Physical Human-Robot Interaction: Increasing Safety by Robot Arm's Posture Optimization

Omar W. Maaroof and Mehmet İsmet Can Dede

Department of Mechanical Engineering, Izmir Institute of Technology
Izmir, Turkey

Abstract. To have robot manipulators working alongside with humans is a necessity in service robots. Obviously, in these robotics applications, human safety has precedence over precision and repeatability, which are the most important qualification of the conventional industrial manipulators. The safety measures can be taken either in the hardware or in the software or in both. This work by using a redundant manipulator aims at providing a safety measure through controlling the self-motion of the manipulator. The self-motion of the manipulator is controlled to change the posture of the manipulator to minimize or maximize the forces it can exert along a given direction. In this way, by knowing the location of the human or a delicate piece that it should not harm, manipulator's posture is optimized to exert the minimum amount of forces during an unexpected collision. The control algorithm for this objective is described in this paper and it is evaluated through simulation tests on a redundant lightweight robot manipulator.

Keywords—self-motion; redundant manipulator; service robots; robot posture optimization; sub-task control

1. Introduction

Robot manipulators have been used in industrial production lines for many years. In recent years, robotic arms are started to be used in other areas of life as in surgeries, rehabilitation and in daily life services. In these new fields of use, human-robot interaction becomes a necessity. A new term, Socially Assistive Robotics (SAR), was defined in Feil-Seifer et al. (2005) for the service robots that work with humans. This new generation of robots are built to be directly in contact with new kind of users which are not necessarily the operators. As Forlizzi et al., (2004) suggest, these users are either disabled or elderly individuals who do not have full awareness of robots but need robot service for every-day assistance. In recent years, robot manufacturers and researchers designed robots specifically for SAR applications.

SAR research field is focused on the physical Human–Robot Interaction (pHRI) since the safety and dependability measures for the robots used in SAR applications have an increased importance compared with precision and repeatability measures set for the conventional industrial robots. The need of safety and dependability measures were discussed in the studies of Haddadin et al., (2009) based on impact tests of a lightweight robot with a crash-test dummy for possible injuries that can happen in a SAR system and the severity of these injuries.

In research project with the acronym PHRIDOM, see (De Santis et al., 2008), the components of a robotic application are discussed based on safety and dependability in pHRI. The components mentioned were mechanics, actuation control techniques and real-time planning for safety measures. Sensors and fault handling were also discussed for dependability issues in a SAR design. In order to improve the safety and dependability of SAR systems, Pervez and Ryu (2008) give future directions of pHRI: multi-level protection strategy, failure management, safety enhancement through diversity and redundancy of sensors, and further work on robot control.

One of the approaches to avoid injuries in SAR applications is real-time planning. Some studies on evaluating the danger level based on the attributes of the human are provided by Najmaei et al. (2010), and impact force, effective robot inertia, the relative velocity, and the distance between the robot and the human are discussed by Kulić and Croft (2007). In these studies, real-time planning is performed based on the estimated danger levels. However, the human behavior is not easy to predict. Therefore, it is always possible to have a collision of the robot with the human during operation. For such cases, design of the robot and the controller plays a big part to avoid injuries. Haddadin et al. (2008) show how reactive control strategies can contribute to ensuring human safety during physical interaction.

In this work, increasing the safety during pHRI with respect to impact forces is studied. A new approach is proposed by utilizing a redundant robot to minimize the injury possibilities prior to detecting the contact. In order to accomplish this, posture optimization is employed to make a redundant robot arm to change its posture to minimize the impact forces along a given direction while carrying out the main task. This direction is selected to be along the vector from the end-effector to the assisted human. As a result of this, during a collision of the human and the end-effector, forces exerted on the human is minimized, which is another safety measure that can be employed in software. Previously, some researchers Walker (1990) controlled the self-motion of the redundant robot to resolve redundancy for reducing impact forces.

Our previously designed controller appeared in Maaroof et al. (2012) is selected for the purpose of controlling the self-motion of a redundant manipulator. The desired sub-task to be performed by this controller is selected to optimize the posture for static force exertion based on the work presented in (Chiu, 1988).

A commercially available redundant robot arm, 7-Degrees-of-Freedom (DoF) LWA4-Arm by SCHUNK, is selected to be used in the tests of the developed controller since this robot arm was used in a SAR application in Martens et al. (2007). When the main-task is designated as following a 3D position trajectory, this robot arm has 4 extra DoF, which makes the system more flexible in terms of possible number of different postures during operation. Simulation tests of the controller for minimizing the impact forces by optimizing the posture are carried out and results that are validating the efficiency of the new designs are given in the next sections.

