FORCE/POSITION CONTROL OF ROBOT MANIPULATOR FOR HUMAN-ROBOT INTERACTION

by

Paramin NERANON [a*] ***and Robert BICKER*** [b]

[a] Faculty of Engineering, Prince of Songkla University, Songkhla, Thailand
[b] School of Mechanical and Systems Engineering, Newcastl University,Newcastl, UK

Original scientific paper
DOI:10.2298/TSCI151005036N

With regard to both human and robot capabilities, human-robot interaction provides several benefits, and this will be significantly developed and implemented. This work focuses on the development of real-time external force/position control used for human-robot interaction. The force-controlled robotic system integrated with proportional integral control was performed and evaluated to ensure its reliably and timely operational characteristics, in which appropriate proportional integral gains were experimentally adopted using a set of virtual crank-turning tests. The designed robotic system is made up of a robot manipulator arm, an ATI Gamma multi-axis force/torque sensor and a real-time external PC based control system. A proportional integral controller has been developed to provide stable and robust force control on unknown environmental stiffness and motion. To quantify its effectiveness, the robotic system has been verified through a comprehensive set of experiments, in which force measurement and ALTER real-time path control systems were evaluated. In summary, the results indicated satisfactorily stable performance of the robot force/position control system. The gain tuning for proportional plus integral control algorithm was successfully implemented. It can be reported that the best performance as specified by the error root mean square method of the radial force is observed with proportional and integral gains of 0.10 and 0.005 respectively.

Key words: *robot force/position control, proportional integral control, human-robot interaction*

Introduction

Human-robot interaction (HRI) has become the crucial aspect when robots have been used for collaboration with humans in industrial applications, due to the requirements of technological feasibility and productivity improvements in terms of quality, accuracy reliability and flexibility. Interest in human-robot interaction has tended to increase significantly. Consequently, various human-robot cooperative technologies, which are used to enable unskilled workers to be able to directly teach intelligent robots, have been developed. For example, when a human operator gives instructions about task trajectory to a manipulator, the ordered trajectory can be automatically created by the robot instead of requiring offline programming [1].

The HRI interaction has been investigated significantly since 1994 [2], and interactive control methods were previously applied in basic on-off control systems or manipulator

* Corresponding author; e-mail: nparamin@me.psu.ac.th

joint control systems using analogue joysticks. Human-robot collaborative technology has since developed so as to be more intelligent, smooth, natural, and safe, as in human-human interactive relationships. The HRI has also been defined as *the study of humans, robots, and the ways they influence each other* [2]. Much of the relevant HRI research has been extensively reviewed and it can be clearly seen that one of the crucial aspects in improving human-robot cooperation is to develop an external real-time force/position control system of robot's behaviour in real-time. Robot force control is a fundamental requirement in the achievement of the control of the robot's real-time path in any physical robot interaction task. It has been developed in the past three decades, using for example force, torque and visual feedback to operate robots to participate in unstructured environments. Therefore, this paper highlights on the development of real-time external force/position control suitably used for human-robot object handover tests, in which a human handler is able to dexterously pass an object to the robot acting as a receiver in a timely and natural manner, and *vice versa*.

Fundamentals of robot force control

Robot manipulators have been widely used in industry for many years to perform several tasks involving interaction with their environment, where the robots are required to satisfy the specific position trajectory and control. In order to achieve effective motion execution, in which robot's end effector has to manipulate an object or moving along a designed surface, robotic behavioural control of the interaction between the robot and its environment is crucial. The key performance issue in design of a robotic force control system is that of stability. The system has to perform in a stable and reliable manner while operating and contacting with various unknown stiffness environments. The term *compliant motion* has been defined as a manipulation task which specifies the contact force between a robot manipulator and its environment. The positions of the robot end effector are appropriately controlled by interactive forces whilst executing a physical interaction task. Motion control can be classified into two key groups: passive compliance and active compliance. Passive compliance is where the robot end effector position is modified by the contact force because of the inherent compliance of the robot, whereas active compliance facilitates a programmable robot reaction using a force feedback signal, for which purpose the robot control system has been designed [3]. Active compliance is used to ensure effective control and overcome the disadvantages of passive compliance. Typically, this contact force and torque feedback signals are measured by a multi-axis force/torque sensor before being transferred to robot controller in order to generate an updated trajectory of robot end effector [4].

