¶
¶

Safe motion planning for human-robot interaction: design and experiments[1]

¶

Dana Kulić & Elizabeth Croft

¶

University of British Columbia
Canada

¶

1. Introduction

Robots have been successfully employed in industrial settings to improve productivity and perform dangerous or monotonous tasks. More recently, attention has turned to the use of robots to aid humans outside the industrial environment, in places such as the home or office. For example, as the population in the developed world ages, robots that can interact with humans in a safe and friendly manner while performing necessary home-care/daily living tasks would allow more seniors to maintain their independence. Such devices could alleviate some of the non-medical workload from health-care professionals, and reduce growing healthcare costs. To achieve such objectives, robotic devices must become more safe and user friendly. Untrained users need to feel comfortable and confident when interacting with a device that, unlike most passive household appliances, is potentially active in its interaction with the user.

Two key issues hampering the entry of robots into unstructured environments populated by humans are safety and dependability (Corke, 1999; Lee, Bien et al., 2001). To ensure the safety and intuitiveness of the interaction, the complete system must incorporate (i) safe mechanical design, (ii) human friendly interfaces such as natural language interaction and (iii) safe planning and control strategies. Our work focuses on this third item. The design of safe planning and control strategies can be divided into three key components: safe planning, human interaction monitoring, and safe control (Kulić and Croft, 2003). Discussion of the human monitoring and control system components can be found in Kulić and Croft (Kulić, 2005; Kulić and Croft, 2006a; Kulić and Croft, 2006b). Here, we specifically address the planning aspects related to safety in human-robot interaction (Kulić and Croft, 2005). First we discuss the development of a motion planning approach for robots that minimize a danger index based on the physical parameters of the robot and its proximity to the user. Then we present experiments where users are asked to evaluate the robot motions with regard to their

[1] This work is revised and expanded from work reported in Kulić and Croft (Kulić, 2005; Kulić and Croft, 2006a; Kulić and Croft, 2006b).

perception of the safety of the robot motion. Subjects reported significantly less anxiety and surprise, and reported feeling more calm when safe planned motions were presented, as compared to a conventional motion planner.¶

1.1. Related Work

In industrial applications, the safety of human–robot interaction is effected by isolating the robot from the human (Gaskill and Went, 1996; Corke, 1999; RIA/ANSI, 1999). In effect there is no interaction. As robots move from isolated industrial environments to interactive environments, this approach is no longer tenable (Corke, 1999). Three main approaches can be used to mitigate the risk during human-robot interaction: (i) redesign the system to eliminate the hazard, (ii) control the hazard through electronic or physical safeguards, and, (iii) warn the operator/user, either during operation or by training (RIA/ANSI, 1999). While the warn/train option has been used in industry, it had not been deemed effective in that setting (RIA/ANSI, 1999), and is even less suitable for robot interaction with untrained users. Examples of redesign include using a whole-body robot visco-elastic covering, and the use of spherical and compliant joints (Yamada, Hirawawa et al., 1997; Yamada, Yamamoto et al., 1999).

In unstructured environments, mechanical design alone is not adequate to ensure safe and human friendly interaction. Additional safety measures, utilizing system control and planning, are necessary. Several approaches have been proposed for ensuring safety through control. They focus on either slowing down or stopping when a hazardous situation is identified (Bearveldt, 1993; Yamada, Hirawawa et al., 1997; Zurada, Wright et al., 2001), moving to evade contact (Traver, del Pobil et al., 2000), or trying to minimize the impact force if contact occurs (Lew, Jou et al., 2000). A key problem for all of these control methods is to identify when safety is threatened. One approach is to use tactile sensors and force/torque sensors to identify a hazard when unplanned contact occurs (Yamada, Hirawawa et al., 1997). Recently, Ikuta et al. (Ikuta and Nokata, 2003) developed a danger evaluation method using the potential impact force as an evaluation measure. In their work, the danger index is defined as a product of factors which affect the potential impact force between the robot and the human, such as relative distance, relative velocity, robot inertia and robot stiffness.

Motion planning and the a priori identification of potentially hazardous situations as a means of reducing potential robot-safety hazards has received less attention than control-based (reactive) techniques. However, safe planning is important for any interaction that involves motion in a human environment, especially those that may contain additional obstacles. Application examples include service scenarios such as a dish clearing robot (Bearveldt, 1993), services for the disabled, such as approaching the human for a feeding task (Kawamura, Bagchi et al., 1995; Guglielmelli, Dario et al., 1996), and pick and place tasks for picking up and delivering common objects (Bischoff and Graefe, 2004). Including safety criteria at the planning stage can place the robot in a better position to respond to unanticipated safety events. Planning is thus used to improve the control outcome, similar to using smooth trajectory design to improve tracking (Erkorkmaz and Altintas, 2001; Macfarlane and Croft, 2003).

Several authors consider an a priori evaluation of the workspace to determine motion parameters within the various zones of the workspace (Bearveldt, 1993; Yamada, Hirawawa et al., 1997). Blanco et al. (Blanco, Balaguer et al., 2002) use distance measures from a laser scanner to generate a Voronoi diagram of the workspace of a mobile manipulator performing co-operative load carrying with a human. Since the Voronoi diagram maximizes distance from obstacles, paths generated along the Voronoi diagram present the safest course in terms of collision potential.

Khatib (Khatib, 1986) developed the potential field approach. In this method, the environment is described by an attractive (goal) potential field, which acts on the end effector, and a repulsive (obstacle) potential field, which acts on the entire robot body. The potential field is specified in the operational space. The potential field is used to generate forces to pull the robot away from any obstacles, and the end effector towards the goal. This approach does not require extensive pre-computation of the global path, can operate on-line, and can be easily adapted to sensor based planning and dynamic obstacles. When a redundant robot is used, this approach can be extended to allow the robot to continue executing the task while avoiding obstacles (Khatib, 1995). Maciejewski and Klein (Maciejewski and Klein, 1985) proposed a similar method for redundant manipulators and tasks where a goal trajectory is specified, and not just a goal location. In this approach, the force generated by the obstacle avoidance potential field is mapped to the null space of the redundant manipulator, so that the robot can continue to execute the goal trajectory while using its redundant degrees of freedom to avoid obstacles. A major issue with these planning methods is that only local search is used, so the robot can reach a local-minimum that is not at the goal location. A second issue is the formulation of the forces applied to the robot in the operational space. This requires the use of the robot Jacobian to translate these forces to joint torques, and introduces position and velocity error near any robot singularities.

Nokata et al. (Nokata, Ikuta et al., 2002) use a danger index based on the impact force between a human and the end effector. The danger index is the ratio of the actual force to the largest "safe" impact force (an impact force that does not cause injury to the human). The danger index is calculated based on factors such as the distance and velocity between the human and the manipulator end effector. Two approaches are proposed for planning the motion of a planar robot end effector: minimizing the greatest danger index along the path, and minimizing the total amount (integral) of the danger index along the path. However, their approach considers only the end effector motion.

Chen and Zalzala (Chen and Zalzala, 1997) use the distance between the robot and any obstacles as a measure of "safeness" in the cost function for path planning for mobile manipulators. A genetic programming approach is used to generate the optimum path given multiple optimization criteria, including actuator torque minimization, torque distribution between joints, obstacle avoidance and manipulability.

Oustaloup et al. (Oustaloup, Orsoni et al., 2003) describe a method for path planning using potential fields. The method is described for mobile robots, but is extendable to robot manipulators in configuration space. In pre-planning, obstacles are classified

according to how much danger they pose. The magnitude of the potential field gradient is varied by fractional integration, with a steeper slope for more dangerous obstacles. The fractional differentiation approach allows for a smooth transition between obstacles, however, this approach is susceptible to local minima.

Brock and Khatib (Brock and Khatib, 2002) describe the Elastic Strips framework for motion planning for highly articulated robots moving in a human environment. This method assumes a rough plan for accomplishing the task is available, and is fine-tuned on-line based on changes in the environment. The potential field method in operational space is used to plan the motion, with an attractive field pulling the robot towards the nominal off-line plan, and a repulsive force pushing the robot away from any obstacles. The existence of the pre-planned global path to the goal ensures that the robot does not get stuck in local minima. For redundant manipulators, an additional posture potential field is defined to specify a preferred posture for the robot. The posture field is projected into the null-space of the manipulator, so that it does not interfere with task execution. Although their paper does not deal explicitly with ensuring safety, the posture potential can be used to formulate safety-based constraints.

Most path planning algorithms for human environments focus on maximizing the distance between the robot and any obstacles in the environment. In this work it is proposed that the robot posture can also be optimized during path planning to significantly improve the safety of the manipulator.

1.2. System Overview

The system architecture assumes a user-directed robot system. The user must initiate each interaction, but the robot has sufficient autonomy to perform commanded actions without detailed instructions from the user. An overview of the idealized system is presented in Figure 1. The user issues a command to the robot to initiate the interaction. The command interpreter translates the natural language command (e.g.: pick up the red cup) into a set of target locations and actions (e.g., execute a grip maneuver at coordinates [x,y,z]). The planning module is divided into two parts: the global path planner and the local trajectory planner. The global planner module begins planning a geometric path for the robot over large segments of the task, utilizing the safety strategy described herein. Segment end points are defined by locations where the robot must stop and execute a grip or release maneuver. For example, one path segment is defined from the initial position of the robot to the object to be picked up. The local planner generates the trajectory along the globally planned path based on real-time information obtained during task execution. The local planner generates the required control signal at each control point. Because the local planner utilizes real-time information, it generates the trajectory in short segments.

