Safe Exploration: Addressing Various Uncertainty Levels in Human Robot Interactions

Changliu Liu and Masayoshi Tomizuka

***Abstract*— To address the safety issues in human robot interactions (HRI), a safe set algorithm (SSA) was developed previously. However, during HRI, the uncertainty levels are changing in different phases of the interaction, which is not captured by SSA. A safe exploration algorithm (SEA) is proposed in this paper to address the uncertainty levels in the robot control. To estimate the uncertainty levels online, a learning method in the belief space is developed. A comparative study between SSA and SEA is conducted. The simulation results confirm that SEA can capture the uncertainty reduction behavior which is observed in human-human interactions.**

I. Introduction

Recent advances in robotics suggest that human robot interaction (HRI) is no longer a fantasy, but is happening in various fields, such as industrial cooperative robots, autonomous cars and medical robots.

Human safety is one of the biggest concerns in HRI. Various robot control methods have been developed to ensure a collision free HRI, such as potential field methods [1], numerical optimization approaches [2], receding horizon control (RHC) methods [3] and sliding mode methods [4]. However, as humans will respond to the robot's movement, interactions need to be considered explicitly in the design of the robot controller. In other words, the robot should be equipped with the ability to conduct social behavior.

As the study of human brains inspires modern AI, a close look at the interactions among humans may also offer new ideas in dealing with HRI. The social behavior of a person is usually constrained by the social norm and the uncertainties that he perceives for other people. The social norm can be viewed as a hard constraint which cannot be bypassed. But the uncertainties can be attenuated through active learning (refer to uncertainty reduction theory [5]). A common experience is: a newcomer tends to behave conservatively in a new environment, due to large uncertainties. But through observing and learning his peers, he will gradually behave freely, due to the reduction of uncertainties.

In the previous work [6], the authors introduced a safe set algorithm (SSA) in designing the robot controller for safe HRI. In this method, a safe set is constructed and the control law is selected such that the safe set is time-invariant. This set can be viewed as a "social norm", which is a hard constraint on the robot's behavior.

One of the limitation of SSA is that the robot control is independent of the perceived uncertainties of humans' behavior, which are essential for a smooth and efficient HRI (regarding the effect of uncertainty reduction by active robot learning). It is noted that some researchers incorporated the information on uncertainties in solving HRI problems, e.g. in the framework of partially observable Markov decision process (POMDP) [7]. In this paper, SSA is extended to a safe exploration algorithm (SEA), where the levels of uncertainties are considered explicitly in the control strategy.

The remainder of the paper is organized as follows: in section II, a multi-agent interaction model will be introduced, followed by the discussion of the safe set algorithm (SSA) in section III. The safe exploration algorithm (SEA) will be proposed in section IV, followed by a new learning algorithm in section V. Finally, a comparative study between SSA and SEA will be conducted in section VI. Section VII concludes the paper.

C.Liu and M.Tomizuka are with University of California, Berkeley, CA 94720 USA (e-mail: changliuliu@berkeley.edu, tomizuka@me.berkeley.edu).

II. Modeling the Human Robot Interactions

The HRI is modeled in a multi-agent framework, where robots and humans are regarded as agents. In this section, an agent model will be introduced first, followed by the system model (refer to [6] for more detailed discussions).

A. The Agent Model

Suppose there are N agents in the system and are indexed from 1 to N. R denotes the set of all indices for the robots, and H for the humans. Denote agent i's state as x_i, its control input as u_i, its information set as π_i and its goal as G_i, for $i = 1, ..., N$. For simplicity, write x_R as the union of the states of all robots and x_H as the union of the states of all humans. Denote the system state as $x = [x_R^T, x_H^T]^T \in X$ where X is the state space of the system.

Assume that agents are independent to each other. Thus the open loop dynamics of x_i will not be affected by other agents, but be determined by x_i, u_i and the noise w_i, i.e.

$$\dot{x}_i = f_i(x_i, u_i, w_i), \forall i = 1, ..., N \tag{1}$$

Moreover, it is assumed that the dynamics of all robots are affine in the control term, i.e.

$$\dot{x}_i = f_{ix}^*(x_i, w_i) + f_{iu}^*(x_i) u_i, \forall i \in R \tag{2}$$

Agent i chooses the control u_i based on the information set π_i and its goal G_i. The information set is a combination of the measured data and the communicated information. In this paper, it is assumed that there is no direct communication among the defined agents. If a group of agents do communicate with each other, it is assumed that they can be coordinated and will be treated as one agent. In this