In practice, it may impossible to appropriately control commercial robot manipulators using explicit hybrid position/force control or force-based impedance control because the commercial robots are generally developed as positioning devices. However, when using implicit or position-based force control (external force control) it allows robot manipulator to respond to the environment and also to compensate for variations in robot positioning at the contact surface [5]. The key features of this technique provide reliability and stability because switching between position and force loops is avoided. Both position and force control are handled in the same Cartesian direction. De Schutter and Van Brussel [6] reported that a fundamental requirement for success of human-robot interaction is the capability of the robot to handle physical contact between the robot and the human. Using implicit position-based force control or external force control (see fig. 1) developed by De Schutter and Van Brussel [6] is considered to offer a better solution regarding the safety constraints, simplicity and implementation efficiency [7]. The most key aspect in this control method is to achieve a suitable compromise between the system response and stability, where the response time was required to be as short as possible. It should be noted that the system oscillations will be introduced when the control gains are too high.

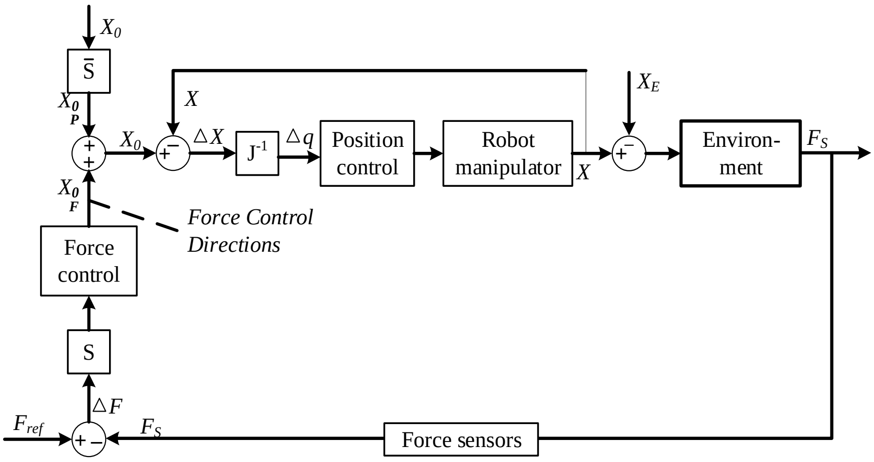


Figure 1. Position-based implicit force control (external force/position control) [6]

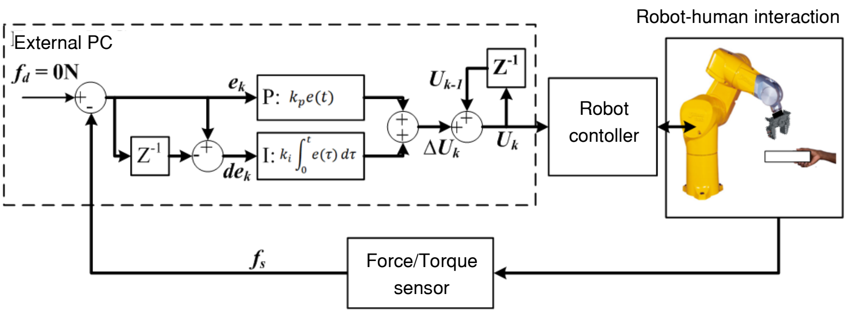


Figure 2. Schematic diagram of the force control strategy based on PI control

Implementation of force robot control

This section describes the design and implementation of external force control of robotic system in HRI. The control of physical HRI is a challenging area of research, and a number of research projects have proposed force-feedback control using external force control algorithms to alter the robot trajectory. Figure 2 illustrates an overall schematic diagram of the external force/position control algorithm of a robot manipulator used in HRI. The interactive force between the human and robot is measured using a 6-axis force/torque sensor. The Staubli robot controller communicates in real-time with an external PC via an Ethernet port using the TCP/IP protocol. The external PC (running under Linux) processes data transmitted by sensors and generates changes in incremental position to modify the robot's path using proportional integral control which is explained in the section *Implementation of proportional integral force control*. A detailed description of hardware configuration and integration is given in the following sections.