2. Null Space Concept in Redundant Robot Arm Kinematics

Redundant manipulators have larger number of DoF, *n*, than the DoF required by the task, *m*. The end-effector pose in the task space, denoted by $x(t) \in \Re^{m}$, is defined as a function of joint position vector as $x = k(q)$. $k(q) \in \Re^{m}$, where $m \in \aleph$, represents the forward kinematics calculation, $q(t) \in \Re^{n}$ is the link position vector of an *n*-link manipulator and the relationships between the end-effector motion and the link motion in velocity and acceleration levels are obtained as $\dot{x} = J(q)\dot{q}$ and $\ddot{x} = \dot{J}(q)\dot{q} + J(q)\ddot{q}$, respectively.

$J(q) = \partial k(q) / \partial q \in \Re^{m \times n}$, is the Jacobian matrix of the manipulator, and from here on, it will be referred as *J*. $\dot{q}(t), \ddot{q}(t) \in \Re^{n}$ denote the link velocity and acceleration vectors, respectively. Since *J* is not a square matrix for redundant manipulators (*m* < *n*), one can use the *pseudo-inverse*, $J^{+} = J^{T}(JJ^{T})^{-1}$, to obtain the inverse kinematics relations. This can be accomplished when *J* has full rank (the manipulator is not in a singular configuration). The pseudo-inverse is defined by (Golub and Van Loan, 1983) such that the equalities in (1) are satisfied.

$$(J^{+}J)^{T} = J^{+}J, \quad (JJ^{+})^{T} = JJ^{+}, JJ^{+}J = J, \quad J^{+}JJ^{+} = J^{+}. \tag{1}$$

However, in order to represent the inverse kinematics calculations in velocity and acceleration levels, joint velocities, $\dot{\theta}_{N}$, and accelerations, $\ddot{\theta}_{N}$, in the null space of *J* are included to the formulation as $\dot{q} = J^{+}\dot{x} + \dot{\theta}_{N}$ and $\ddot{q} = J^{+}(\ddot{x} - \dot{J}\dot{q}) + \ddot{\theta}_{N}$, respectively.

The velocities in the null space are designed so that the main task execution is not affected by these extra motion of the joints. The design of the self-motion controller to regulate the null space motion is explained in the next section.

3. Control Design for the Redundant Robot

In this work, controller's main objective is to track a desired end-effector motion demand taking into account the dynamics of the manipulator. In order to achieve this, a control torque input signal, τ(t), has to be designed. The torque command signal should also include necessary information to execute sub-tasks defined by an optimization measure that regulates the self-motion of the manipulator. The controller design is a modification based on the controller developed by Hsu et al., (1989). In this paper, the motion tracking task taking place in task space is called as the main-task and regulating the self-motion of the manipulator for the selected optimization criterion is called as the sub-task.

Control Design for the Main-Task. The dynamic model for an n-link, robot manipulator can be represented as $M(q)\ddot{q} + C(q,\dot{q}) + G(q) + F(\dot{q}) + \xi_d = \tau$, where $M(q) \in \Re^{n \times n}$ represents the generalized inertia matrix, $C(q,\dot{q}) \in \Re^{n \times n}$ represents the torques due to centripetal-Coriolis effects vector, $G(q) \in \Re^n$ is the gravity vector, $F(\dot{q}) \in \Re^n$ represents the friction effects vector, $\xi_d \in \Re^n$ is a vector containing the unknown but bounded, disturbance effects and $\tau(t) \in \Re^n$ is the torque input vector which can be calculated, if the computed torque technique is used, as;

$$\tau = M_c \left\{ J^+ (\ddot{x}_d + K_v \dot{e} + K_p e - \dot{J}\dot{q}) + \ddot{\theta}_N \right\} + N_c \tag{2}$$

where x_d is the desired position defined in task space, $e = x_d - x$ is the tracking error, K_v and K_p are the constant feedback gain matrices, $M_c(q)$ is the computed generalized inertia matrix, $N_c(q,\dot{q})$ is the computed nonlinear terms that appear in the dynamics equation of the robot, which are the Coriolis, centripetal, gravitational, frictional effects, and $\ddot{\theta}_N$ is the designed joint acceleration vector designed in the null space of *J*. If the manipulator does not go through a singular condition, then the control law presented in (2) guarantees that the tracking error converges to zero exponentially given that we can calculate the generalized inertia matrix and nonlinear terms with some precision ($M_c \cong M$, $N_c \cong N$) as presented by Maaroof et al. (2012).