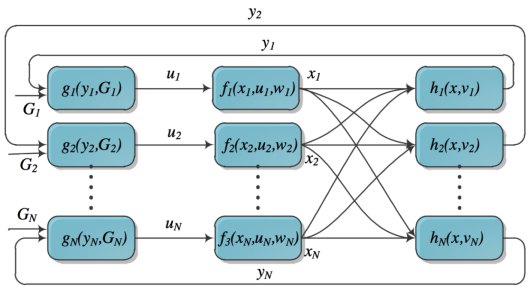


Fig. 1: A Multi-Agent System Block Diagram

way, agent i's information set at time T contains all the measurements up to time T, i.e. $\pi_i(T) = \{y_i(t)\}_{t\in[t_0,T]}$ where

$$y_i = h_i(x, v_i), \forall i = 1, ..., N \tag{3}$$

and v_i is the measurement noise. The controller can be written as

$$u_i = g_i(y_i, G_i), \forall i = 1, ..., N \tag{4}$$

B. The Closed Loop System

Based on (1) (3) (4), the system block diagram for the multi-agent system can be plotted as in Fig.1, where every row represents one agent. Using (3) (4) in (1) and using the system state x to represent the combination of all agents' states, the closed loop dynamic equation can be written as

$$\dot{x} = f^{cl}(x, G, v_1, ..., v_N, w_1, ...w_N) \tag{5}$$

where $G = \left[G_1^T, ..., G_N^T\right]^T \in X$ is the system goal. Notice that due to measurement feedbacks in control, all agents are coupled together in the closed loop system.

C. The Safety Oriented Design

The design method for the robot controller needs to be studied. Denote the system's safe set as X_S, which is a closed subset of the state space X that is collision free.

The Safety Principle: the function $g_i(.,.), i \in R$ should be chosen such that $\forall t$, $x(t) \in X_S$, 1) for all possible noises $v_1, ...v_N, w_1, ..., w_N$ and human decisions $g_i(.,.), i \in H$ if they are bounded, or 2) for almost all possible noises and human decisions if they are unbounded (those with negligible probabilities will be ignored).

III. The Safe Set Algorithm (SSA)

The safe set algorithm offers a fast online solution concerning the safety principle. In this method, the interaction is regarded as a sequential game, where robots are followers who play reactive strategies [8]. The system in Fig.1 can be reduced to the system in Fig.2, by 1) considering one effective robot and 2) combining all the blocks for human agents to be the closed loop block $\dot{x}_H = f_H^{'}(x_H, x_R, G_H, v_H, w_H)$. Then the multi-agent system is simplified as a two agent system with one effective robot and one effective human. The human's closed loop dynamics are estimated online using a parameter adaptation algorithm, to be discussed in section III-B. The control law of the robot is calculated with respect

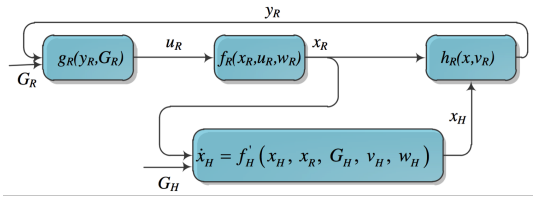


Fig. 2: Simplified System Model

to an invariant subset of the safe set X_S, to be discussed in section III-A.

A. The Set of Safe Control

To guarantee safety, the designed control law should make the safe set invariant. A safety index is introduced as $\phi : X \to \mathbb{R}$, a function on the state space such that 1) ϕ is differentiable with respect to t, i.e. $\dot{\phi} = (\partial\phi/\partial x)\dot{x}$ exists everywhere; 2) $\partial\dot{\phi}/\partial u_R \neq 0$; 3) the unsafe set X_S^c is not reachable given the control law $\dot{\phi} < 0 \; when \; \phi \geq 0$ and the initial condition $x(t_0) \in X_S$.

Existence Lemma of the Safety Index: The function ϕ satisfying all three conditions exists for any $X_S = \{x : d(x) \geq d_{min}\}$, where $d(x)$ is a function on the state space that calculates the minimum distance among all agents and $d_{min} \in \mathbb{R}^+$ is the threshold.[1]

To ensure safety, the robot's control must be chosen from the set of safe control $U_S^R(t) = \{u_R(t) : \dot{\phi} \leq -\eta_R \; when \; \phi \geq 0\}$ where $\eta_R \in \mathbb{R}^+$ is a safety margin. By Eq.(2), the set of safe control can be written as

$$U_S^R(t) = \{u_R(t) : L(t)\, u_R(t) \leq S(t)\} \tag{6}$$

where

$$L(t) = \frac{\partial\phi}{\partial x_R} f_{Ru}^* \tag{7}$$

$$S(t) = \begin{cases} -\eta_R - \sum_{j\in H} \frac{\partial\phi}{\partial x_j}\dot{x}_j - \frac{\partial\phi}{\partial x_R} f_{Rx}^* & \phi \geq 0 \\ \infty & \phi < 0 \end{cases} \tag{8}$$

In the following arguments, when there is no ambiguity, $S(t)$ denotes the value in the case $\phi \geq 0$ only. Since the dynamics of $\dot{x}_H$ is not known, estimation is needed in $S(t)$. Meanwhile, a constant $\lambda_R^{SSA} \in \mathbb{R}^+$ is introduced to bound the noises and uncertainties in the estimation, i.e.