During the interaction, the user is monitored to assess the user's level of approval of robot actions. The local planner uses this information to modify the velocity of the robot along the planned path. The safety control module evaluates the safety of the plan generated by the trajectory planner at each control step. The safety control module initiates a deviation from the planned path if a change in the environment is detected

that threatens the safety of the interaction. This deviation will move the robot to a safer location. Meanwhile, the recovery evaluator will initiate a re-assessment of the plan and initiate re-planning if necessary.

The focus of this work is the development of a motion planning strategy that explicitly defines a measure of danger in the interaction, and incorporates this measure as a criterion during planning. The developed planning strategy is a standalone component that can be used as part of the architecture described above, or as part of a different control and planning architecture. In this work, we develop the danger evaluation measure and its use during path planning. Following this description, we present simulations and experiments to demonstrate the behavior and effectiveness of the planner, and characterize the user evaluation of the proposed strategy.

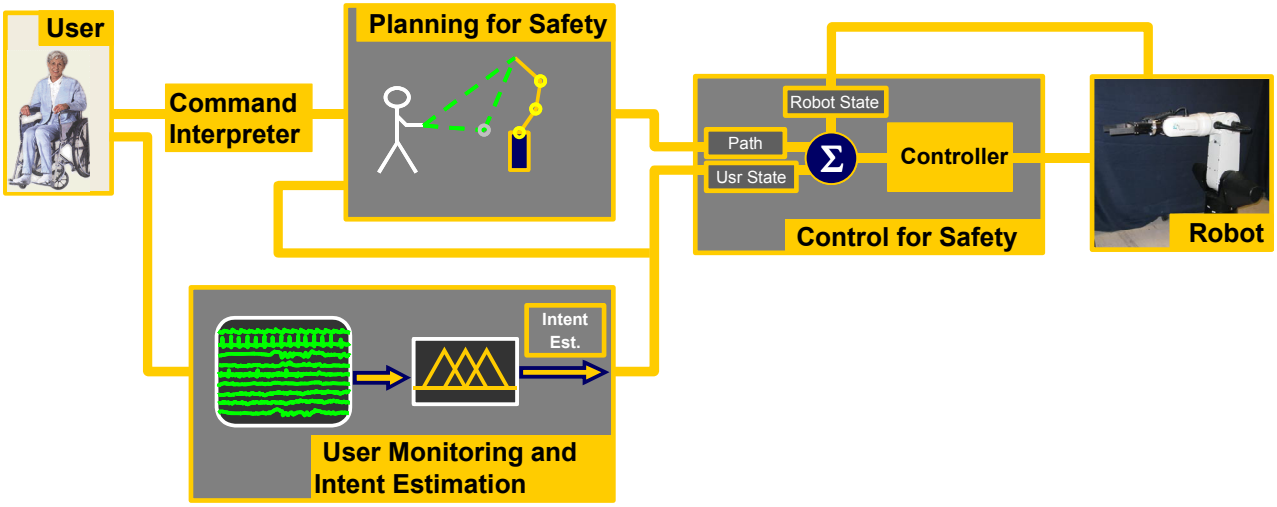


Figure 1 - System Overview.

¶

2. Measuring the safety of a planned interaction

A hazard requiring a change in robot behavior can be defined by a minimum distance between the robot and the person (Bearveldt, 1993; Traver, del Pobil et al., 2000), or by using a threshold level of the danger index based on impact force (Nokata, Ikuta et al., 2002; Ikuta and Nokata, 2003). Here, an index similar to (Nokata, Ikuta et al., 2002; Ikuta and Nokata, 2003) is proposed, and applied to configuration space planning of the robot motion. By selecting safer configurations at the planning stage, potential hazards can be avoided, and the computational load for hazard response during real time control can be reduced, as shown in Figure 2. In both panels the robot has the same end-effector location, but in the panel of the right (b), the hazard to the user is minimized by the posture adopted by the robot.

Safe planning is an important component of the safety strategy. For example, if the path to be followed is planned with a general path planning method, the robot may spend the majority of the path in high inertia configurations. If the user suddenly moves closer to the robot, the potential collision impact force will be much higher than if the robot

had been in a low inertia configuration, regardless of the real-time controller used to deal with potential collision events.

When selecting a path planning strategy, there is a tradeoff between fast local methods that may fail to find the goal, and slow global methods (Latombe, 1991). To exploit advantages of both methods, recent path planning algorithms have used a hybrid approach, where global path planning is used to find a coarse region through which the robot should pass, and local methods are used to find the exact path through the region (Brock and Khatib, 2002).

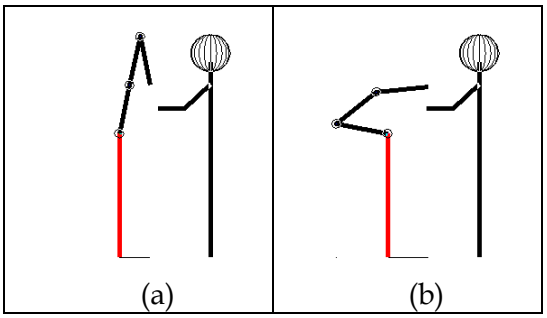


Figure 2. Planning a safe interaction. Posture (b) has minimized potential hazard to the user.

Similarly, in this approach, the global planner generates a safe contiguous region in space through which the robot can move to complete the given task. This region in space is described by a set of contiguous configurations, which represent the path. It is then left to the on-line trajectory planner to generate the exact path in the region, and the trajectory along that path. This trajectory is evaluated and, if necessary, corrected at every control step by the safety module to handle the real-time aspects of the interaction.

Since the task planning is done following a user request, the global planner must execute within several seconds at most, to avoid a significant delay between a user request and robot response. To ease the computational load on the global planner, the task is separated into segments. Natural segment separation points occur when the robot is required to pause at a particular location, for example at each grasp or release point. Only the first segment must planned before the planned path can be passed on to the local planner and the robot can begin executing the task. In this way, global planning of the next segment can continue in parallel with execution of the current segment.

2.1. Danger Criterion

The planning module uses the best-first planning approach (Latombe, 1991). In this method, the robot configuration space is discretized into 0.1 rad cells, and a path is found by iteratively building a tree of configurations, with the first configuration at the root. At each iteration step, the neighbours of the leaf configuration with the lowest cost value are added to the tree. The algorithm therefore follows the steepest descent of the cost function, and escapes from local minima by well-filling. The search stops when the target configuration is reached or the entire space has been mapped. The algorithm is

resolution complete. In cases when the number of degrees of freedom (DoF) of the robot affecting gross end-effector motion are small (less than 5), the best-first planning approach provides a fast and reliable solution (Latombe, 1991). For highly redundant robots, a different search strategy can be employed, such as randomized planning (Barraquand and Latombe, 1991), or probabilistic roadmap planning (Kavraki, Svestka et al., 1996; Ahuactzin and Gupta, 1999; Choset and Burdick, 2000; Yu and Gupta, 2000). For example, either random sampling of the configuration space (Kavraki, Svestka et al., 1996), or the Generalized Voronoi Graph (Choset and Burdick, 2000), can be used to generate a roadmap of the connected free regions in the configuration space. The roadmap represents a subspace of the entire configuration space. The search based on lowering the danger criterion can then be applied to the roadmap, rather than the entire configuration space, reducing the search time for high DoF manipulators. However, the search criteria presented herein remain identical regardless of the search strategy used. The safest path is found by searching for contiguous regions that: (i) remain free of obstacles, (ii) lead to the goal, and, (iii) minimize a measure of danger (a danger criterion). The planning algorithm seeks a path that minimizes a cost function consisting of a quadratic goal seeking function, a quadratic obstacle avoidance function, and the danger criterion (DC).

The danger criterion is the central contribution of the planner cost function. Since path planning (as opposed to trajectory planning) does not consider robot velocities, a configuration-based (quasi-static) danger criterion is required. To be effective, the danger criterion should be constructed from measures that contribute to reducing the impact force in the case of unexpected human-robot impact, as well as reducing of the likelihood of impact. These can include the relative distance between the robot and the user, the robot stiffness, the robot inertia, the end-effector movement between contiguous configurations, or some combination of these measures, similar to those proposed in (Ikuta and Nokata, 2003). In (Nokata, Ikuta et al., 2002), Nokata et al. use the danger index to find an optimum safe path, however, only the end-effector trajectory with respect to the user is considered. As an expansion of this approach, a safe path for the entire robot structure is planned, explicitly planning the robot posture. However, since some of the factors affecting danger can conflict (e.g., a low stiffness configuration can also be high inertia configuration) it is important to re-formulate the danger criterion so that conflicting factors do not act to cancel each other out. Herein, the robot inertia and the relative distance between the robot and the user's center of mass are used. The robot stiffness was not included as it can be more effectively lowered through mechanical design (Bicchi, Rizzini et al., 2001; Ikuta and Nokata, 2003). Instead, the proposed approach modifies the robot inertia, lowering the effective impedance of the robot. Dynamic factors, such as the relative velocity and acceleration between the robot and the user, are handled by the trajectory planner and the safety module (Kulić and Croft, 2006b).