Staubli robot manipulator arm (TX60)

A six-degree of freedom Staubli TX60 robot manipulator arm was adopted since it is able to perform real-time path control, with appropriate speed, accuracy and reliability. The TX60 robot has a normal payload capacity of 3.5 kg (maximum of 9 kg) and repeatability of ±0.02 mm. The real-time path control can be updated every 4 ms, and transmission control protocol/internet protocol (TCP/IP) interface is available with a net bit rate capacity of 100 Mbit/s. The Staubli robot system is made up of three key components consisting of a Staubli manipulator, a robot controller and a robot manual control panel (MCP). The CS8C Staubli robot controller is a multi-processor system which is able to control the basic robot inputs/outputs, with a fieldbus interface board (also supporting TCP/IP client).

Multi-axis force/torque sensor

A 6-axis force/torque sensor was used to detect the interaction force between human participant and robot manipulator arm. The ATI Gamma Multi-Axis Force/Torque sensor was mounted between the robot end effector and gripper. The sensor system is made up of an ATI F/T Gamma sensor, an electrically shielded and twisted transducer cable and a stand-alone ATI controller in which optional analogue, parallel and serial outputs have been already attached. The ranges of force/torque measurements are up to ±130 N with 0.1 N resolution and ±10 Nm with 0.0025 Nm resolution, respectively. The ATI controller converts all strain gauge signals into the magnitudes of the Cartesian force/torque components using a calibration matrix computation.

Real-time Linux operating system

A real-time Linux operating system (RT Linux OS) was employed to ensure the achievement of robust control. The Ubuntu with Linux 3.2.0-23-realtime version was adopted because it is effective, stable, reliable, fast and powerful. Other outstanding features of Linux are that it allows program multitasking, multiplatform, multiprocessor and multithreading operation. The RT Linux supports multi-task execution using the multi-tasking kernel to manage user programs so that they can run simultaneously. In addition, transmit control protocol and internet protocol (TCP/IP) has been developed to facilitate the transfer of data between the external PC and the CS8C Staubli controller.

Software configuration

The software development is one of the key requirements in the HRI system design. Two crucial software operating systems were used consisting of the real-time Linux and Staubli VAL3 OS. Due to create a program, the RT Linux OS requires three components. The text editor is program employed for writing and editing texts, and the GNU compiler collection (GCC), C compiler is available in the RT Linux and associated with the standard C library. The library is a collection of sub-programs officially developed by programmers and can be used to reduce the amount of complex and repetitive source code. C code was developed to communicate with ATI F/T *Gamma* sensor and the CS8C Staubli robot controller over TCP/IP communication, and to facilitate the effective force feedback control. The VAL3 language is a high-level programming language developed to control Staubli robots. It combines the basic features of a standard real-time computer language with several specified functions, such as robot control, geometrical modelling and input/output control tools. The VAL3 program was designed to handle the applications of the path modification of robot moving in real-time, gripper operation control and multitasking systems.

Implementation of proportional integral force control

As suggested by De Schutter and Van Brussel [6], Volpe and Khosla [8] and Zeng and Hemami [9], proportional integral (PI) control is appropriate for robot force/position control in order to provide the smallest possible force control error, and because this technique facilitates an increase in the accuracy and stability of control system. Therefore, it was decided to apply simple proportional plus integral robot force control in this project. This control algorithm is preferable to proportional-integral-derivative (PID) control since the derivative term is sensitive to noise and this could lead to a destabilizing effect on the HRI system. Although, the derivative gain (K_D) which gives a reduction in the system overshoot and thus, settling time has been removed, the overshoot response can be controlled using an appropriate proportional gain [10].

An incremental discrete-time PI control algorithm with sampling time period τ and the discrete time interval k can be calculated by applying the eqs. (1) and (2).

The discrete-time PI control output:

$$U(k) = k_P e(k) + k_I \sum_{j=0}^{k} e(j) \tag{1}$$

The incremental PI control value represented by $\Delta u(k)$ can be calculated:

$$\Delta u(k) = k_P e(k) + k_I \sum_{j=0}^{k} e(j) - k_P e(k-1) + k_I \sum_{j=0}^{k-1} e(j) \tag{2}$$

$$\Delta u(k) = k_P[e(k) - e(k-1)] + k_I e(k) \tag{3}$$

Therefore, the incremental PI algorithm is defined by:

$$u(k) = u(k-1) + k_P[e(k) - e(k-1)] + k_I e(k) \tag{4}$$

$$e(k) = f_d - f_s \tag{5}$$

where $u(t)$ is the PI control output, K_P – the proportional gain, K_I – the integral gain, f_d – the desired force, which was initially defined as 0, and f_s – the actual force (measured by the ATI force sensor).