$$S^{SSA}(t) = -\eta_R - \lambda_R^{SSA} - \sum_{j\in H} \frac{\partial\phi_0}{\partial x_j}\hat{\dot{x}}_j - \frac{\partial\phi_0}{\partial x_R} f_{Rx}^* \tag{9}$$

where $\hat{\dot{x}}_j = \frac{\hat{x}_j(k+1|k) - \hat{x}_j(k|k)}{T_s}, \forall j \in H$ and $\hat{x}_j(k+1|k), \hat{x}_j(k|k)$ are defined in Table I. k represents the time step of the last measurement before t, while $k+1$ is the time step for the next anticipated measurement. T_s is the sampling time. $L(t), S(t)$ will also be written as $L(k), S(k)$ to denote that the last measurement is taken at k-th time step.

[1]The Lemma is proved in [6]. ϕ can be constructed in the following procedure: first, define $\phi_0(x) = d_{min} - d(x)$; then check the order from ϕ_0 to u_R in the lie derivative sense, denote it by n; then define ϕ as $\phi_0 + k_1\dot{\phi}_0 + ... + k_{n-1}\phi_0^{(n-1)}$. The coefficients $k_1, ..., k_n$ are chosen such that the roots of $1 + k_1 s + ... + k_{n-1}s^{n-1} = 0$ all lie on the negative real line.

B. Online Learning and Prediction of Humans' Dynamics

In order to learn human behavior and make predictions, the nonlinear continuous time dynamic equation of the human agent $\dot{x}_H = f'_H(.)$ is linearized and discretized:

$$x_H(k+1) = A_H(k)x_H(k) + B_H(k)u_H^{cl}(k) + w_H^*(k) \quad (10)$$

where $u_H^{cl}(k) = \left[\hat{x}_R(k|k)^T, G_H^T(k)\right]^T$[2]. $\hat{x}_R$ is the estimate of the robot state from a state estimator (i.e. Kalman Filter). $A_H(k)$, $B_H(k)$ are time varying parameters. $w_H^*(k)$ is a noise term assumed to be zero-mean Gaussian and white. Assume that the robot's measurement of the human is:

$$y_R^H(k) = x_H(k) + v_R^H(k) \quad (11)$$

Equations (10) and (11) form a linear time varying (LTV) Gaussian system with unknown parameters. A parameter adaptation algorithm (PAA) is developed to identify the system online. The notations for the PAA is shown in Table I. Define $\hat{A}_H(k), \hat{B}_H(k)$ to be the estimates of the matrices given the information up to the k-th time step.

1) State Estimation: At $k+1$-th time step, $\hat{x}_H$ is first updated according to the closed loop dynamics in (12). Then the measurement information is incorporated in the *a posteriori* estimate in (13). A constant update gain $\alpha \in (0,1)$ is chosen to ensure that the measurement information is always incorporated.

$$\hat{x}_H(k+1|k) = \hat{A}_H(k)\hat{x}_H(k|k) + \hat{B}_H(k)u_H^{cl}(k) \quad (12)$$

$$\hat{x}_H(k+1|k+1) = (1-\alpha)\hat{x}_H(k+1|k) + \alpha y_R^H(k+1) \quad (13)$$

2) Parameter Estimation: The closed loop matrices are estimated using recursive least square (RLS) PAA as shown in (14):

$$\left[\hat{A}_H(k+1), \hat{B}_H(k+1)\right] = \left[\hat{A}_H(k), \hat{B}_H(k)\right] + \left(\hat{x}_H(k+1|k+1) - \hat{x}_H(k+1|k)\right)\varphi(k)^T F(k+1) \quad (14)$$

where $\varphi(k) = \begin{bmatrix} \hat{x}_H(k|k)^T & u_H^{cl}(k)^T \end{bmatrix}^T$. F is the learning gain with the update equation:

$$F(k+1) = \frac{1}{\lambda}\left[F(k) - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{\lambda + \varphi^T(k)F(k)\varphi(k)}\right] \quad (15)$$

where $\lambda \in (0,1)$ is a forgetting factor.

C. A Projection to the Set of Safe Control

Suppose there is a baseline control law $u_R^o(t)$ for the robot R, which drives the robot to its goal without safety considerations. Then $u_R^o(t)$ is mapped to the set of safe control $U_S^R(t)$ according to the following cost function

$$u_R^* = \min_{u_R \in U_S^R} J_R(u_R) = \frac{1}{2}(u_R - u_R^o)^T Q(u_R - u_R^o) \quad (16)$$

[2]Methods for inferring $G_H(k)$ are discussed in [9]. In this paper, we assume it is known.

where Q is positive definite. Solving (16), the safe control law $u_R^*(t)$ is in the following form

$$u_R^*(t) = u_R^o(t) - c\frac{Q^{-1}L(t)^T}{L(t)Q^{-1}L(t)^T} \quad (17)$$

where $c = \min_{u \in U_S^R(t)} |L(t)(u - u_R^o(t))|$ [6].