For optimization purposes, a scalar value representing the effective robot inertia at each configuration must be computed. For a general robot architecture, where the robot's inertia may be distributed in more than one plane, the largest eigenvalue of the 3x3

inertia tensor may be used as the scalar measure. For robots with a single sagittal[2] plane (e.g. anthropomorphic, SCARA), the scalar inertial value is extracted by calculating the robot inertia about an axis originating at the robot base and normal to the robot's sagittal plane.

$$I_S = \vec{v}^T I_b \vec{v} \,. \tag{1}$$

Here, I_s is the inertia about the v axis, v is the unit vector normal to the robot sagittal plane and I_b is the 3x3 robot inertia tensor about the base.

For each danger criterion factor, a potential field function is formulated as a quadratic function. Quadratic potential functions are most commonly used in general potential field planners. They have good stabilization characteristics, since the gradient converges linearly towards zero as the robot's configuration approaches the minimum (Khatib, 1986; Latombe, 1991).

2.1.1 Sum-Based Criterion

Two danger criterion formulations are proposed: a sum-based and a product-based criterion. For the sum-based danger criterion, the inertia factor is:

$$f_{I_{sum}}(I_S) = \frac{I_S}{m} \,, \tag{2}$$

where, m is the total mass of the robot. This function can be interpreted as a quadratic attractive function, attracting each link towards the robot base.

The relative distance factor for the sum-based danger criterion is implemented by a repulsive function between the user and the robot center of mass (CoM). The center of mass distance is used (instead of the closest point distance) to allow the robot end-effector to contact the user during interaction tasks, while maximizing the distance between the user and the bulk of the robot. The potential field is described by equation (3) below:

$$D_{CMO} = \left| D_{CM} - D_{\min} \right|$$

$$f_{CM_{sum}}(D_{CM}) = \begin{cases} \frac{1}{2\varepsilon} : & D_{CMO} \le \varepsilon \\ \frac{1}{2}\left(\frac{1}{D_{CMO}} - \frac{1}{D_{\max}} \right)^2 : & \varepsilon < D_{CMO} < D_{\max} \\ 0 : & D_{CMO} \ge D_{\max} \end{cases} \,. \tag{3}$$

In Equation (3), D_{CM} is the current distance between the robot center of mass and the user, D_{min} is the minimum allowable distance between the robot centre of mass and the user, and D_{CMO} is computed as the difference between these two distance measures (namely, the separation distance above the minimum limit). D_{max} is the distance at which D_{CMO}, the current distance above the minimum limit, no longer contributes to the cost function (for example, if no human is visible in the environment). ε is a small

[2] The sagittal plane is the vertical plane (plane of symmetry) passing through the center of the outstretched robot arm.

number used to limit the function for D_{CM} near D_{min}. This potential field is analogous to an obstacle potential field acting between the centers of mass of the user and the robot.
The sum based danger criterion is comprised of the inertia factor (Equation (2)) and the centre of mass distance factor described above (Equation (3)), as follows:

$$DC_{sum} = W_i \cdot f_{I_{sum}}(I_S) + W_d \cdot f_{CM_{sum}}(D_{CM}) \tag{4}$$

Here, W_i and W_d are weights of the inertia and distance term, scaled such that $W_i + W_d = 1$. The weights W_i and W_d are tuned based on the particular robot structure. For low inertia robots, and when the robot is close to the user, the distance factor will dominate the danger criterion, because the distance factor approaches infinity as the robot approaches the person. If inertia reducing behavior is desired for the path in these cases, W_i should be greater than W_d.

2.1.2 Product Based Criterion

For the product-based danger criterion, the criteria are scaled such that for each potential function, the level of danger is indicated within the range [0 – 1]. Values greater than one indicate an unsafe configuration.
The product based inertia criterion is defined as:

$$f_{I_{prod}}(I_S) = \frac{I_S}{I_{\max}}, \tag{5}$$

where, $I_{\max}$ is the maximum safe value of the robot inertia. In the simulations described in Section 5, the maximum robot inertia is used; however, a lower value can be used for high-inertia manipulators. In this case, the maximum safe value can be established based on the largest force magnitude that does not cause pain (Yamada, Suita et al., 1996) and the maximum robot acceleration.
For the product based distance criterion, similar to the sum based distance criterion, the center of mass distance between the robot and the user is used. The relative distance criterion for the product-based danger criterion is:

$$f_{CM_{prod}}(D_{CM}) = \begin{cases} k\left(\frac{1}{D_{CM}} - \frac{1}{D_{\max}}\right)^2 : & D_{CM} \le D_{\max} \\ 0 : & D_{CM} > D_{\max} \end{cases}. \tag{6}$$

The scaling constant k is used to scale the potential function such that the value of the potential function is zero when the distance between the user and the robot is large enough (larger than D_{max}), and is one when the distance between the user and the robot is the minimum allowable distance (D_{min}):

$$k = \left(\frac{D_{\min} \cdot D_{\max}}{D_{\min} - D_{\max}}\right)^2. \tag{7}$$

Values of the product-based distance criterion above one indicate an unsafe distance.
The product-based danger criterion is then computed as a product of these contributing factors.

$$DC_{prod} = f_{I_{prod}}(I_S) \cdot f_{CM_{prod}}(D_{CM}). \quad (8)$$

2.1.3 Goal and Obstacle Potential Fields

For the goal seeking and obstacle avoidance functions, the customary quadratic potential field functions are used (Khatib, 1986; Latombe, 1991; Brock and Khatib, 2002). The goal seeking function f_G is defined as:

$$f_G(D_G) = \frac{1}{2} D_G^2, \quad (9)$$

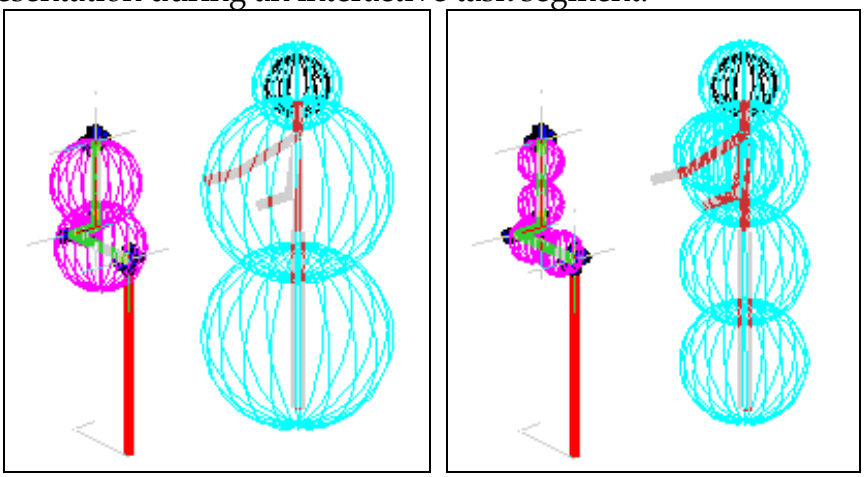
where, D_G is the distance between the end-effector and the goal.
The obstacle avoidance function f_O is defined as:

$$f_O(D_O) = \begin{cases} \frac{1}{2}\left(\frac{1}{D_O} - \frac{1}{D_{O\min}}\right)^2 : & D_O \le D_{O\min}, \\ 0 : & D_O > D_{O\min} \end{cases} \quad (10)$$

where, D_O is the distance between the robot and the nearest obstacle, D_{Omin} is the distance from the obstacle at which the obstacle begins to repel the robot (the influence distance). For the obstacle avoidance function, the distance between the robot and the nearest obstacle is taken as the distance between the closest point on the robot and the closest point on the obstacle. The distance between the robot and the nearest obstacle, as well as the distance between the robot and the non-interacting parts of the user are estimated using the hierarchy of spheres representation (Martinez-Salvador, del Pobil et al., 2000), illustrated in Figure 3. In this approach, the robot and the obstacles in the environment are described as a set of enveloping spheres[3]. Initially, a small set of large enveloping spheres is used for each object. If no intersecting spheres are found, the distance between the two closest sphere centers is returned as the distance between the robot and the nearest obstacle or human. If two intersecting spheres are found, the robot and the obstacles are decomposed into a set of smaller enveloping spheres. The process is repeated until a non-intersecting set of spheres is found, or until the maximum level of decomposition is reached, in which case the algorithm reports that a collision has been detected. The level of decomposition required to find a collision free set of spheres is also used to determine the size of the region within which local trajectory planning may be executed, as in (Brock and Khatib, 2002).
When defining the enveloping spheres for the user, the current robot task also becomes important. If the goal of the interaction is for the robot to approach and/or contact the user, then it is not appropriate to represent the user simply as an obstacle, as in (Traver, del Pobil et al., 2000). Instead, in this work, during pre-planning, each segment of the path is classified as interactive or non-interactive. If the segment is classified as non-interactive, the entire region of space occupied by the user is treated as an obstacle. If the segment is classified as interactive, a smaller set of spheres is used, such that the target area of the user (for example, the hand) is excluded from the obstacle area. By

[3] Other representations could also be used (for example, blobs) (Wren, Azarbayejani et al., 1997), however, this approach provides fast computation and the necessary accuracy level required

only excluding the contact area, this approach ensures that the robot can reach the intended target, while motion is still restricted to non-target areas of the body. Using this representation also ensures that the robot will slow down as it approaches the target, due to the effect of nearby spheres. Figure 3 (a) shows the robot and the user represented with enveloping spheres in a non-interactive task segment. Figure 3 (b) shows the representation during an interactive task segment.



