) \int p(T_1(k), T_2(k-1)|Per_1(k-1:k))dT_2(k-1) \cdot \text{filter}_1(T_1(k-1)) \tag{3.19}
\]

Substituting 3.19 back into 3.18

\[
\text{filter}_1(T_1(k)) = \frac{\int \text{belief}_1(T_1(k-1:k))dT_1(k-1)}{\int \int \text{belief}_1(T_1(k-1:k))dT_1(k-1)dT_1(k)}. \tag{3.20}
\]

Filtered belief considers effects from observation and human inputs from the past and present. To further personalize the trust model for each participant accurately, we need to have a training session to train the participant based on the data from the entire training session.

### 3.2.2 Derivation of Smoothed Trust Beliefs

Assuming that the total time step of the training session is \( K \). Let \( \text{smooth}_1(T_1(k)) = p(T_1(k)|P_{R,1}(1:K), P_{H}(1:K), F_1(1:K), m_1(1:K), c_1(1:K), f_1(1:K), T_2(1:K)) \) represent...
the smoothed beliefs given the data from the entire session, where \( k \in 1 : K \). The smoothed trust belief at time step \( k \) considers not only the data from the past and present \( k \in 1 : k \), but also the data in the future \( k \in (k + 1) : K \). The derivation of the smoothed beliefs are based on the “forward filtering- backward smoothing” approach in [17].

**smooth\(_1\)(T\(_1\)(k))**

\[
\begin{align*}
= p(T_1(k) | P_{k+1}(1:K), P_{1+1}(1:K), F_1(1:K), m_1(1:K), c_1(1:K), f_1(1:K), T_2(1:K)) \\
= p(T_1(k) | Per_1(1:K), Hum_1(1:K), T_2(1:K)) \\
\text{Rules}(T_1(k+1)) \int p(T_1(k) | T_1(k+1), Per_1(1:K), Hum_1(1:K), T_2(1:K)) \, dT_1(k+1) \\
\text{Rules}(T_1(k), T_1(k+1)) \int \left( \frac{p(T_1(k) | T_1(k+1), Per_1(1:K), Hum_1(1:K), T_2(1:K))}{p(T_1(k+1) | Per_1(1:K), Hum_1(1:K), T_2(1:K))} \right) \, dT_1(k+1) \\
\text{Rules}(T_1(k), T_1(k+1), Per_1(1:K), Hum_1(1:K), T_2(1:K)) \int \frac{p(T_1(k) | T_1(k+1), Per_1(1:K), Hum_1(1:K), T_2(1:K))}{p(T_1(k+1) | Per_1(1:K), Hum_1(1:K), T_2(1:K))} \, dT_1(k+1) \\
= \int p(T_1(k) | T_1(k+1), Per_1(1:K), Hum_1(1:K), T_2(1:K)) \, dT_1(k+1),
\end{align*}
\]

(3.21)

where the first term of the integrand \( p(T_1(k) | T_1(k+1), Per_1(1:K), Hum_1(1:K), T_2(1:K)) \) can be converted as such:

\[
\begin{align*}
p(T_1(k) | T_1(k+1), Per_1(1:K), Hum_1(1:K), T_2(1:K)) \\
\text{Rules}(T_1(k), T_1(k+1), Hum_1(k+1)) \int p(T_1(k) | T_1(k+1), Hum_1(1:k+1), Per_1(k+1), Hum_1(1:k), T_2(1:k+1)) \, dT_1(k+1) \\
\text{Rules}(T_1(k+1)) \int \frac{p(T_1(k) | T_1(k+1), Hum_1(k+1), Per_1(1:k+1), Hum_1(1:k), T_2(1:k+1))}{p(T_1(k+1), Hum_1(1:k+1), Per_1(1:k+1), Hum_1(1:k), T_2(1:k+1))} \, dT_1(k+1) \\
\text{Rules}(T_1(k+1)) \int \frac{p(T_1(k) | T_1(k+1), Hum_1(k+1), Per_1(1:k+1), Hum_1(1:k), T_2(1:k+1))}{p(T_1(k), T_1(k+1), Hum_1(1:k), T_2(1:k+1))} \, dT_1(k+1) \\
= \int p(T_1(k) | T_1(k+1), Hum_1(1:k+1), T_2(1:k+1)) \, dT_1(k+1)
\end{align*}
\]