IV. The Safe Exploration Algorithm (SEA)

One of the limitation of SSA is that the bound for the uncertainties (i.e. λ_R^{SSA}) is a constant. However, the mean squared estimation error (MSEE) of the human's state is changing from time to time. A larger bound is needed if the MSEE is larger. To capture this property, the safe exploration algorithm (SEA) is introduced, where the control strategy changes for different levels of uncertainties.

A. The Belief Space

A belief space [3] is introduced in the safe exploration algorithm, where $\forall j \in H$, the state estimate of x_j is no longer a point in the Euclidean space, but a distribution, i.e. $\mathcal{N}(\hat{x}_j, X_j)$, where X_j is the covariance. All the distributions are assumed to be Gaussian. Since $x_j \sim \mathcal{N}(\hat{x}_j, X_j)$, the covariance can be written as

$$X_j = E\left[(x_j - \hat{x}_j)(x_j - \hat{x}_j)^T\right] \quad (18)$$

which is the mean squared estimation error (MSEE).

The definition of the *a priori* and *a posteriori* estimates, estimation errors and MSEEs are shown in Table I, where $\tilde{x}_j(k|k) = x_j(k) - \hat{x}_j(k|k)$ and $\tilde{x}_j(k+1|k) = x_j(k+1) - \hat{x}_j(k+1|k)$. At the k-th time step, the best prediction for $x_j(k+1)$ has the following distribution

$$\mathcal{N}(\hat{x}_j(k+1|k), X_j(k+1|k)) \quad (19)$$

B. The Safe Set in the Belief Space

As discussed in section III-A, the safe control at time step $t = kT_s$ needs to satisfy (6-8) where $\sum_{j\in H}\frac{\partial\phi}{\partial x_j}\dot{x}_j \approx \frac{1}{T_s}\sum_{j\in H}\frac{\partial\phi}{\partial x_j}[x_j(k+1) - x_j(k)]$. In the belief space, since the prediction of $x_j(k+1)$ is unbounded, the inequality in (6) is ill-defined. Indeed, a probability constraint is needed, i.e.

$$P(\{x_j(k+1) : L(k)u_R \le S(k)\}) \ge 1 - \epsilon, \forall j \in H \quad (20)$$

where $\epsilon > 0$ is a small number. When (20) holds, (6) is satisfied almost surely (ϵ cannot be 0 since the noises are unbounded in Gaussian distributions). Moreover, (20) is equivalent to the following optimization problem

$$L(k)u_R \le S^{SEA}(k) = \min_{x_j(k+1)\in\Gamma_j(k), \forall j\in H}\{S(k)\} \quad (21)$$

TABLE I: Definitions of Notations in State Estimation

	State Estimate	Estimation Error	MSEE
a posteriori	$\hat{x}_j(k\|k)$	$\tilde{x}_j(k\|k)$	$X_j(k\|k)$
a priori	$\hat{x}_j(k+1\|k)$	$\tilde{x}_j(k+1\|k)$	$X_j(k+1\|k)$

where $\Gamma_j(k)$ is a set of possible x_j such that the probability density of $x_j \notin \Gamma_j(k)$ is small and $P(\Gamma_j(k)) \geq 1-\epsilon$.

For a Gaussian distribution, the probability mass lying within the 3σ deviation is 0.997. Set $\epsilon = 0.003$, and define

$$\Gamma_j(k) = \left\{ x_j : \Delta x_j^T X_j (k+1|k)^{-1} \Delta x_j \leq 9 \right\} \quad (22)$$

where $\Delta x_j = x_j - \hat{x}_j(k+1|k)$. By (8), the RHS of (21) can be decoupled as a sequence of optimization problems, i.e. $\forall j \in H$,

$$\begin{array}{ll} \min_{x_j(k+1)} & \frac{\partial \phi}{\partial x_j} x_j(k+1) \\ s.t. & x_j(k+1) \in \Gamma_j(k) \end{array} \quad (23)$$

By Lagrangian method[3], the optimal solution $x_j^*(k+1)$, $\forall j \in H$ is obtained as

$$x_j^*(k+1) = \hat{x}_j(k+1|k) + \frac{3X_j(k+1|k)\left(\frac{\partial \phi}{\partial x_j}\right)^T}{\left[\left(\frac{\partial \phi}{\partial x_j}\right) X_j(k+1|k) \left(\frac{\partial \phi}{\partial x_j}\right)^T\right]^{\frac{1}{2}}} \quad (27)$$