(a) (b)

Figure 3. (a) Human, robot representation in a non-interactive task. (b) Human, robot representation in an interactive task.

2.2. The Overall Cost Function

The planning-cost function is generated by combining the goal seeking, obstacle avoidance, and danger criteria. The planned path is generated by searching for a set of configurations that minimize the cost function:

$$J = W_G \cdot f_G(D_G) + W_O \cdot f_O(D_O) + W_D \cdot K \cdot DC. \quad (11)$$

Here, W_G is the weighting of the goal seeking criterion, W_O is the weighting of the obstacle avoidance criterion, W_D is the weighting of the danger criterion, and K is a scaling factor. The selection of the weight levels is discussed in the following section.

3. Implementation

Using the above cost function, it is likely that the danger criterion will conflict with the goal seeking criteria during the search, leading to local minima and long search times. To avoid this problem, a two-stage search is proposed. In the first stage, maximum priority is placed on minimizing the danger criterion. A threshold is established for determining when an acceptable maximum level of danger is achieved. Once a path is found that places the robot below this threshold, the second stage of the search is initiated. In this stage, maximum priority is placed on the goal-seeking criterion. In the resulting overall path, the robot will spend most of its time in low danger regions. One can note that, this approach will not result in the shortest distance path. The tradeoff between increased safety and reduced distance can be controlled by modifying the

threshold where switching from the first stage to the second stage occurs. The two stages are implemented by modifying the weighting factors. In the first (danger minimization) stage, the danger weighting, W_D is greater than the goal seeking weighting, W_G, while in the second stage, W_G is greater than W_D. As long as the relative weights are set in this manner, the algorithm does not require tuning of the weight levels when using the product based danger criterion. For the sum based danger criterion, if the robot is approaching the person, W_D must be small (0.1 or less) in the second stage to avoid interference with the goal attraction criterion.

Even when the proposed two stage planning approach is used to minimize the conflict between the danger and goal seeking criteria, it is still possible for the goal seeking and the obstacle avoidance to conflict in a cluttered environment, or when joint limits are encountered during the search. The search time is also extended if the robot needs to reverse configurations during the path (for example, from an elbow down starting configuration to an elbow up final configuration). In these cases, a circuitous path is often generated, requiring some post-process smoothing (Latombe, 1991). In particular, if there are several obstacles positioned close to the robot, it may not be possible to complete the stage 1 search within the given threshold. In this case, the user should be notified that a safe path cannot be found in the current environment.

The problem of long search times can also be addressed by taking advantage of particular robot geometry, and searching only through joints that affect the end-effector position. For example, although the PUMA560 is a 6 DoF robot, only the first 3 joints contribute to the gross end-effector movement. After the position path is generated, the remaining 3 joints can be used to maintain a desired end-effector orientation, as required by the payload.

4. Search Strategy Improvements (Backwards Search)

The global planning strategy presented above is generally valid for non-redundant as well as redundant robots, as well as robots with either prismatic or articulated joints, or mobile robots. This is because the search is conducted forwards from an initial configuration, the search steps are generated from that initial configuration, and therefore only forward kinematics are required to calculate the workspace potential field functions.

If an inverse kinematics routine is available for the robot, the algorithm search time can be improved by adding a backwards search stage. This addition is useful in those cases when the robot goal is in a crowded area, for example when the robot's goal is the user's hand. In this case, to get to the goal, the robot must go into an area of higher potential field, since the goal is surrounded by obstacles generating a repulsive field. Therefore, the algorithm must perform "well-filling" to find the path, increasing the search time, and, potentially, resulting in a convoluted final path. On the other hand, if the search can be performed backwards, gradient descent can be used to find the lowest potential path to the goal, reducing the search time.

In general, it is always more efficient to search from the cluttered end of the path (Kondo, 1991; Hwang and Ahuja, 1992). The inverse kinematics routine is used to

generate the goal configuration, given the goal workspace position and the desired end-effector orientation at the goal. The search is then initiated backwards from the goal configuration towards the start configuration. Once obstacle influence is minimal, the backwards search stops, and the forward search (as described in Section 3) is initiated, with the last configuration of the backwards generated path as the goal. This location at which the backwards search stops, and the new goal location for the forwards search is named the intermediate location (IL). If there are multiple solutions to the inverse kinematics problem, the danger criterion at each solution is evaluated, and the solution with the lowest danger criterion is selected.

For continuity, the algorithm must also ensure that the forwards and backwards-generated paths meet at the same point in configuration space. Since the goal and obstacle potential fields are defined in the workspace, it is possible for an articulated robot to reach the starting point of the backwards path in an incorrect posture (e.g., elbow up vs. elbow down). In this case, the two paths cannot be joined by simply generating a spline between the two postures. This could cause the robot to move into obstacles or move through a more dangerous configuration. Instead, during the initial stage of the forward search, an additional "posture" potential function (Brock and Khatib, 2002) is added, that favors the starting posture of the backwards path. The posture function is defined as:

$$f_{pos}(q) = \begin{cases} \frac{1}{2}(q - quad_{\min})^2 : & q < quad_{\min} \\ 0 : & quad_{\min} \le q \le quad_{\max} \\ \frac{1}{2}(quad_{\max} - q)^2 : & q > quad_{\max} \end{cases}. \qquad (12)$$

Here, q is the search configuration joint angle, and $quad_{\min}$ and $quad_{\max}$ are the quadrant boundaries centered around the each joint angle found in the final configuration of the backwards search (i.e., the joint angles associated with the IL).

The posture function is calculated for each joint; the total posture function is the sum over all the joints. The posture potential is only active while the robot is in the incorrect posture. When the robot reaches the correct posture, the posture potential becomes inactive. To ensure that the correct posture is reached prior to merging with the backwards planned path, an additional condition is added to the switchover from the first to the second stage of the planning. Namely, in addition to the low danger index requirement, the posture potential must also be zero. A flow chart for the complete algorithm, including the backwards search, is shown in Figure 4.

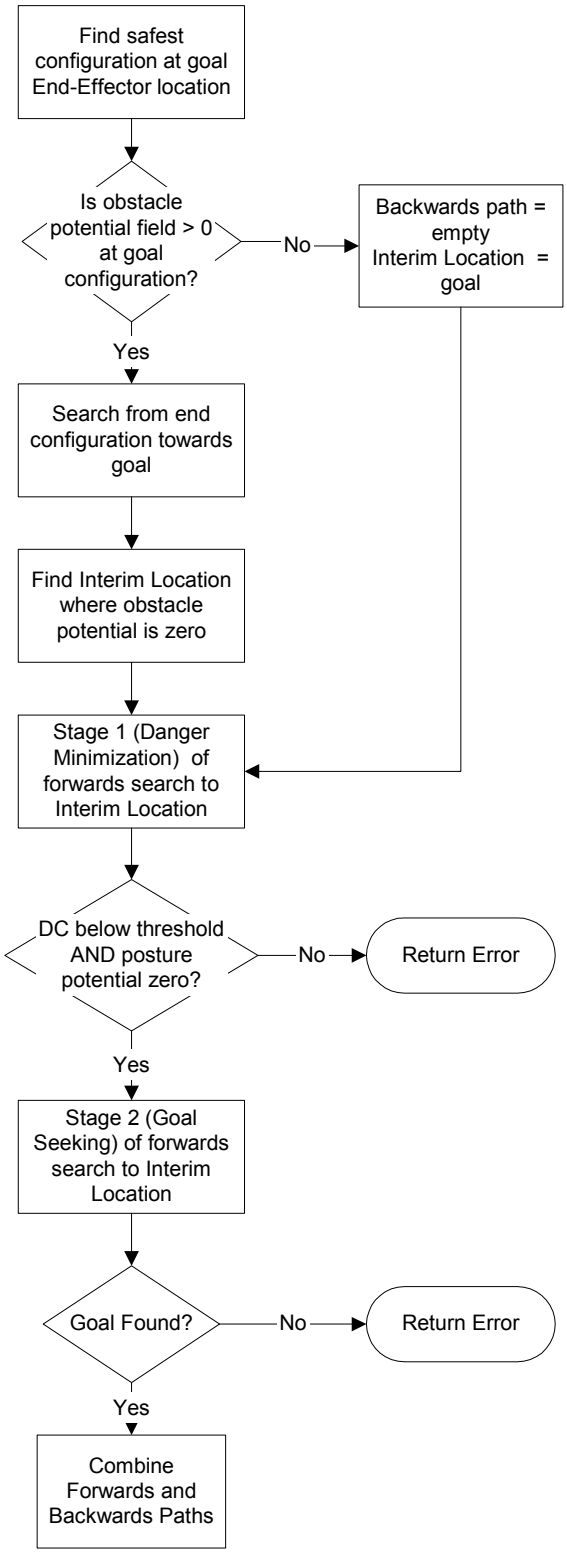


Figure 4. Combined backwards-forwards search algorithm flowchart.