74
Substituting Eqn. (3.22) into Eqn. (3.21), we have

\[
\text{smooth}_1(T_1(k)) = \int \left( \frac{p(T_1(k)) T_2(k + 1), \text{Per}_1(1 : k + 1), \text{Hum}_1(1 : k), T_2(1 : k + 1)}{p(T_1(k)) T_2(k + 1), \text{Per}_1(1 : k + 1), \text{Hum}_1(1 : k), T_2(1 : k + 1)} \right) dT_1(k + 1)
\]

\[
= \int \left( \frac{p(\text{Hum}_1(1 : k + 1)|T_1(k), T_1(k + 1))}{p(T_1(k)|\text{Per}_1(1 : k), \text{Hum}_1(1 : k), T_2(1 : k + 1))} \right) dT_1(k + 1)
\]

\[
\text{filter}_1(T_1(k)) = \int \left( \frac{p(\text{Hum}_1(1 : k + 1)|T_1(k), T_1(k + 1))}{p(T_1(k)|\text{Per}_1(1 : k + 1), \text{Hum}_1(1 : k), T_2(1 : k + 1))} \right) dT_1(k + 1)
\]

\[
= \int \left( \frac{\text{belief}_1(T_1(k : k + 1))}{p(T_1(k + 1), \text{Hum}_1(1 : k)|\text{Per}_1(1 : k + 1), \text{Hum}_1(1 : k), T_2(1 : k + 1))} \right) dT_1(k + 1),
\]

where the denominator \( p(T_1(k + 1), \text{Hum}_1(1 : k + 1)|\text{Per}_1(1 : k + 1), \text{Hum}_1(1 : k), T_2(1 : k + 1)) \)
can be converted by using RuleM:

\[
p(T_1(k + 1), \text{Hum}_1(1 : k + 1)|\text{Per}_1(1 : k + 1), \text{Hum}_1(1 : k), T_2(1 : k + 1))
\]

\[
\text{RuleM}_1(T_1(k)) = \int p(T_1(k : k + 1)|\text{Per}_1(1 : k + 1), \text{Hum}_1(1 : k), T_2(1 : k + 1)) dT_1(k),
\]

(3.24)
where the integrand is equal to the numerator of the 1st equation in Eqn. (3.22). Followed the same steps, Eqn. (3.24) becomes

$$
p(T_1(k+1), Hum_1(k+1)| Per_1(1:k+1), Hum_1(1:k), T_2(1:k+1))
$$

$$
\text{Rule}_{M(T_1(k))} \left\{ \begin{array}{l}
p(Hum_1(k+1)|T_1(k), T_1(k+1)) \\
p(T_1(k+1)|T_1(k), Per_1(k:k+1), T_2(k)) \\
p(T_1(k)|Per_1(1:k), Hum_1(1:k), T_2(1:k-1))
\end{array} \right\} dT_1(k)
$$

$$
= \int \overline{belief_1}(T_1(k-1:k))dT_1(k).
$$

(3.25)

Substituting Eqn. (3.25) back into Eqn. (3.23).

$$
smooth_1(T_1(k)) = \int \frac{\overline{belief_1}(T_1(k+1))}{\int \overline{belief_1}(T_1(k+1))dT_1(k)} smooth_1(T_1(k+1))dT_1(k+1).
$$

(3.26)

To verify our new derived filtered and smoothed belief functions for 2-robot correlation cases, we followed the trust evaluation process in [63]. Firstly, we use a set of initial parameters shown in Table 3.1 to train a human to perform the motion planning simulation. Then, to personalize the trust model based on behaviors and trust preference of each individual, we use the Expectation Maximization (EM) algorithm [34] to optimize each parameters given some initial values. The EM algorithm finds the optimized parameters which maximize the total likelihood of the data.