Using (27), $S^{SEA}(k)$ can be expressed as

$$S^{SEA}(k) = -\eta_R - \lambda_R^{SEA}(k) - \sum_{j \in H} \frac{\partial \phi}{\partial x_j} \hat{\dot{x}}_j - \frac{\partial \phi}{\partial x_R} f_{Rx}^* \quad (28)$$

$$\lambda_R^{SEA}(k) = \frac{3}{T_s} \sum_{j \in H} \left[\left(\frac{\partial \phi}{\partial x_j}\right) X_j(k+1|k) \left(\frac{\partial \phi}{\partial x_j}\right)^T \right]^{\frac{1}{2}} + \lambda_R^o \quad (29)$$

where $\hat{\dot{x}}_j = \frac{\hat{x}_j(k+1|k) - \hat{x}_j(k|k)}{T_s}, \forall j \in H$, and $\lambda_R^o \in \mathbb{R}^+$ is the bound for other uncertainties. All other equations follow from the safe set algorithm except for the learning and prediction part, where new methods are needed to estimate $X_j(k+1|k)$ online.

V. Learning in the Belief Space

In this section, the MSEE propagation algorithm in parameter adaptation will be proposed, followed by the discussion of its application to human motion prediction.

A. Propagation of a priori MSEE in PAA

Suppose the LTV system is

$$x(k+1) = A(k)x(k) + B(k)u(k) + w(k) \quad (30)$$

where $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$ and $w(k)$ is the dynamic noise assumed to be zero-mean, Gaussian and white with covariance W. It is assumed that the state $x(k)$ is fully known. $A(k), B(k)$ are the unknown parameters that need to be estimated online. Define the parameter matrix $C(k) = [A(k), B(k)] \in \mathbb{R}^{n \times (n+m)}$, the data vector $\varphi(k) = \left[x^T(k), u^T(k)\right]^T \in \mathbb{R}^{n+m}$. Since random matrices are hard to deal with, transform matrix $C(k)$ to a vector $\theta(k)$. Suppose the row vectors in $C(k)$ are $C_1(k), C_2(k), \ldots, C_n(k) \in \mathbb{R}^{1 \times (n+m)}$. Define

$$\theta(k) = [C_1(k), C_2(k), ..., C_d(k)]^T \in \mathbb{R}^{n(n+m) \times 1} \quad (31)$$

Define a new data matrix $\Phi(k)$ as

$$\Phi(k) = \begin{bmatrix} \varphi^T(k) & 0 & \cdots & 0 \\ 0 & \varphi^T(k) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \varphi^T(k) \end{bmatrix} \in \mathbb{R}^{n \times n(n+m)} \quad (32)$$

Using $\Phi(k), \theta(k)$, the system dynamics can be written as

$$x(k+1) = \Phi(k)\theta(k) + w(k) \quad (33)$$

Let $\hat{\theta}(k)$ be the estimate of $\theta(k)$ and $\tilde{\theta}(k) = \theta(k) - \hat{\theta}(k)$ be the estimation error.

1) State estimation: The *a priori* estimate of the state and the estimation error is

$$\hat{x}(k+1|k) = \Phi(k)\hat{\theta}(k) \quad (34)$$

$$\tilde{x}(k+1|k) = \Phi(k)\tilde{\theta}(k) + w(k) \quad (35)$$

Since $\hat{\theta}(k)$ only contains information up to the $(k-1)$-th time step, $\tilde{\theta}(k)$ is independent of $w(k)$. Thus the *a priori* MSEE $X_{\tilde{x}\tilde{x}}(k+1|k) = E\left[\tilde{x}(k+1|k)\tilde{x}(k+1|k)^T\right]$ is

$$X_{\tilde{x}\tilde{x}}(k+1|k) = \Phi(k) X_{\tilde{\theta}\tilde{\theta}}(k) \Phi^T(k) + W \quad (36)$$

where $X_{\tilde{\theta}\tilde{\theta}}(k) = E\left[\tilde{\theta}(k)\tilde{\theta}(k)^T\right]$ is the mean squared error of the parameter estimation.

2) Parameter estimation: In the standard PAA, the parameter is estimated as

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k+1)\Phi^T(k)\tilde{x}(k+1|k) \quad (37)$$

where $F(k+1)$ is the learning gain in (15) with $\varphi(k)$ replaced by $\Phi^T(k)$. The parameter estimation error is

$$\tilde{\theta}(k+1) = \tilde{\theta}(k) - F(k+1)\Phi^T(k)\tilde{x}(k+1|k) + \Delta\theta(k) \quad (38)$$