A block diagram of external force/position control based on the PI control algorithm of a robot manipulator for human-robot interaction is shown schematically in fig. 2, where e is the error defined as the difference in magnitude between the desired (f_d) and actual (f_s) forces, while de is the change in error (e). The PI control output was determined as the incremental displacement (ΔU_k) modified by the previous computed value of ΔU_{k-1}, which is scaled before being transferred to the TX60 Staubli robot's ALTER function to modify its trajectory.

Evaluation of force robot control

The HRI hardware and software architectures have been discussed in the previous section including those for the Staubli robot (TX60) and ATI *Gamma* multi-axis force/torque sensor; the real-time Linux operating system, TCP/IP communication, and the multitasking software design were also outlined. Proportional plus integral (PI) control method has been adopted to achieve the robot position's control based on force control system. In this section, it has been described how PI gains can be experimentally tuned based on the virtual crank-turning preliminary test. To ensure the reliability of robot force/position control in performing effective and accurate human-robot object handover tasks under varying conditions, three key experiments have been strategically evaluated as detailed in the following sections.

Evaluation of ATI Gamma multi-axis force/torque sensor

To establish the reliability and stability of force data acquisition in performing effective and accurate HRI tasks under varying conditions, the force sensor outputs were monitored and captured in real-time using of RT Linux. Four different conditions, in which modes 1 and 2 represent that the robot is not moving and modes 3 and 4 show that the robot is moving, were selected in order to investigate the robot's dynamic behaviour influencing the force sensing.

(a) Mode 1: Robot controller power off,
(b) Mode 2: Robot controller power on and arm power on,
(c) Mode 3: Robot controller power on, arm power on and robot moving in the x, y, and z axes with a standard moving command, and
(d) Mode 4: Robot controller power on, arm power on and robot moving in the x, y, and z axes under ALTER real-time path control.

Due to statistically collect sufficient volumes of data, 2,000 data were captured every 4 ms whilst executing the modes 1 and 2, whereas modes 3 and 4 allowed the robot to move 200 mm in the x-y-z plane at a velocity of 50 mm/s. Additionally, each mode was undertaken for 5 repetition sets, and the overall mean and standard deviation (SD) of the force reading errors were calculated. Table 1 demonstrates the results obtained with the four modes, in which there was no significant difference between the force values recorded along the x, y, and z axes represented by f_x, f_y, and f_z respectively. The 3-axis force values fluctuated between ±0.1 N in the different modes. The maximum SD for z-axis is approximately double the x and y values at ±0.11 N, while the maximum SD values for x and y were around ±0.056 and ±0.054 N, respectively.

Table 1. Means and standard deviations of 3-axis force sensor readings

Mode	f_x [N]		f_y [N]		fz [N]	
	Mean	*SD*	*Mean*	*SD*	*Mean*	*SD*
1	-0.088	0.032	-0.005	0.022	-0.068	0.065
2	-0.053	0.052	-0.049	0.054	-0.019	0.092
3	-0.006	0.032	-0.101	0.029	-0.025	0.105
4	-0.006	0.056	-0.098	0.048	0.024	0.108

Evaluation of ALTER real-time control path

When considering the human-robot object handover process, it was necessary to evaluate the robot's ALTER control system in order to ensure effective HRI performance, and thus, a set of experiments have been carried out. The main objective was to assess the performance of the robot real-time path control in terms of its reliability and accuracy. In these tests, a set of required free-space reference positions transmitted to robot controller were compared to actual robot end-effecter positions which were recorded by the CS8C Staubli controller. The less error measured in an experiment, the more effective in improving real-time path control.