<table>
<thead>
<tr>
<th>Initial</th>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$B$</th>
<th>$C$</th>
<th>$D$</th>
<th>$\sigma_m$</th>
<th>$\rho_{12}$</th>
<th>$\epsilon_{ib}$</th>
<th>$\nu$</th>
<th>$\sigma_f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robot 1</td>
<td>131.2</td>
<td>-157.1</td>
<td>-9887</td>
<td>0.5</td>
<td>0.5</td>
<td>-1</td>
<td>0.005</td>
<td>0.9</td>
<td>$1.063 \times 10^{-7}$</td>
<td>1277</td>
<td>0.0003</td>
</tr>
<tr>
<td>Robot 2</td>
<td>131.2</td>
<td>-157.1</td>
<td>-9887</td>
<td>0.5</td>
<td>0.5</td>
<td>-1</td>
<td>0.005</td>
<td>0.9</td>
<td>$1.063 \times 10^{-7}$</td>
<td>1277</td>
<td>0.0003</td>
</tr>
</tbody>
</table>

Table 3.1: Initial parameters

- Expectation steps: (1) calculate the $filter_1$ at the end of the training simulation forward in time, given that a uniform prior trust $filter_1(T_1(1)) = 1$; (2) calculate the $smooth_1$ for all data backward in time given that $smooth_1(T_1(K)) = filter_1(T_1(K))$; (3) take the expectation for each $smooth_1$ to get a single "likely" sequence of trust
• Maximization step: use this calculated sequence of trust states with other performance and human inputs to find the optimized parameters for each CPD separately.

<table>
<thead>
<tr>
<th>Optimized</th>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$B$</th>
<th>$C$</th>
<th>$D$</th>
<th>$\sigma_m$</th>
<th>$\rho_{12}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robot 1</td>
<td>100.0015</td>
<td>-316.6815</td>
<td>-9889.9</td>
<td>0.5002</td>
<td>0.4958</td>
<td>-0.5496</td>
<td>0.0038</td>
<td>0.7416</td>
</tr>
<tr>
<td>Robot 2</td>
<td>100.2953</td>
<td>-330.2513</td>
<td>-9939.7</td>
<td>0.5002</td>
<td>0.4959</td>
<td>-0.5496</td>
<td>0.0038</td>
<td>0.7389</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Optimized</th>
<th>$e_{ib}$</th>
<th>$\nu$</th>
<th>$\sigma_f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robot 1</td>
<td>0.0022</td>
<td>2593.1</td>
<td>0.1768</td>
</tr>
<tr>
<td>Robot 2</td>
<td>$1.0642 \times 10^{-7}$</td>
<td>1495.2</td>
<td>0.2646</td>
</tr>
</tbody>
</table>

Table 3.2: Optimized parameters

After the optimized parameters are found (Table 3.2), we substitute them into the simulation and let the same human user perform the same motion planning simulation again given the same experimental scenario. The results are shown in Fig. 3.14a, 3.14b, 3.15a and 3.15b, where we can see from that test sessions with the new set of parameters results in better trust trend.

Similarly, the interdependent trust model can be extended to $n$ robot correlations. Then, based on the multi-stream DBNs in Fig. 3.16, the $\text{filter}$ and $\text{smooth}$ beliefs for robot $i \in n$ are

\[
\text{filter}_{n,i}(T_i(k)) = \frac{\int \int \text{belief}_{n,i}(T_i(k-1 : k))dT_i(k-1) \cdot \int \text{belief}_{n,i}(T_i(k-1 : k))dT_i(k)}{\int \int \text{belief}_{n,i}(T_i(k-1 : k))dT_i(k-1) \cdot \int \text{belief}_{n,i}(T_i(k-1 : k))dT_i(k)},
\]

\[
\text{smooth}_{n,i}(T_i(k)) = \frac{\int \text{belief}_{n,i}(T_i(k-1 : k)) \cdot \int \text{smooth}_{n,i}(T_i(k+1))dT_i(k+1)}{\int \text{belief}_{n,i}(T_i(k-1 : k)) \cdot \int \text{smooth}_{n,i}(T_i(k+1))dT_i(k+1)},
\]