5. Simulations

A simulation environment was developed to test the planning algorithms with various robot architectures. The robots are modeled using the Robotics Toolbox (Corke, 1996). Figure 5 shows the planned motion of a 3 link planar robot using the basic algorithm (i.e., without the backwards search), with the sum-based danger criterion. The robot's goal is to pick up the object being held by the user. The same task is shown as planned by the product-based danger criterion in Figure 6. In both cases, to better illustrate the effect of the danger criterion, only the goal and danger criterion cost functions are included. The cost function weights used for both plans are given in Table 1. Figure 7 shows a comparison between the user-robot center of mass distance and the robot inertia for the sum-based and the product-based danger criteria.

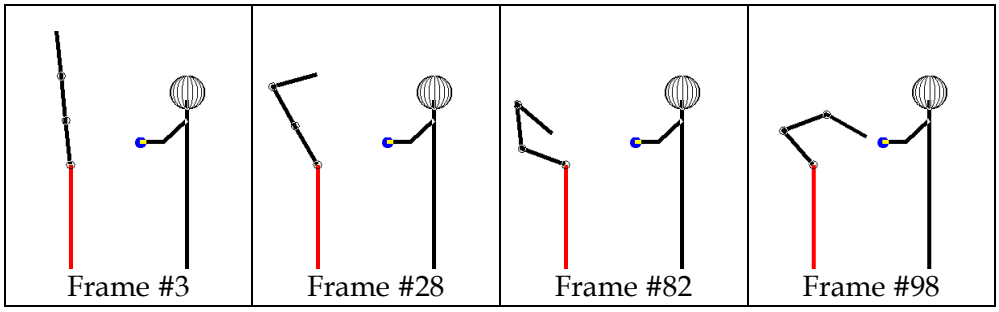


Figure 5. Planned path with sum-based danger criterion.

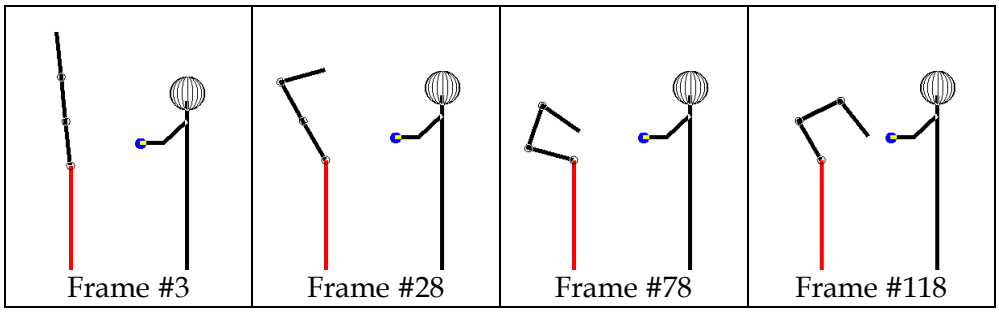


Figure 6. Planned path with product-based danger criterion.

Figure 5 and Figure 6 illustrate the differences between the two danger criterion formulations. The sum-based danger criterion implies that the factors affecting the danger can be considered independently of one another when assessing the level of danger. One advantage of the sum-based criterion is that the formulation is similar to other quadratic cost functions normally used in the potential field approach, and is distance based. Therefore, the sum-based criterion does not need to be scaled when combined with the other criteria (i.e., $K = 1$). The center of mass distance factor is a repulsive potential field, and can, therefore, become infinite in magnitude when the center of mass distance between the robot and the user (D_{CM}) is close to the minimum safe distance (D_{min}). Thus, when the robot and the user are close together, the distance factor will dominate over the inertia factor. This effect is illustrated in the last frame of the Figure 5 sequence.

Table 1. Planar robot simulations weights.

	W_G	W_O	W_D
Stage 1	0.2	0	0.8
Stage 2	0.9	0	0.1

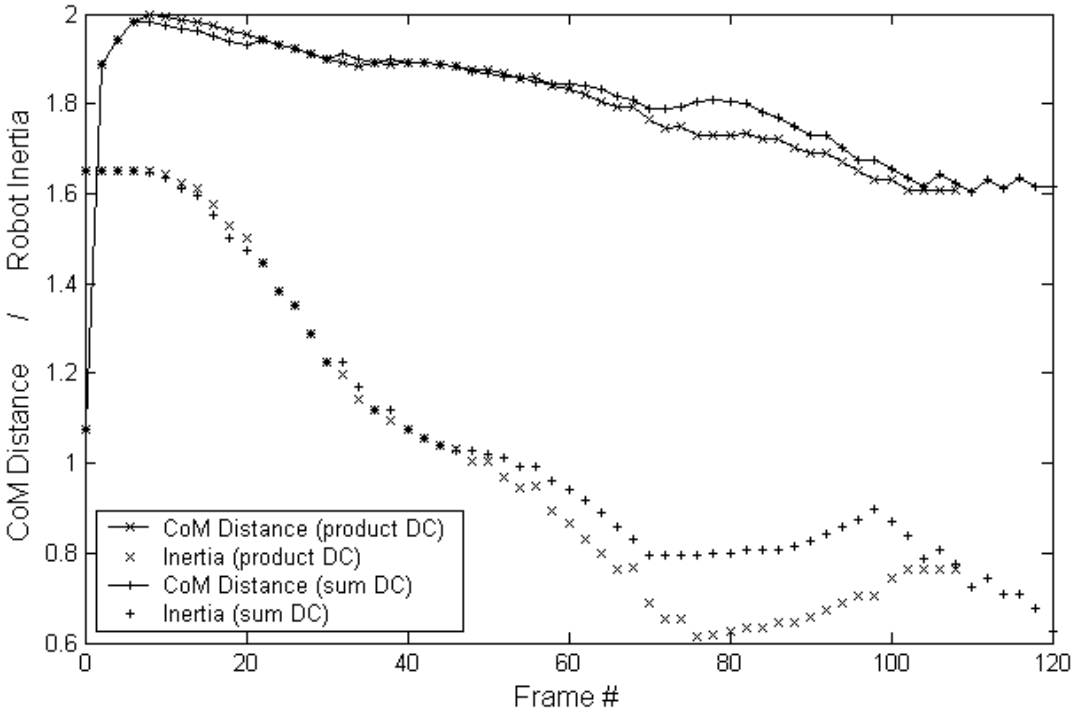


Figure 7. Comparison between the sum-based and product-based danger criteria.

As a result, near the point of interaction, the sum based criterion results in a higher inertia, as can be seen in Figure 7. In general, the disadvantage of such a sum based formulation is that one of the factors always tends to dominate the others. Furthermore, for the sum-based criterion it is difficult to define the threshold at which one should switch from the danger minimization stage to the goal seeking stage, since the danger criterion is a combination of the robot link distances from the robot base and the distance from the robot to the person.

The product-based danger criterion implies that the factors affecting the danger criterion need to be considered collectively when assessing the level of danger. For example, if the distance between the robot and the person is large, the other contributing factors will not be minimized either. In Figure 6, since the distance between the robot and the person is small, both the distance factor and the inertia factor are minimized. In addition, when both factors have significant magnitude, the danger criterion gradient will be steepest, ensuring that the danger criterion is prioritized over the other criteria. Because the two factors scale each other, both are minimized to achieve the required safety level. Another advantage of the product-based criterion is that the criterion represents a clear indication of the level of danger, ranging from 0 to 1 (values greater than 1 are possible when the distance between the robot and the user (D_{CM}) is smaller

than the minimum safe distance (D_{CMmin}). Therefore, it is easier to specify the switch threshold as the desired level of danger. However, for the product-based criterion, a scaling factor (K) must be chosen so that the danger criterion is on the same scale as the goal and obstacle criteria.

In the majority of cases, the product-based danger criterion is more suitable. The product-based criterion is more suitable for redundant robots, where both the inertia and Centre of Mass (CoM) distance factors can be minimized, regardless of the CoM distance. When the robot is close to the person, the product-based danger criterion will decrease inertia and increase CoM distance. On the other hand, close to the person, the sum-based danger criterion becomes dominated by the distance factor, so inertia is not reduced as significantly. The sum-based danger criterion may be more suitable with large, under-articulated robots. In this case, the difference between the maximum and minimum robot inertia may not be very significant, whereas the strong CoM distance action will ensure that the robot does not get too close to the user.

Figure 8 shows a planned motion sequence with a PUMA 560 robot, using the basic algorithm with the goal, obstacle and danger criteria. The product based danger criterion is used. Table 2 gives the weights used in the search. For comparison, a path was generated using the best-first planner without any danger criterion. As illustrated in Figure 9, the danger criterion pushes the CoM of the robot away from the person along the majority of the path, as well as significantly reducing the robot's inertia.