Control Design for the Sub-Task. For any subtask assigned, a vector function $g(\cdot) \in \Re^n$ is designed. This vector function may be a function of time, current state, etc. The objective of the sub-task controller is to make the null space joint velocity to track the projection of g onto the null space of *J*. Since $(I - J^+ J)$ projects vectors onto the null space of *J*, this can be formulated by an error signal calculation as presented in (3) which can approach to zero by a suitable controller.

$$\dot{e}_N = (I - J^+ J) g - \dot{\theta}_N \tag{3}$$

Assuming that the manipulator does not go through a singularity condition, it is required to design $\ddot{\theta}_N$ to meet the sub-task objective. The missing part in the control design given in (2), which is $\ddot{\theta}_N$, is formulated in (4).

$$\ddot{\theta}_N = (I - J^+ J)\dot{g} - (J^+ \dot{J} J^+ + \dot{J}^+) J g + K_N \dot{e}_N \tag{4}$$

In (4), K_N is a positive definite feedback gain matrix. With this formulation, the joint velocities in the null space converge to, $(I - J^+ J)$ i.e., $\dot{e}_N$, and the tracking error e converges to zero. The proof of this convergence is extensively explained in Maaroof et al. (2012).

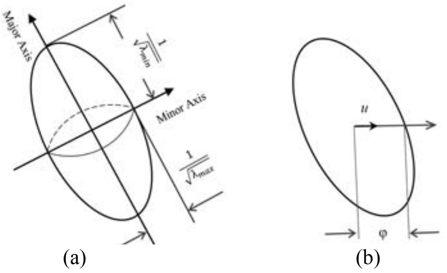
Objective Function for Sub-tasks in General. The projection of function g into the null space of J can be considered as formulation of the desired null space joint velocities that are needed to accomplish a given subtask. In order to control the self-motion of the joint velocities, the gradient g (or its negative) of the objective function $f(q)$ with a gain, k, can be used as $g = k\nabla f$. In the next section, the objective function for minimization of static force configuration is presented.

4. Posture Optimization for the Static Impact Force Minimization

The objective for this sub-task is to keep the robot manipulator in a posture that minimizes the ability to withstand external static impact force from the environment in a given end-effector position and for specific direction of force. The manipulating force measure is a scalar, w_f, based on the static environment-manipulator reaction force/torque vector, F, relationship given in (5) given by Walker (1990).

$$w_f = 1/w_m = 1/(\sigma_1 \ \sigma_2 \cdots \sigma_m) \tag{5}$$

Manipulating force ellipsoid is defined similar to the manipulability ellipsoid, w_m . In this case, large principal axis directions are associated with directions in which large static forces can be generated and vice-versa.


Major Axis
$\frac{1}{\sqrt{\lambda_{min}}}$
Minor Axis
$\frac{1}{\sqrt{\lambda_{max}}}$
u
φ

(a) (b)

Figure 1. Force ellipsoid (a) Ellipsoid axes, (b) Force transfer ratio in direction *u*.

The force ellipsoid is perpendicular to the manipulability ellipsoid in the sense that their magnitudes in each principal axis direction are inversely proportional. Force ellipsoid can be defined by; $F^T(JJ^T)F$. The optimal direction for exerting the maximum force is along the major axis of the force ellipsoid which coincides with the eigenvector of the matrix JJ^T corresponding to its largest eigenvalue λ_{max} as indicated in Figure 1.a. The force transfer ratio along a certain direction is equal to the distance from the center to the surface of the force ellipsoid along this vector as it can be observed in Figure 1.b. In Figure 1.b, *u*, is the unit vector along the desired force direction and φ is the force transmission ratio along *u*. Since φ*u* is a point on the surface of the ellipsoid, it should satisfy the following equation:

$$(\varphi u)^T (JJ^T)(\varphi u) = 1 \tag{6}$$

Re-arranging (6), an impact force magnitude measure can be derived for φ. Chiu (1988) proposed to maximize the following kinematic function presented in (7) (task compatibility index) for maximum force configuration making use of φ.