The robot was required to move along circular paths of 100, 150, and 200 mm in diameter in a fixed time period, whereas its motion equation was simulated using MATLAB and drawn using 1500 points (N) in which the step size is defined by $2\pi/N$. To evaluate the quantitative performance of the ALTER function, the robot's actual positions whilst moving was compared to the desired values, and the following data recorded and compared, namely demanded, received (through TCP/IP) and actual values. Demanded values of incremental position were used to modify the robot's path and were generated in the external real-time Linux

PC and transmitted to the Staubli CS8C controller via an Ethernet ports using the TCP/IP protocol with a 4 ms cycle interrupt. Subsequently, received data, which represent the information acquired by the controller, were computed to establish an ALTER 3-axis transformation matrix in order to enable the changes in actual robotic movements. The circular paths were executed in a counter-clockwise direction, with the home position defined to allow robot to start at the same location. The robot's actual positions during moving along the path were concurrently stored and compared with the desired positions in real-time whilst performing the test in order to calculate the overall mean and standard deviation of position errors.

The results indicating the overall average and corresponding standard deviations of the x-y position errors whilst drawing the circular paths 100, 150, and 200 mm in diameter have been summarized in tab. 2. The mean errors of the x and y axes slightly increased and varied from minimum of 0.46% and 1.57% up to maximum of 2.82% and 2.64%, respectively. Additionally, the standard deviations of the two dimensions were in the range of 0.35-0.53 mm. According to the results, it can be concluded that the performance of ALTER real-time control in path modification can be improved by decreasing robot's velocity.

Table 2. Means and standard deviations percentage error for x-y axes

Circle of 50 mm diameter		
Test	Mean (%)	Standard deviation (%)
x-axis	0.46	0.44
y-axis	1.57	0.47
Circle of 75mm diameter		
Test	Mean (%)	Standard deviation (%)
x-axis	0.84	0.43
y-axis	1.70	0.35
Circle of 100mm diameter		
Test	Mean (%)	Standard deviation (%)
x-axis	2.82	0.39
y-axis	2.95	0.53

Virtual crank-turning preliminary tests

The main objective of this research is to develop real-time external force/position control which can be applied for human-robot object handover tasks. Therefore, this section addresses how to adopt appropriate proportional integral (PI) gains to ensure the effective position control algorithm which can be transferred directly to the robot to modify its trajectory. The trial and error method based on virtual crank-turning preliminary tests were carried out to establish appropriate PI control. In these tests, the robot end effector was programmed to move along constrained circular paths in which its velocity has been modified by human's external force applied to the robot. Two parameters consisting human's external radial force (F_R) and tangential force (F_T) were measured in real-time. The performance of the system response can be evaluated in terms of variation in radial forces, in which the lower the variation in radial forces, the better the performance of the system. The tests were undertaken by 18 human participants, and they were first instructed to perform the tasks with the best of their ability and to attempt to minimize the radial forces during task execution.

The procedure of the virtual crank-turning task permitted the robot to move with a constrained trajectory around the virtual crank radius, at a diameter of 200 mm, in a clockwise direction. The task was required to commence at the proposed home position, and the human participant was required to manipulate the robot gripper around the circular path, whilst attempting to minimize the radial force (F_R). The performance of the system response can be evaluated in terms of variation in the radial forces, in which the lower the variation in radial forces, the better the performance of the system. It also can be assumed that F_T represents tangential force.

The external force exerted by the participant was measured as forces in the x and y directions, represented as F_x and F_y respectively, in which the noises modified in the signals as mentioned in the section *Evaluation of ATI Gamma multi-axis force/torque sensor* was too small and then it was omitted. However these were subsequently transformed into tangential and radial forces. To calculate the incremental change in displacement (ΔU) based on PI control, eq. (6) is used, and by using the ALTER command, an ALTER transformation matrix ($ALTER_{trsf}$) is represented as:

$$\mathrm{ALTER}_{trsf}\begin{bmatrix}\mathrm{ALTER}_{trsf}.x\\ \mathrm{ALTER}_{trsf}.y\\ \mathrm{ALTER}_{trsf}.z\end{bmatrix} \tag{6}$$

The performance was analyzed based on the root mean square error (*E_RMS*) of the radial forces [11, 12]. The equation used to calculate the magnitude of error deviations of E_RMS is expressed as:

$$E_{RMS} = \sqrt{\frac{\sum_{i=1}^{n}(F_R - F_{De})^2}{n}} \tag{7}$$

where n is the number of evaluated values, F_R – the radial forces exerted by participant, and F_{De} – the demanded radial force (0).

The experimental virtual crank test was undertaken to examine the relationship between the root mean square error (E_RMS) of the radial force (F_R) and the tangential force (F_T) applied to the virtual crank. The results of the preliminary virtual crank-turning tests are illustrated the tab. 3.