(3.27)
where the new $belief_{n,i}$ is

$$belief_{n,i}(T_i(k-1:k)) =$$

$$p(\text{Hum}_i(k)|T_i(k-1:k)) \cdot$$

$$\int \cdots \int \left( p(T_1(k-1), \cdots , T_{i-1}(k-1), T_i(k), T_{i+1}(k-1), \cdots , T_n(k-1)|\text{Per}_i(k-1:k)) \right) dT_1(k-1) \cdots dT_{n-1}(k-1) \cdot dT_{i}(k-1),$$

$$filter_i(T_i(k-1))$$  \hspace{1cm} (3.29)

where the integrand inside the $belief_{n,i}(T_i(k-1:k))$,

$$p(T_1(k-1), \cdots , T_{i-1}(k-1), T_i(k), T_{i+1}(k-1), \cdots , T_n(k-1)|\text{Per}_i(k-1:k)),$$  \hspace{1cm} (3.30)

follows a multivariate Gaussian Distribution:

$$N(T_1(k-1), \cdots , T_{i-1}(k-1), T_i(k), T_{i+1}(k-1), \cdots , T_n(k-1); \mu_i(k), \sigma_i).$$  \hspace{1cm} (3.31)

But the computation of a multivariate Gaussian distribution is always a difficult problem to solve. There exist some literature [22, 27, 23, 24] that may simplify this problem. Our future work will focus on the optimization of our interdependent multi-robot trust model, as well as the simplification of such multivariate distribution problem.
Figure 3.14: Robot 1: Evolution of $P_H, P_{R,1}, F_1, f_{\text{filter}_1}(T_1(k)), s_{\text{smooth}_1}(T_1(k)), m_1, c_1,$ and $f_1$. 
Figure 3.15: Robot 2: Evolution of $P_H$, $P_{R2}$, $F_2$, $filter_2(T_2(k))$, $smooth_2(T_2(k))$, $m_2$, $c_2$, and $f_2$. 
Figure 3.16: Interdependent Multi-Robot (n robots) Dynamic Bayesian Networks (DBNs) based model for dynamic, quantitative, and probabilistic trust estimates.
Chapter 4

Conclusions

In this thesis, we first proposed a quantitative measure of regret in real-time human-robot collaborative domain search tasks aiming to improve HRC and the overall task performance. We designed experiments following the proposed measurement approach to elicit data points and further to determine the utility functions $W$ and the $Q$ functions in regret theory. An individual-based parameterized regret model was introduced and verified by the data points measured from the experiments. The proposed regret model and measure meets all the properties requested by regret theory. Implementing the proposed regret model into BSD for HRC systems in domain search tasks embodies human-like decision-making behaviors among team members.

Then, we included regret in Bayesian sequential decision-making function for human-robot collaborative decision-making system in manufacturing assembly. The effectiveness of the human-like regret-based approach in increasing HRI and performance in collaborative assembly in flexible manufacturing is verified through real human-robot collaborative experiments. The experimental results showed that the proposed approach for allocation of autonomy in making observations in shared vision system is novel and helpful for increasing human-friendliness and performance in manufacturing especially in flexible light
assembly.

Furthermore, as human-robot trust affects the overall results of interaction, we sought to build a quantitative trust model to measure trust between a human and a robot. We integrated time-series trust model with a single-stream Dynamic Bayesian Network (DBN) based robot performance-centric trust model named OPTIMo and formulated a new quantitative trust model considering the human-robot joint performance. Then we embedded it in a multi-robot motion planning task with a human-in-the-loop. Moreover, as the proposed trust model estimated trust for each robot independently, we sought to find correlative trust among different robots. We first extended the single-stream DBN into a multi-stream DBN. Then, we formulate an interdependent multi-robot trust model for 2-robot case and $n$-robot general case based on the multi-stream DBN to measure correlative trust in the multi-robot systems. We tested the new trust model for 2-robot correlation case in the motion planning simulation. The proposed interdependent multi-robot trust model can measure correlative trust in multi-robot systems.

Future research will focus on more in-depth analysis of the elicited functions and regret model will also be investigated. We will also search for more objective criteria to evaluate the regret, HRI and collaborative performance and develop regret-based control algorithm for HRC in manufacturing assembly. Moreover, we will try to optimize our interdependent multi-robot trust model to measure trust more accurately and efficiently.
Bibliography