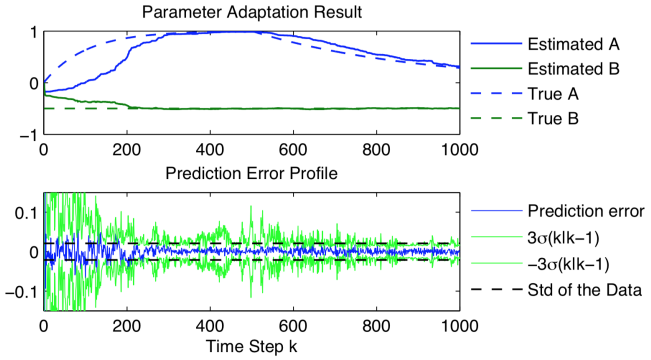


Fig. 3: Simulation of the MSEE Propagation Algorithm

[3]The objective function is linear while the constraint function defines an ellipsoid. The optimal solution must lie on the boundary of the ellipsoid. Let γ be a Lagrange multiplier. Define the new cost function as:

$$J_j^* = \frac{\partial \phi}{\partial x_j} x_j(k+1) + \gamma \left[9 - \Delta x_j^T X_j(k+1|k)^{-1} \Delta x_j\right] \quad (24)$$

The optimal solution satisfies $\frac{\partial J_j^*}{\partial x_j(k+1)} = \frac{\partial J_j^*}{\partial \gamma} = 0$, i.e.

$$\left(\frac{\partial \phi}{\partial x_j}\right)^T - 2\gamma X_j(k+1|k)^{-1} \Delta x_j = 0 \quad (25)$$

$$9 - \Delta x_j^T X_j(k+1|k)^{-1} \Delta x_j = 0 \quad (26)$$

Then (27) follows from (25-26).

where $\Delta\theta(k) = \theta(k+1) - \theta(k)$. Since the system is time varying, the estimated parameter is biased and the expectation of the error can be expressed as

$$\begin{aligned} & E\left(\tilde{\theta}(k+1)\right) \\ = & \left[I - F(k+1)\Phi^T(k)\Phi(k)\right] E\left(\tilde{\theta}(k)\right) + \Delta\theta(k) \\ = & \sum_{n=0}^{k}\prod_{i=n+1}^{k}\left[I - F(i+1)\Phi^T(i)\Phi(i)\right]\Delta\theta(n) \end{aligned} \quad (39)$$

The mean squared error of parameter estimation follows from (38) and (39):

$$\begin{aligned} & X_{\tilde{\theta}\tilde{\theta}}(k+1) \\ = & F(k+1)\Phi^T(k)X_{\tilde{x}\tilde{x}}(k+1|k)\Phi(k)F(k+1) \\ & -X_{\tilde{\theta}\tilde{\theta}}(k)\Phi^T(k)\Phi(k)F(k+1) \\ & -F(k+1)\Phi^T(k)\Phi(k)X_{\tilde{\theta}\tilde{\theta}}(k) \\ & +E\left[\tilde{\theta}(k+1)\right]\Delta\theta^T(k) + \Delta\theta(k)E\left[\tilde{\theta}(k+1)\right]^T \\ & -\Delta\theta(k)\Delta\theta(k)^T + X_{\tilde{\theta}\tilde{\theta}}(k) \end{aligned} \quad (40)$$

Since $\Delta\theta(k)$ is unknown in (39) and (40), it is set to an average time varying rate $d\theta$.

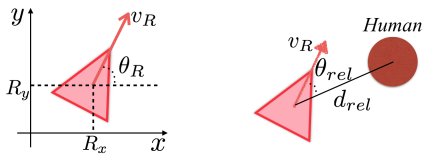
Fig.3 shows the simulation result of a first order system with a noise covariance $W = 0.005^2$. A forgetting factor $\lambda = 0.98$ is used. The solid and dashed blue lines in the upper figure are $\hat{A}(k)$ and $A(k)$, while the solid and dashed green lines are $\hat{B}(k)$ and the constant parameter B respectively. As observed in the figure, the time varying parameter $A(k)$ is well approximated by $\hat{A}(k)$, while $\hat{B}(k)$ converges to B. In the lower figure, the blue curve is the one step prediction error $x(k) - \hat{x}(k|k-1)$. The green curves are the 3σ bound ($\sigma = \sqrt{X_{\tilde{x}\tilde{x}}(k|k-1)}$). The black dashed line is the statistical standard deviation of the data $x(k) - \hat{x}(k|k-1)$ from $k = 1$ to $k = 1000$. As shown in the figure, the 3σ value offers a good bound for the prediction errors as all measured errors lie between the green curves. Moreover, the MSEE is larger when the parameter is changing faster, which captures the time varying property of the system. On the other hand, the statistical standard deviation of the data $x(k) - \hat{x}(k|k-1)$ does not give a good description of the data in real time.

B. Estimation of Human Behavior

In section III-B, a RLS-PAA algorithm is adopted in learning the closed loop behavior of the humans. In the safe exploration algorithm, the MSEE of $\hat{x}_j(k+1|k)$, $\forall j \in H$ also needs to be estimated. The system follows from (10) and (11), which is different from (30) in that the state is not exactly known. But it is assumed that the measurement noise is small, thus can be neglected. So the system is approximated by

$$y_H^R(k+1) = A_H(k)y_H^R(k) + B_H(k)u_H^c(k) + w_H^*(k) \quad (41)$$

This is equivalent to setting $\alpha = 1$ in (13). The prediction algorithm then follows from (34-40). In the implementation, the covariance of the noise W, the time varying rate $d\theta$ and the initial values are hand-tuned.