Table 3. The results of virtual crank-turning preliminary tests to evaluate the gain K_P

K_P	*E_RMS* of radial force [N]		Tangential force [N]	
	Average	*Standard deviation*	*Average*	*Standard deviation*
0.025	2.34	0.13	8.37	0.34
0.050	1.21	0.11	4.24	0.23
0.075	0.77	0.09	2.78	0.15
0.100	0.53	0.08	2.14	0.12
0.125	0.65	0.08	1.69	0.10
0.150	0.92	0.12	1.39	0.13

Figure 3 presents the performance results of the virtual crank test for different values of proportional gain K_P ranging between 0.025-0.15 with 0.025 N resolution. The system performance can be identified based on the E_RMS of radial force.

The best performance of this test is represented by minimum E_RMS of (F_R), and was achieved at a gain K_P of 0.100, where minimum E_RMS value is 0.53 N with the minimum standard deviation of 0.08 N. As expected, the tangential force (F_T) decreases when the gain K_P increases because tangential force is approximately inversely proportional to the gain K_P gain value.

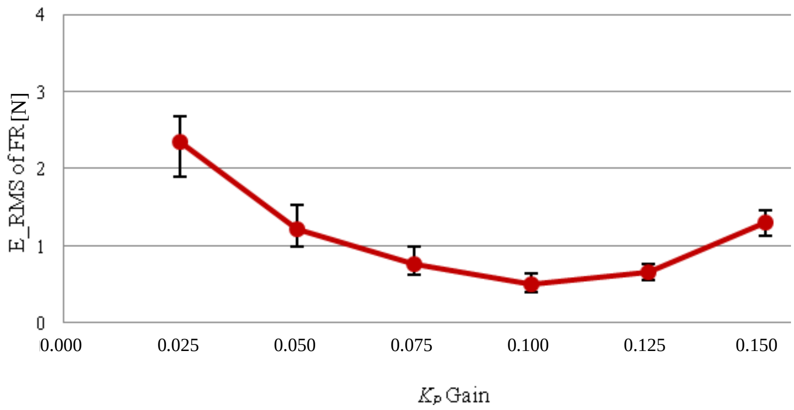
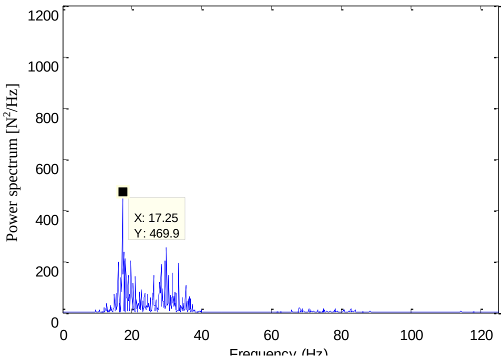


Figure 3. The E_RMS of radial forces with various proportional (K_P) gains applied

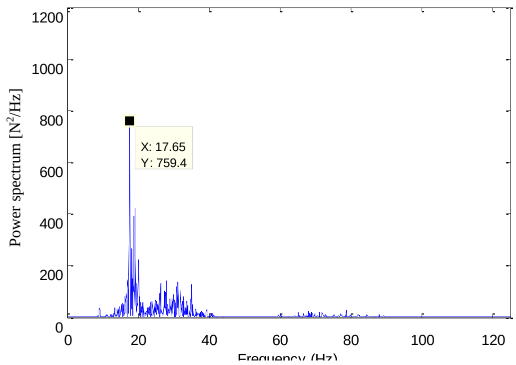
The results indicated that there was small oscillation moderating in the signal; therefore, the frequency domain evaluation of the force in the virtual crank tests at the six different K_P gains was determined using fast Fourier transform (FFT) [13]. To suitably identify the noise, a high-pass filter with a cut-off frequency at 10Hz was used. Figure 4 shows the results of the FFT analysis over the range of K_P (0.025-0.150); dominant frequency is in the range between 17-20 Hz with the density power spectrum varied between 450-1050 N^2, and higher frequency (28-30 Hz) was clearly seen at the highest K_P, as shown is fig. 4(f). It can be highlighted that an increase in a K_P gain gives an increase in higher frequency of the system response; however, if the $K_P \geq 0.175$, the robot system has very high unstable oscillation which could damage the robot. To optimize the integral gain (K_I), the gain K_P was set at 0.100, and then tuning of the integral gain mark by increasing K_I until the best E_RMS of F_R is achieved. The same procedure was used for the virtual crank test developed for evaluating the performance of the gain K_P. The same group of the participants was used to perform the assigned tests. A range of integral gains varying from 0.0025 to 0.0175 with 0.0025 N resolution was selected. The test results are shown in tab. 4.