$$f(q) = \varphi^2 = \frac{1}{u^T (JJ^T) u} \tag{7}$$

In this work, utilizing (7) to formulate the objective function as shown in (8), impact force sub-task objective is defined to limit the impact force with the environment at the lowest level as used by Tatlicioglu (2009).

$$f(q) = k\, u^T (JJ^T) u \tag{8}$$

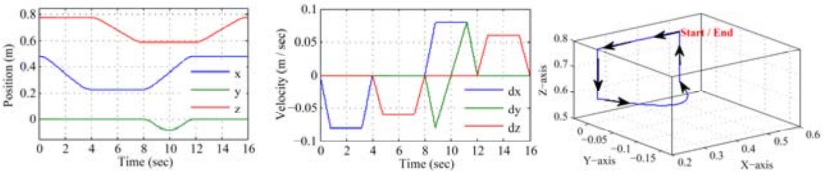
5. Simulation Test Results

A set of simulation results are presented in this section in order to evaluate the performance of the proposed controller. In these simulations, the aim is to utilize the virtual dynamic model of the 7-DoF LWA4-Arm produced by SCHUNK GmbH. The manipulator's CAD model is retrieved from the producer's website and the dynamic model is then transferred to the simulation environment, MATLAB® Simulink, in which simulations are carried out at a fixed-step sample time of 0.1 kHz.

The manipulator is initialized from a rest condition at the following link positions q = [0 -25 0 -35 0 -10 0]^T in degrees. Figure 2 shows the desired task-space trajectories for all simulations. The main-task is selected as tracking a position trajectory in Cartesian space and the end-effector orientation is left free.

The 7-DoF manipulator has 4 extra DoF as a consequence of this task description, which provides more flexibility in optimization. In the controller that was presented in (2), the nonlinear terms that include computed centripetal and Coriolis, frictional and disturbance effects are neglected since robot moves in slow motion. However, gravitational effects are used in the nonlinear effect cancellation term, N_c.

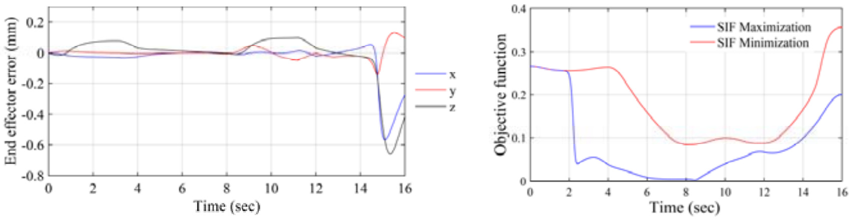
In the simulation tests, the objective functions are set in one of them to minimize and in the other one to maximize the static impact force magnitude. Force direction, which is the direction that the static impact force to be minimized, is selected to be along *x*-axis direction, u = [1 0 0]^T. Figure 3(a), shows position trajectory tracking error for the end-effector's tip point in the static impact force minimization subtask. The error is bounded with 0.7 mm that indicates the main-task objective is satisfied.



(a) (b) (c)

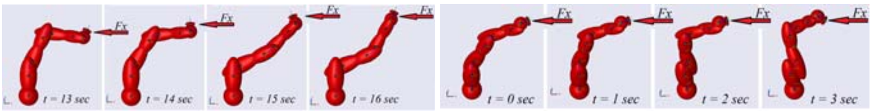
Figure 2. Desired task-space trajectories: (a) in 3D, (b) position level, (c) velocity level.

The sub-task objective results for both sub-tasks are presented in Figure 3(b). It can be observed that the sub-task objective measure for maximization of manipulability along x-axis is kept at higher values in static impact force minimization task with respect to maximization task. As a final note, subtask controller performance is validated since the subtask error signal is bounded by 0.015 rad/sec. A configuration change appears in the result presented in Figure 3(a) after second 14. This sudden change results in higher velocity demands at joint level and thus, larger errors in the main-task trajectory tracking. Configuration change after second 14 is illustrated in Figure 4(a) with virtual manipulator's screenshots taken during the simulation.



(a) (b)

Figure 3. (a) Tip point position error (b) Objective function magnitude.



(a) (b)

Figure 4. Robot arm motion during simulation: (a) minimizing SIF, (b) maximizing SIF

After second 14, the second, fourth and sixth joints are moved with higher velocities to change the configuration as shown in Figure 4(a). As a result of this, the manipulability along *x*-axis is maximized and the static impact force is minimized and thus, the subtask objective is satisfied. In contrast, Figure 4(b) reveals how the static impact force is maximized when robot arm configuration is changed to rotate manipulability ellipsoid placing the shortest diameter along the x-axis.