(a) The Vehicle Model (b) The Measurement

Fig. 4: Model of the Autonomous Vehicle

VI. A Comparative Study between SSA and SEA

In this section, a comparative study between SSA and SEA is performed on an autonomous vehicle model shown in Fig.4a. The vehicle's state is denoted by $x_R = [R_x, R_y, v_R, \theta_R]^T$ where R_x is the x-position of the vehicle, R_y the y-position, v_R the speed and θ_R the direction. The control input of the vehicle is $u_R = [\dot{v}_R, \dot{\theta}_R]^T$ (saturations apply: $|\dot{v}_R| \le a_{max}$ and $|\dot{\theta}_R| \le \omega_{max}$, where a_{max}, ω_{max} are positive constants). The state equation is

$$\dot{x}_R = f_{Rx}^*(x_R) + Bu_R \quad (42)$$

where $f_{Rx}^*(x_R) = \begin{bmatrix} v_R\cos\theta_R \\ v_R\sin\theta_R \\ 0 \\ 0 \end{bmatrix}$, $B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$.

The vehicle can measure its own state directly. It can also measure the relative distance d_{rel} and the relative direction θ_{rel} towards the nearby human as illustrated in Fig.4b. The human's state is defined as $x_H = [H_x, H_y, \dot{H}_x, \dot{H}_y]^T$ where H_x, H_y are the x, y positions and $\dot{H}_x, \dot{H}_y$ are the x, y velocities. The human's states are calculated based on the measurements and the robot state.

A. The Baseline Control

Suppose that the goal point of the robot is $[G_x, G_y]$. The baseline control law is designed as:

$$\dot{v}_R = -\left[(R_x - G_x)\cos\theta_R + (R_y - G_y)\sin\theta_R\right] - k_v v_R \quad (43)$$

$$\dot{\theta}_R = k_\theta\left[\arctan\frac{R_y - G_y}{R_x - G_x} - \theta_R\right] \quad (44)$$

where $k_v, k_\theta \in \mathbb{R}^+$ are constants[4].

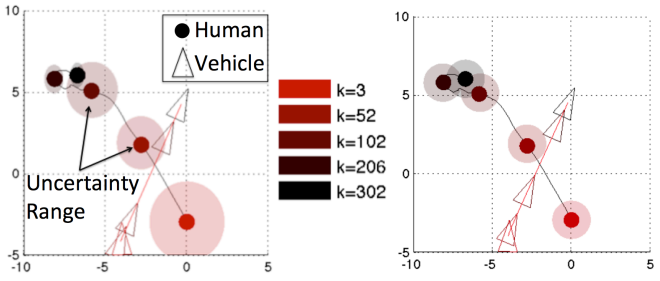
B. The Safe Control

The safety index is designed as $\phi = D^* - d^2 - k_\phi\dot{d}$ [6], [9] where $D^* \in \mathbb{R}^+$, $d = d_{rel}$ is the relative distance between the human and the vehicle and $k_\phi \in \mathbb{R}^+$ is a constant coefficient. S^{SSA} and S^{SEA} follows from (9) and (28), and

$$L(t) = \frac{\partial\phi}{\partial x_R}B = k_\phi\left[\cos\theta_{rel}, -v_R\sin\theta_{rel}\right] \quad (45)$$

In SSA, D^* is chosen as $d_{min}^2 + \eta_R T_s + \lambda_R^{SSA}T_s$; In SEA, D^* is chosen as $d_{min}^2 + \eta_R T_s + \lambda_R^{SEA}(k)T_s$. The final control follows from (17).

[4]This control law is designed according to the Lyapunov function $V = (R_x - G_x)^2 + (R_y - G_y)^2 + v_R^2$. The designed control law implies $\dot{V} \le 0$. Since $\{\dot{V} \equiv 0\}$ does not contain any trajectory of the system, the vehicle's state will converge to the goal point asymptotically.



(a) Simulation with SEA (b) Simulation with SSA

Fig. 5: Simulation of Human Vehicle Interaction under SEA and SSA

C. Simulation Results

Figure 5 shows the vehicle trajectory under SSA and SEA. The vehicle needed to approach (0,5) from (-5,-5) while the human went from (0,-3) to (-5,5). Five time steps are shown in the plots: $k = 3, 52, 102, 206, 302$ from the lightest to the darkest. The solid circles represent the human, which was controlled by a human subject through a multi-touch trackpad in real time (notice there was overshoot as the control was not perfect). The triangles represent the vehicle. The transparent circles in Fig.5a represent the set $\Gamma(k)$ in (22) mapped in 2D, which is shrinking gradually due to the reduction of uncertainties as an effect of learning. In Fig.5b, the transparent circles represent the equivalent uncertainty levels introduced by λ_R^{SSA}, thus the radius remain constant throughout the time.