Table 4.The results of virtual crank-turning preliminary tests to evaluate the gain K_I

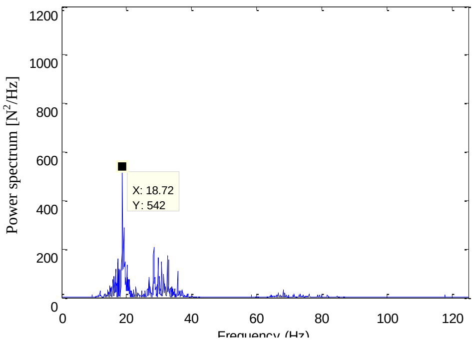
K_I	*E_RMS* of radial force [N]		Tangential force [N]	
	Average	*Standard deviation*	*Average*	*Standard deviation*
0.0025	0.53	0.40	2.19	0.49
0.0050	0.44	0.34	1.91	0.42
0.0075	0.45	0.34	1.79	0.45
0.0100	0.46	0.42	1.59	0.47
0.0125	0.53	0.70	1.52	0.61
0.0150	0.71	0.94	1.51	0.96
0.0175	0.92	1.36	1.49	1.50



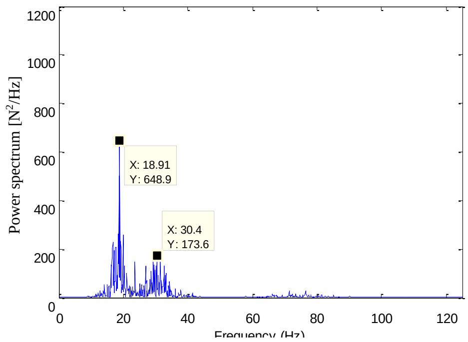
(a) K_P = 0.025



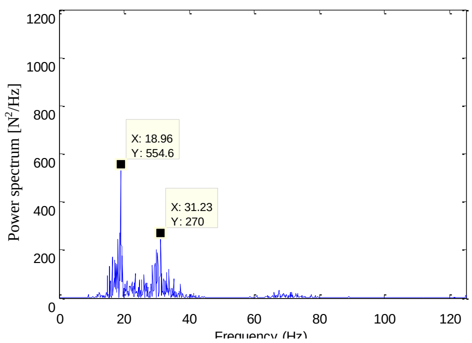
(b) K_P = 0.050



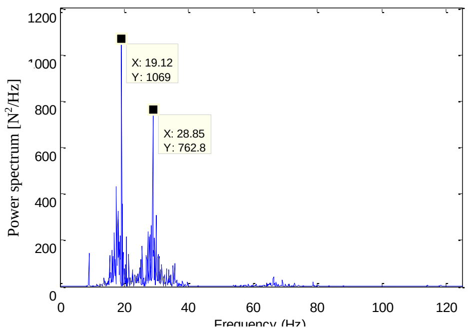
(c) K_P = 0.075



(d) K_P = 0.100



(e) K_P = 0.125



(f) K_P = 0.150

Figure 4. FFT analysis for virtual crank test

Performance of the virtual crank test for different values of integral gain (K_I) has been illustrated in fig. 5. The best performance of the K_I tuning test is defined as the gain K_I of 0.0050, in which the E_RMS value is 0.44 N with a standard deviation of 0.34 N. Increasing the gain K_I is accompanied by a decrease in the tangential force (F_T). In summary, the gain tuning for PI control applied to the robot's velocity and force control was implemented. The best performance as specified by the E_RMS of the radial force is observed with proportional and integral gains of K_P = 0.10 and K_I = 0.005 respectively.