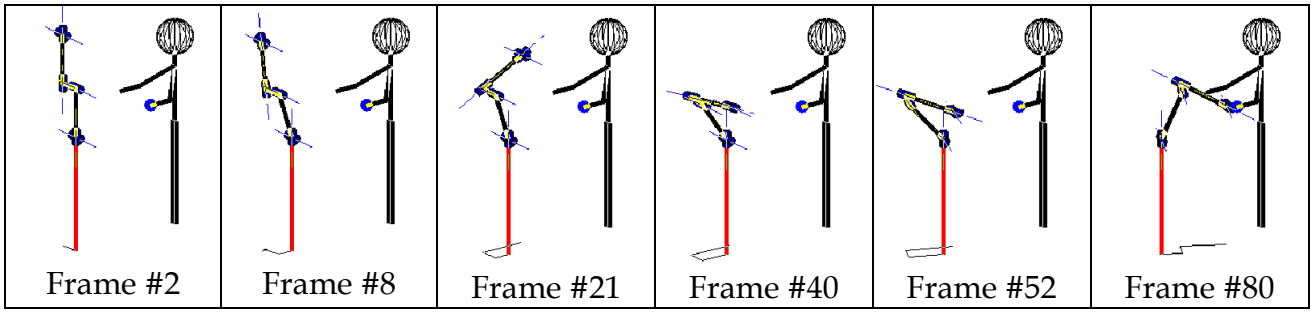


Figure 8. Planned sequence for a PUMA560 robot (product danger criterion).

Table 2. PUMA560 simulations weights.

	W_G	W_O	W_D
Stage 1	0.1	0.2	0.7
Stage 2	0.7	0.2	0.1

Figure 10 shows a planned sequence using the modified algorithm, with the backwards search added. In this case, the initial robot pose is the reverse of the required final pose generated by the backwards plan. The same weights were used as for the basic algorithm, as specified in Table 2.

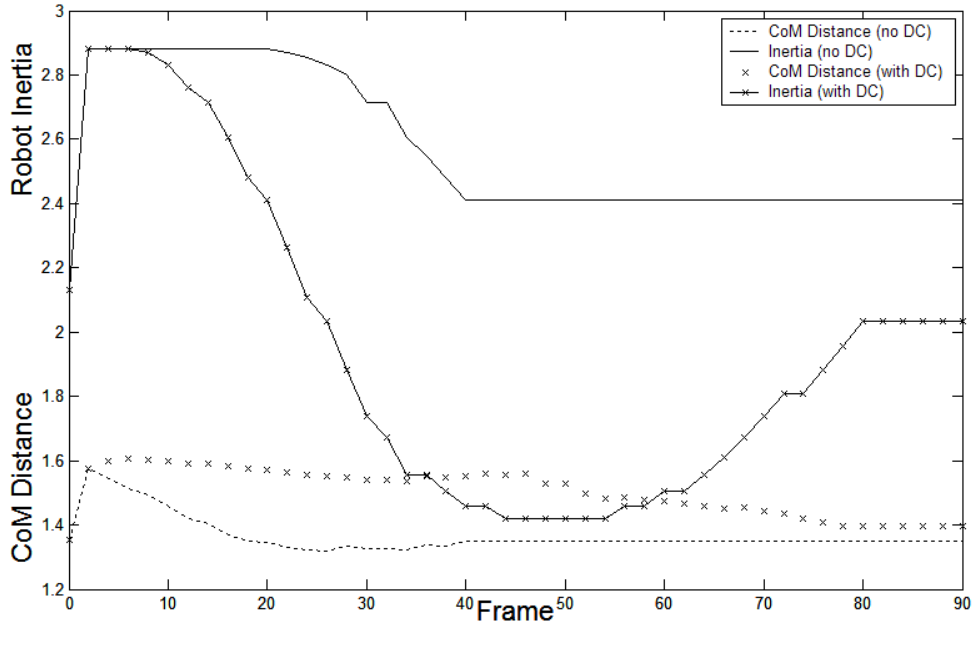


Figure 9. Effect of the danger criterion search on the danger factors (product danger criterion).

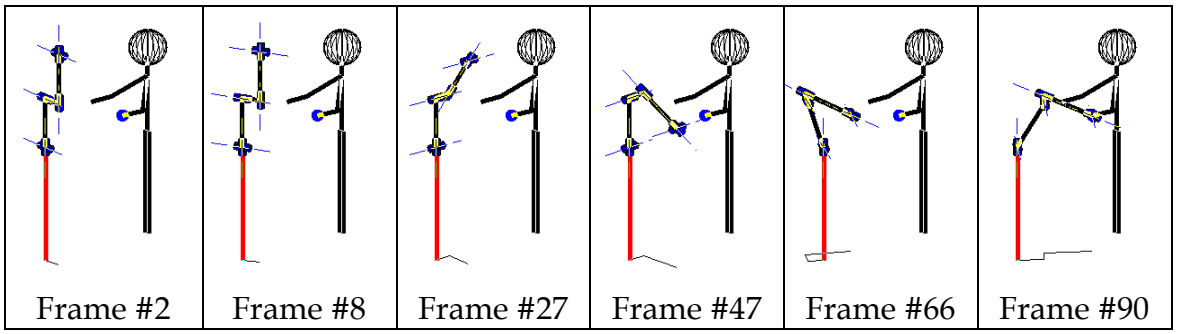


Figure 10. Planned sequence for a PUMA560 robot with backwards search (product danger criterion).

Initially, while the danger is low, the posture function dominates the potential field, and the robot moves first to move to the correct posture. As the robot comes closer to the person, the danger criterion begins to dominate the potential field, and the robot inertia is reduced. Once the danger index has been reduced, the robot moves towards the goal. Posture correction is performed during low danger sections of the path. As discussed in Section 4 and shown in the flowchart in Figure 4, the backwards search is only performed when the goal location is within the influence distance of obstacles. In this case, the basic planner must find the entrance into an obstacle region. Using the backwards search, finding a path from the obstacle enclosed goal location to a free region is much easier (Kondo, 1991). Once a configuration free from obstacle influence is found through the backwards search, the forwards search, incorporating the danger

criterion, is initiated to this configuration. The posture potential must then be added to the forwards search cost function to ensure that the forwards and backwards paths join at the same robot configuration. This allows the planner goal and obstacle fields to be defined in the workspace, while still ensuring a contiguous path in the joint space.

6. Experiments

The planner was implemented and tested in a human-robot interaction trial (Kulić and Croft, 2006a). The experiment was designed to generate various robot motions and to evaluate both the human subjective response and physiological response to the motions. In this section, the subjects' subjective evalutation of the robot motion is described.

6.1. Experimental Method

The experiment was performed using the CRS A460 6 degree of freedom (DoF) manipulator, shown in Figures 11 - 14. The CRS A460 is a typical laboratory scale robot with a payload of 1kg, which is suitable for performing table top assistive activities. A group of 36 human subjects were tested; 16 were female and 20 were male. The age of the subjects ranged from 19 to 56, with an average age of 29.2. Approximately half of the subjects were recruited from the students and staff of the Mechanical Engineering Department and the University of British Columbia, and the other half were recruited from off campus. The subjects were also asked to rate their familiarity with robots on the Likert scale, with 1 indicating no familiarity, and 5 indicating excellent familiarity. Of the 36 subjects, 17 had little or no familiarity with robots (response of 1 or 2), 11 had moderate familiarity (response of 3) and 7 had high familiarity (response of 4 or 5). Each subject was tested once over a contiguous time period of approximately 25 minutes. Single trials of multiple subjects were selected over multiple trials of a single subject in order to capture a general response to the robot motions.

6.1.1 Trajectory Generation

Two different tasks were used for the experiment: a pick-and-place motion (PP), similar to the trajectory displayed to subjects in Nonaka et al. (Nonaka, Inoue et al., 2004) and a reach and retract motion (RR). These tasks were chosen to represent typical motions an articulated robot manipulator could be asked to perform during human-robot interaction, for example during hand-over tasks. For the pick-and-place motion, the pick location was specified to the right and away from the subject, and the place location was directly in front and close to the subject. For the reach and retract motion, the reach location was the same as the place location. For both tasks, the robot started and ended in the "home" upright position. The main difference between the two tasks is the approach direction of the robot. For the PP task, the robot approaches the subject from the side, while during the RR motion, the robot approaches the subject from head on.

Two planning strategies were used to plan the path of the robot for each task: a conventional potential field (PF) method with obstacle avoidance and goal attraction (Khatib, 1986), and the safe path method (S) described in Section 2 above. Figure 11,

Figure 12, Figure 13 and Figure 14 show frames of video data depicting each motion type.

Given the path points generated for each task by the two planners, a motion trajectory was generated using a minimum time cubic trajectory planner planning in configuration space. For each path, trajectories at three different speeds were planned (slow, medium, fast), resulting in 12 trajectories.