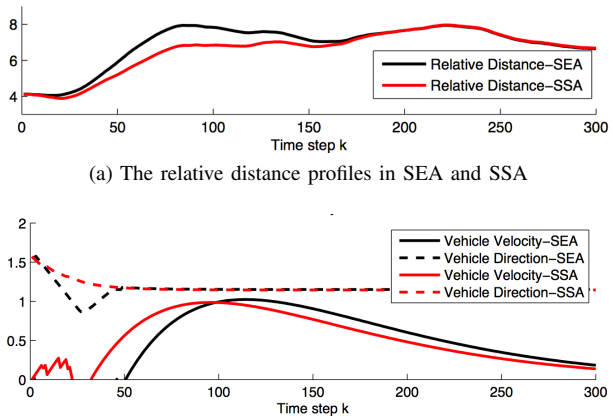
Figure 6 shows the distance profiles and the vehicle velocity profiles under SSA and SEA. Due to large initial uncertainties, the vehicle only started to accelerate after $k = 50$ (when the relative distance is large) in SEA. However, in SSA, the vehicle tried to accelerate in the very beginning, then decelerated when the relative distance to the human decreased. The velocity profile in SSA is serrated, while the one in SEA is much smoother. Meanwhile, in both algorithms, the relative distance is always greater than $d_{min} = 3$. However, before $k = 150$, the relative distance is kept larger in SEA than in SSA, since the vehicle was more conservative in SEA due to large uncertainty.

Fig.7 shows that the *a priori* MSEE provides a perfect bound for the prediction error.

In conclusion, the behavior in SSA is: move and modify; while in SEA, it is: move only if confident. The behavior under SEA is better for a new comer, while the behavior under SSA is better if the robot is already very familiar with the environment, i.e. with low uncertainty levels.

VII. Conclusion

Inspired by human's uncertainty reduction behavior during social interactions, the paper presented a control strategy that can handle different uncertainty levels during HRI. Based on the safe set algorithm (SSA), the safe exploration algorithm (SEA) was proposed. The constant uncertainty bound in SSA was replaced with a dynamic uncertainty bound in SEA. The learning and prediction method in the belief space was



(a) The relative distance profiles in SEA and SSA

(b) The velocity profiles of the vehicle in SEA and SSA

Fig. 6: Comparison between SEA and SSA

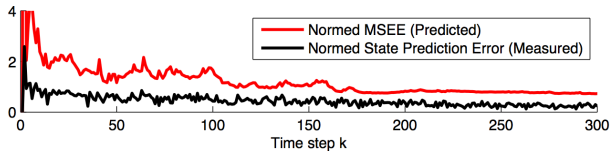


Fig. 7: Performance of the *a priori* MSEE as a Bound of the State Prediction Error

discussed. The simulation results confirmed the effectiveness of the algorithm. In particular, SEA is better than SSA when the uncertainty levels change from time to time, especially in the early stages of human robot interactions.

In the future, experiments with mobile robots will be conducted to further validate the algorithm.

References

[1] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," in *Humanoid Robots, 2008 IEEE-RAS International Conference on*. IEEE, 2008, pp. 91–98.

[2] J. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel, "Finding locally optimal, collision-free trajectories with sequential convex optimization," in *Robotics: Science and Systems (RSS)*, vol. 9, no. 1, 2013, pp. 1–10.

[3] N. E. Du Toit and J. W. Burdick, "Robot motion planning in dynamic, uncertain environments," *Robotics, IEEE Transactions on*, vol. 28, no. 1, pp. 101–115, 2012.

[4] L. Gracia, F. Garelli, and A. Sala, "Reactive sliding-mode algorithm for collision avoidance in robotic systems," *Control Systems Technology, IEEE Transactions on*, vol. 21, no. 6, pp. 2391–2399, 2013.

[5] C. R. Berger and R. J. Calabrese, "Some explorations in initial interaction and beyond: Toward a developmental theory of interpersonal communication," *Human communication research*, vol. 1, no. 2, pp. 99–112, 1975.

[6] C. Liu and M. Tomizuka, "Control in a safe set: addressing safety in human robot interactions," in *ASME 2014 Dynamic Systems and Control Conference*. ASME, 2014, p. V003T42A003.

[7] T. Taha, J. Miro, and G. Dissanayake, "A POMDP framework for modelling human interaction with assistive robots," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 544–549.

[8] T. Basar and G. J. Olsder, *Dynamic noncooperative game theory*. London: Academic press, 1995, vol. 200.

[9] C. Liu and M. Tomizuka, "Modeling and controller design of cooperative robots in workspace sharing human-robot assembly teams," in *Intelligent Robots and Systems (IROS), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 1386–1391.