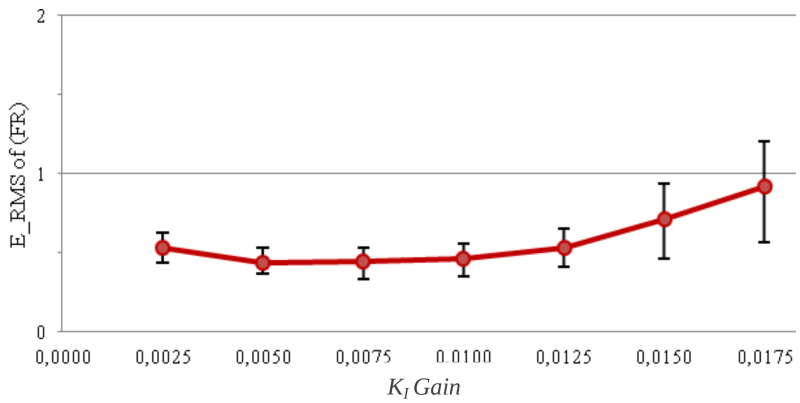


Figure 5. E_RMS of radial forces (F_R) with various integral (K_I) gains applied

Conclusion

This paper describes implementation of real-time force control system for the Staubli TX60 robot. It also outlines the real-time Linux operating system, transmit control protocol and internet protocol (TCP/IP) communication, and multi-tasking software designed for the robot. Outputs from the external force control system were transmitted as incremental displacements transferred to the robot CS8C controller using TCP/IP protocol to modify the robot's trajectory in real-time. The proposed HRI system has been evaluated and criteria used for evaluation of the real-time force sensor and real-time control path of the robot systems have been also discussed. In particular, proportional plus integral (PI) control was applied to robot's velocity and force control algorithms, and the gains (K_P and K_I) have been experimentally tuned based on a virtual crank-turning test. Based on the obtained results, it was confirmed that the best performance (as specified by the E_RMS of radial force) has been observed with proportional and integral gains of $K_P = 0.10$ and $K_I = 0.005$ respectively.

References

[1] Dong Il, P., *et al.*, Design and Analysis of Direct Teaching Robot for Human-Robot Cooperation, *Proceedings,* Assembly and Manufacturing, ISAM IEEE International Symposium, 2009, pp. 220-224

[2] Fong, T., *et al*., Collaboration, Dialogue, Human-Robot Interaction, *Robotics Research, 6* (2003), 5, pp. 255-266

[3] Siciliano, B., *et al.*, *Force Control*, *Springer Handbook of Robotics.* Springer Berlin Heidelberg, 2008, pp. 161-185

[4] Mason, M. T., Compliance and Force Control for Computer Controlled Manipulators, *Systems, Man and Cybernetics, IEEE Transactions on Systems, 11* (1981), 6, pp. 418-432

[5] Vukobratović, M., *et al., Dynamics and Robust Control of Robot-Environment Interaction,* World Scientific Publishing Co., Inc., Singapore, 2009

[6] De Schutter, J., Van Brussel, H., Compliant Robot Motion II. A Control Approach Based on External Control Loops, *Int. J. Rob. Res., 7* (1988), 4, pp. 18-33

[7] Pierrot, F., *et al*., Hippocrate: a Safe Robot Arm for Medical Applications with Force Feedback, *Med Image Analysis, 3* (1999), 3, pp. 285-300

[8] Volpe, R., Khosla, P., A Theoretical and Experimental Investigation of Explicit Force Control Strategies for Manipulators, *Automatic Control, IEEE Transactions on System, 38* (1993), 11, pp. 1634-1650

[9] Zeng, G., Hemami, A., An Overview of Robot Force Control, *Robotica, 15* (1997), 05, pp. 473-482

[10] Stanković, J. A., Misconceptions About Real-time Computing: a Serious Problem for Next-generation Systems, *Computer, 21* (1988), 10, pp. 10-19

[11] Kilgus, C. C,. Gore, W. C., Root-Mean-Square Error in Encoded Digital Telemetry, *Communications, IEEE Transactions on Systems*, 20, (1986), 3, pp. 315-320
[12] Bruckner, H. P., *et al*., Wearable and Implantable Body Sensor Networks (BSN), *Proceedings, 2012 International Conference*, London, UK, 2012
[13] Cooley, J. W, Tukey, J. W., An Algorithm for the Machine Calculation of Complex Fourier Series, *Mathematics of Computation, 19* (1965), 90, pp. 297-301

Paper submitted: October 5, 2015
Paper revised: November 19, 2015
Paper accepted: January 10, 2016