The trajectory planner generated a set of cubic path segments between each set of path points, resulting in a trapezoidal acceleration profile respecting velocity, acceleration and jerk limits. Each segment was described by a cubic polynomial, as shown in Equation 13.

$$Q(r) = b_0 + b_1 r + b_2 r^2 + b_3 r^3 \tag{13}$$

Where r is the parameterized time, b_i are the cubic coefficients, and Q is the resulting joint trajectory.

$$r = kt \tag{14}$$

where t is time and k is a constant scaling factor, which scales the velocity, acceleration and jerk on the trajectory. To generate the slow, medium and fast trajectories, k = 0.1, 0.5 and 1.0 were used, respectively. For both tasks and both planners, the robot comes to a stop in front of the person. The peak speeds along the path for both planners are the same, but because of dynamic and kinematic constraints of the robot, the velocity along the path cannot be identical, however the average speed along the path is within 10% for the two planners.

Figure 11 - Path PP-PF (Pick and Place Task Planned with the Potential Field Planner)

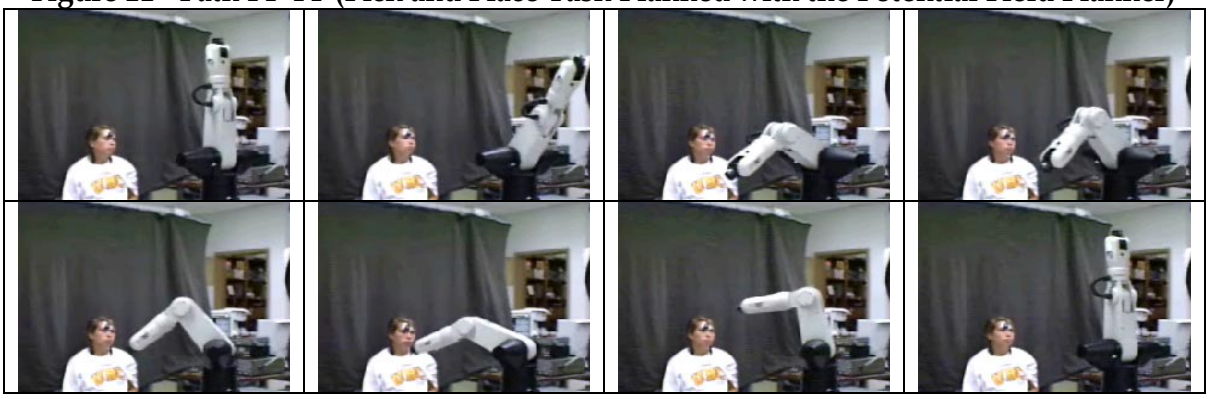


Figure 12 - Path PP-S (Pick and Place Task Planned with the Safe Planner)

Figure 13 - Path RR-PF (Reach and Retract Task Planned with the Potential Field Planner)

Figure 14 - Path RR-S (Reach and Retract Task Planned with the Safe Planner)

6.1.2 Experimental Procedure

For each experiment, the subject was asked to read a description of the experiment and sign a consent form. After signing the consent form, the experimental protocol was explained to the subject, and physiological sensors attached. The human subject was seated facing the robot. The robot then executed the 12 trajectories described above. The trajectories were presented to each subject in randomized order. After each trajectory had executed, the subject was asked to rate their own response to the motion in the following affective response categories: anxiety, calm and surprise. The Likert scale (from 1 to 5) was used to characterize the response, with 5 representing "extremely" or "completely" and 1 representing "not at all".

6.2. Results

The subject reported responses were analyzed to determine how the various robot motions affected the subject's perceived anxiety, calm and surprise, and to determine if the safe planned motions were perceived to be less threatening. Figure 15 and Figure 16 show a comparison of the average responses between the potential field and the safe planned paths for the subject rated anxiety and surprise, respectively.

As expected, for each trajectory, there is a strong positive correlation between anxiety and speed, and surprise and speed, and a negative correlation between calm and speed.

A comparison of the graphs in Figure 15 and Figure 16 indicates that for each motion type (pick and place or reach and retract), on average the subjects reported lower levels of anxiety and surprise, and higher levels of calm, for the safe planned paths. This observation is confirmed by a three factor analysis of variance (ANOVA) performed on the anxiety and surprise responses. The three factors are Plan (potential field (PF) vs. safe plan (S)), Task (reach and retract (RR) vs. pick and place (PP)), and Speed. Statistically insignificant factors were removed and the data re-analyzed until only statistically significant factors remained (Hicks, 1993). The statistically significant factors at $p < 0.05$ for anxiety and surprise are shown in Table 3 and Table 4, respectively. For these responses, the plan, speed and plan*speed interaction were found to be

statistically significant factors. The task factor and all the task interaction factors were found to be statistically insignificant. A Levene test was performed and confirmed the homogeneity of variances assumption at a significance level of 0.01%. For the subjective ratings (anxiety, calm and surprise), the results show a statistically significant reduction in anxiety and surprise (and increase in calm) when the safe planner is used when compared with the generic potential field planner at medium and high speeds. The plan*speed interaction indicates that at low speeds, the plan type does not affect subjective response, while at higher speeds, the motion plan significantly affects the perceived anxiety, surprise and calm. These relationships are found regardless of the type of task performed.

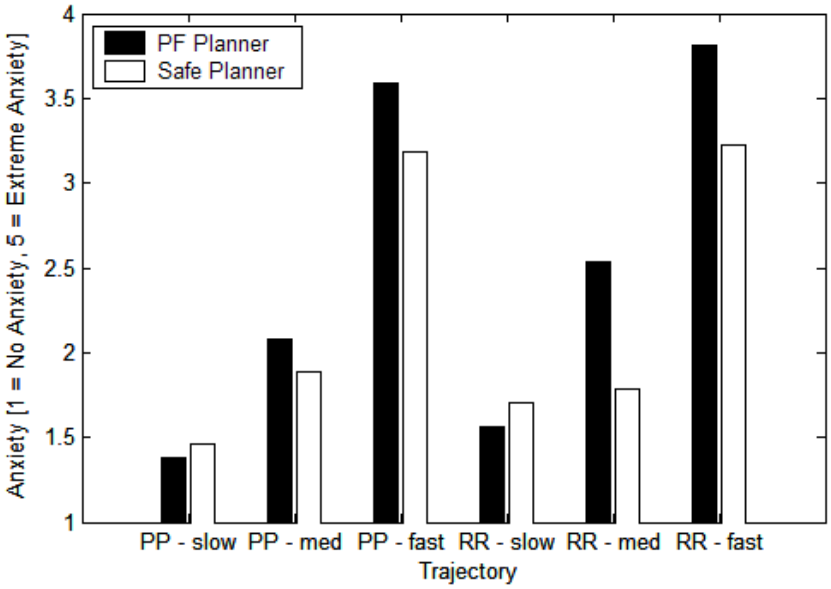


Figure 15 – Subject Reported Average Anxiety Response.

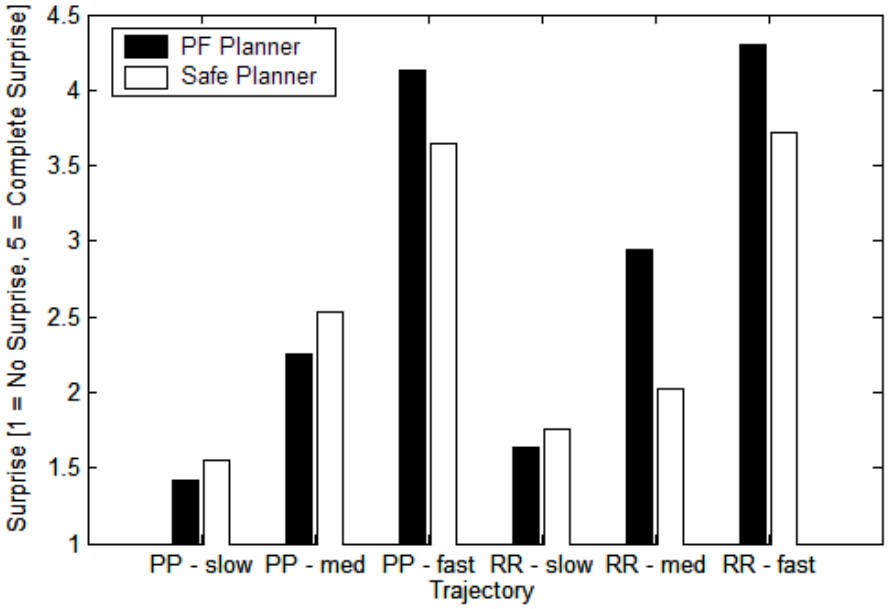


Figure 16 – Subject Reported Average Surprise Response.

Table 3 - ANOVA for Anxiety

Significant Factor	Sum Sq.	d.f.	Mean Sq.	F	Prob>F
Plan	9.0422	1	9.0422	9.8287	0.001837
Speed	301.39	2	150.695	163.8018	0
Plan*Speed	7.5428	2	3.7714	4.0994	0.017241
Error	391.9132	426	0.91998		
Total	709.8883	431			

Table 4 - ANOVA for Surprise

Significant Factor	Sum Sq.	d.f.	Mean Sq.	F	Prob>F
Plan	6.5023	1	6.5023	5.9369	0.015236
Speed	430.5104	2	215.2552	196.5382	0
Plan*Speed	8.4178	2	4.2089	3.8429	0.022177
Error	466.5694	426	1.0952		
Total	912	431			

7. Conclusions

The proposed safe planner reduces the factors affecting danger along the path. Using the two-stage planning approach reduces the depth of local minima in the cost function, allowing the planner to execute quickly. Minimizing the danger criterion during the planning stage ensures that the robot is in a low inertia configuration in the case of an unanticipated collision, as well as reducing the chance of a collision by distancing the robot centre of mass from the user. This advance-planning approach puts the robot in a better position to deal with real-time safety hazards.