6. Conclusions

The aim in this paper is to study a different safety measure in SAR applications by making use of redundant robot and the control of a redundant robot's self-motion. To achieve this, a general sub-task controller utilizing self-motion property of redundant robot manipulator was adapted to change the posture of the manipulator to minimize the static force exertion to the environment along a certain direction. The controller is then tested in simulations on a 7-DoF robot arm model. The main-task assigned for the manipulator was to track the position trajectory for the end-effector's tip point in Cartesian space, which allowed more DoF to be used for posture optimization. Test results indicated that the main-task objective is reached by keeping the tracing error bounded. It should be noted that maximum errors occurred during sudden configuration changes when higher joint velocities are required. This is a result of excluding centripetal and Coriolis terms in nonlinearity cancellation. The sub-task objective for minimizing the static force along *x*-axis, is also satisfied, which is observed by bounded errors in the sub-task error signal and the posture changes in the animation of the robot arm during tests. Overall, it can be stated that the stability and effectiveness of the designed controller is verified and this controller can be used in SAR applications using redundant manipulators to increase the safety of the application.

Acknowledgements

This work is supported in part by The Scientific and Technological Research Council of Turkey via grant number 115E726.

Bibliography

Chiu, S. L. (1988). Task compatibility of manipulator postures. *The International Journal of Robotics Research* 7: 13-21.

De Santis, A., Siciliano, B., De Luca, A., and Bicchi, A. (2008). An atlas of physical human–robot interaction. *Mechanism and Machine Theory* 43: 253-270.

Feil-Seifer, D., and Matarić, M. J. (2005). Defining socially assistive robotics. *In proceeding of the 9th International Conference on Rehabilitation Robotics*, June 2005, ICORR 2005, IEEE, 465-468.

Forlizzi, J., DiSalvo, C., and Gemperle, F. (2004). Assistive robotics and an ecology of elders living independently in their homes. *Human-Computer Interaction*, June 2004, 19(1), 25-59.

Golub, G.H., and Van Loan, C.F. (1983). Matrix Computations, Baltimore, MD: The Johns Hopkins Press.

Haddadin, S., Albu-Schäffer, A., Luca, A. D., and Hirzinger, G. (2008). Collision detection and reaction: A contribution to safe physical human-robot interaction. In *proceeding of the International Conference on Intelligent Robots and Systems,* Nice, France, Septemper, 2008, IROS 2008. IEEE/RSJ, 3356-3363.

Haddadin, S., Albu-Schäffer, A., and Hirzinger, G. (2009). Requirements for safe robots: Measurements, analysis and new insights. *The International Journal of Robotics Research*, 28(11-12), 1507-1527.

Hsu, P., Mauser, J., and Sastry, S. (1989). Dynamic control of redundant manipulators. *Journal of Robotic Systems* 6: 133-148.

Kulić, D., and Croft, E. (2007). Pre-collision safety strategies for human-robot interaction. *Autonomous Robots* 22: 149-164.

Maaroof, O. W., Gezgin, E., and Dede, M. İ. C. (2012). General subtask controller for redundant robot manipulators. In *proceeding of the 12th International Conference on Control, Automation and Systems,* (ICCAS), JeJu Island, South Korea, October, 2012, IEEE, 1352-1357.

Martens, C., Prenzel, O., and Gräser, A. (2007). The rehabilitation robots FRIEND-I & II: Daily life independency through semi-autonomous task-execution. *INTECH Open Access Publisher*, Vienna, Austria, 2007, ISBN 978-3-902613-01-1.

Najmaei, N., Lele, S., Kermani, M. R., and Sobot, R. (2010). Human factors for robot safety assessment. In *proceeding of ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, July, 2010, IEEE, 539-544.

Pervez, A., and Ryu, J. (2008). Safe physical human robot interaction-past, present and future. *Journal of Mechanical Science and Technology* 22: 469-483.

Tatlicioglu, E., Braganza, D., Burg, T. C., and Dawson, D. M. (2009). Adaptive control of redundant robot manipulators with sub-task objectives. Robotica 27: 873-881.

Walker, I. D. (1990). The use of kinematic redundancy in reducing impact and contact effects in manipulation. In *proceeding of the International Conference on Robotics and Automation,* OH, USA, May, 1990, IEEE, 434-439.