When an inverse kinematics routine is available for an articulated robot, the performance of the planner can be further improved by adding a backwards search. That is, the path is generated backwards from the goal when the goal location is in an area crowded by obstacles. To ensure that the forwards and backwards generated paths meet, a posture potential is added to the total cost function. By including the posture potential directly into the cost function, rather than splining the two paths after they are generated, the algorithm ensures that posture correction occurs during low-danger sections of the path.

The proposed method was verified in simulation and experiments. A user study was also carried out to determine if the safe planned motion affected the perception of safety by users. Two types of robot motions were presented to human subjects during the study: motions planned with a conventional potential field planner, and motions planned with the safe planner. Subjects reported significantly less anxiety and surprise, and reported feeling more calm when safe planned motions were presented, as compared to the conventional potential field planner.

8. References

Ahuactzin, J. M. and K. K. Gupta (1999). The Kinematic Roadmap: A Motion Planning Based Global Approach for Inverse Kinematics of Redundant Robots. *IEEE Transactions on Robotics and Automation,* Vol. 15, No. 4, 653 - 669.

Barraquand, J. and J.-C. Latombe (1991). Robot motion planning: A distributed representation approach. *The International Journal of Robotics Research,* Vol. 10, No. 6, 628 - 649.

Bearveldt, A. J. (1993). Cooperation between Man and Robot: Interface and Safety. *Proceedings of IEEE International Workshop on Robot Human Communication*, pp. 183-187.

Bicchi, A., S. L. Rizzini, et al. (2001). Compliant design for intrinsic safety: General Issues and Preliminary Design. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1864-1869.

Bischoff, R. and V. Graefe (2004). HERMES - A Versatile Personal Robotic Assistant. *Proceedings of the IEEE,* Vol. 92, No. 11, 1759 - 1779.

Blanco, D., C. Balaguer, et al. (2002). Safe Local Path Planning for Human - Mobile Manipulator Cooperation. *Proceedings of IARP/IEEE-RAS Joint Workshop on Technical Challenge for Dependable Robots in Human Environments*.

Brock, O. and O. Khatib (2002). Elastic Strips: A Framework for Motion Generation in Human Environments. *The International Journal of Robotics Research,* Vol. 21, No. 12, 1031-1053.

Chen, M. and A. M. S. Zalzala (1997). A Genetic Approach to Motion Planning of Redundant Mobile Manipulator Systems Considering Safety and Configuration. *Journal of Robotic Systems,* Vol. 14, No. 7, 529-544.

Choset, H. and J. Burdick (2000). Sensor-Based Exploration: The Hierarchical Generalized Voronoi Graph. *The International Journal of Robotics Research,* Vol. 19, No. 2, 96 - 125.

Corke, P. I. (1996). A Robotics Toolbox for Matlab. *IEEE Robotics and Automation Magazine,* Vol. 3, No. 1, 24-32.

Corke, P. I. (1999). Safety of advanced robots in human environments, Discussion Paper for IARP.

Erkorkmaz, K. and Y. Altintas (2001). High Speed CNC System Design: Part I - Jerk Limited Trajectory Generation and Quintic Spline Interpolation. *International Journal of Machine Tools and Manufacture,* Vol. 41, No. 9, 1323-1345.

Gaskill, S. P. and S. R. G. Went (1996). Safety Issues in Modern Applications of Robots. *Reliability Engineering and System Safety,* Vol. 52, 301-307.

Guglielmelli, E., P. Dario, et al. (1996). Humans and technologies at home: from friendly appliances to robotic interfaces. *Proceedings of IEEE International Workshop on Robot and Human Communication*, pp. 71 - 79.

Hicks, C. R. (1993). *Fundamental Concepts in the Design of Experiments*. Saunders College Publishing New York.

Hwang, Y. K. and N. Ahuja (1992). Gross Motion Planning - A Survey. *ACM Computing Surveys,* Vol. 24, No. 3, 219-291.

Ikuta, K. and M. Nokata (2003). Safety Evaluation Method of Design and Control for Human-Care Robots. *The International Journal of Robotics Research,* Vol. 22, No. 5, 281-297.

Kavraki, L. E., P. Svestka, et al. (1996). Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces. *IEEe Transactions on Robotics and Automation,* Vol. 12, No. 4, 566 - 580.

Kawamura, K., S. Bagchi, et al. (1995). Intelligent Robotic Systems in the Service of the Disabled. *IEEE Transactions on Rehabilitation Engineering,* Vol. 3, No. 1, 14 - 21.

Khatib, O. (1986). Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *The International Journal of Robotics Research,* Vol. 5, No. 1, 90-98.

Khatib, O. (1995). Inertial Properties in Robotic Manipulation: An Object-Level Framework. *The International Journal of Robotics Research,* Vol. 14, No. 1, 19 - 36.

Kondo, K. (1991). Motion Planning with six degrees of freedom by multistrategic, bidirectional heuristic free space enumeration. *IEEE Transactions on Robotics and Automation,* Vol. 7, No. 3, 267-277.

Kulić, D. (2005). Safety for Human-Robot Interaction, University of British Columbia. Ph. D. Thesis.

Kulić, D. and E. Croft (2003). Strategies for Safety in Human Robot Interaction. *Proceedings of IEEE International Conference on Advanced Robotics*, pp. 644-649, Coimbra, Portugal.

Kulić, D. and E. Croft (2005). Safe Planning for Human-Robot Interaction. *Journal of Robotic Systems,* Vol. 22, No. 7, 383 - 396.

Kulić, D. and E. Croft (2006a). Physiological and Subjective Responses to Articulated Robot Motion. *Robotica,* In Press.

Kulić, D. and E. Croft (2006b). Safety Based Control Strategy for Human-Robot Interaction. *Journal of Robotics and Autonomous Systems,* Vol. 54, No. 1, 1 - 12.

Latombe, J.-C. (1991). *Robot Motion Planning*. Kluwer Academic Publishers Boston, MA.

Lee, C. W., Z. Bien, et al. (2001). Report on the First IART/IEEE-RAS Joint Workshop: Technical Challenge for Dependable Robots in Human Environments, IART/IEEE-RAS.

Lew, J. Y., Y. T. Jou, et al. (2000). Interactive Control of Human/Robot Sharing Same Workspace. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 535-539.

Macfarlane, S. and E. Croft (2003). Jerk-Bounded Robot Trajectory Planning - Design for Real-Time Applications. *IEEE Transactions on Robotics and Automation,* Vol. 19, No. 1, 42-52.

Maciejewski, A. A. and C. A. Klein (1985). Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments. *The International Journal of Robotics Research,* Vol. 4, No. 3, 109 - 117.

Martinez-Salvador, B., A. P. del Pobil, et al. (2000). A Hierarchy of Detail for Fast Collision Detection. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)*, pp. 745-750.

Nokata, M., K. Ikuta, et al. (2002). Safety-optimizing Method of Human-care Robot Design and Control. *Proceedings of Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, pp. 1991-1996, Washington, DC.

Nonaka, S., K. Inoue, et al. (2004). Evaluation of Human Sense of Security for Coexisting Robots using Virtual Reality. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2770-2775, New Orleans, LA, USA.

Oustaloup, A., B. Orsoni, et al. (2003). Path Planning by fractional differentiation. *Robotica,* Vol. 21, 59 - 69.

RIA/ANSI (1999). RIA/ANSI R15.06 - 1999 American National Standard for Industrial Robots and Robot Systems - Safety Requirements. New York, American National Standards Institute.

Traver, V. J., A. P. del Pobil, et al. (2000). Making Service Robots Human-Safe. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)*, pp. 696-701.

Wren, C., A. Azerbayejani, et al. (1997). Pfinder: Real time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, 780 - 785.

Yamada, Y., Y. Hirawawa, et al. (1997). Human - Robot Contact in the Safeguarding Space. *IEEE/ASME Transactions on Mechatronics,* Vol. 2, No. 4, 230-236.

Yamada, Y., K. Suita, et al. (1996). A failure-to-safety robot system for human-robot coexistence. *Journal of Robotics and Autonomous Systems,* Vol. 18, 283 - 291.

Yamada, Y., T. Yamamoto, et al. (1999). FTA-Based Issues on Securing Human Safety in a Human/Robot Coexistance System. *Proceedings of IEEE Systems, Man and Cybernetics SMC'99*, pp. 1068-1063.

Yu, Y. and K. K. Gupta (2000). Sensor-based Probabilistic Roadmaps: Experiments with an Eye-in-Hand System. *Advanced Robotics,* Vol. 14, No. 6, 515 - 536.

Zurada, J., A. L. Wright, et al. (2001). A Neuro-Fuzzy Approach for Robot System Safety. *IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews,* Vol. 31, No. 1, 49-64.