The PC based parallel robot control

EtherCAT and real time based robot control

Sun Lim
Intelligent Robot Research Center(IRRC)
Korea Electronic Technology Institute(KETI)
Bucheon-si, South of Korea
Sunishot@keti.re.kr

Il-Kyun Jung
Intelligent Robot Research Center(IRRC)
Korea Electronic Technology Institute(KETI)
Bucheon-si, South of Korea
mickey3D@keti.re.kr

***Abstract*— In this paper, we describe the control strategy of high speed parallel robot system with EtherCAT network. This work deals the parallel robot system with centralized control on the real-time operating system such as window TwinCAT3. Most control scheme and algorithm is implemented master platform on the PC, the input and output interface is ported on the slave side. The data is transferred by maximum 20usecond with 1000byte. EtherCAT is very high speed and stable industrial network. The control strategy with EtherCAT is very useful and robust on Ethernet network environment. The developed parallel robot is controlled pre-design nonlinear controller for 6G/0.43 cycle time of pick and place motion tracking. The experiment shows the good design and validation of the controller.**

Keywords—parallel robot control, etherCAT, nonlinear control, parallel robot inverse kinematic

I. INTRODUCTION (*Heading 1*)

The Parallel Robot as Shown in Fig 1(a)(b) is one of the successful industrial robot in many application domains, pick-and-place, packing even assembling. Because of this, Parallel Robot have many advantages compared to traditional serial robots, such as speed and accuracy. based on the end-effect of parallel robot can move a light weight objects at a speed of several meter per second, and has high accuracy within a limited range as shown in Fig 1(c).[1]

The developed parallel robot is able to move the three degree of freedom and can reach accelerations near 6G, 0.43 cycle time. In order to achieve such accelerations and perform an accurate motion movement such as pick and place, a nonlinear controller must be used. By using conventional controller, i.e. PD linear control, the tracking performance cannot be guaranteed and be limited, especially when the robot has highly nonlinear dynamics and/or when the velocity/acceleration are high. Also the robot pick the unknown load and drop the load on the belt, so the sudden unknown dynamic is occurred on normal operation. These are very strong condition that the controller is not easy to design for specification of parallel robot.

The most important advantage of such parallel robot over serial chain robot is certainly the possibility to keep all motor fixed to a base, with the consequence that the moved mass is much smaller and fast movement can be performed and guaranteed. Thus parallel robot can also achieve a higher stiffness and are well suited for the application pick and place and high speed milling machine.

In modern motion system, the real-time industrial ethernet offers many advantages over the analog links and flexible system configuration. The individual axis servo control architecture is need to synchronized and has strict exact execute time. Whenever added and mounted additional axis, the system platform only need to change the software architecture. Thus the designer is easy to change or implement the motion control algorithm and data exchange interface. [2]

networked control platform is introduced into the motion control field and builds networked control system. A networked control system is a control system wherein the control loops are closed through a real-time network[3]. It can eliminate unnecessary wiring thus, reducing the complexity and the overall cost in designing and implementing the control system. EtherCAT technology is mainly being promoted by Beckhoff and they provide several kind of solutions for EtherCAT master and slave, including ASIC chips and software driver. For example, Beckhoff launced ET1100 and ET1200 chip for developing EtherCAT Slave Moudle device, EtherCAT master sample code and library for developing EtherCAT master device easier.

The software architecture is set up considering strict timing constraints on their behavior. EtherCAT is a promising real-time Ethernet which is standardized in IEC 61158 and 61784[3].

The rest of paper is organized as follows: In section 2, we derive the EtherCAT based control algorithm for parallel robot. The adaptive nonlinear control technique is then proposed to improve transient tracking response time and fast motion in Section 3. Section 4 present real-time implementation results on parallel robot to demonstrate the practical and theoretical development of this paper. Finally, section 5 concludes the paper.

Fig. 1. Industrial Parallel robot system

II. ETHERCAT BASED CONTROL ARCHITECTURE

A. Industrial EtherCAT and real-time OS

Most of the robot control systems are being developed using window operating system. Windows has many advantage of easy developing environment, such as, supporting abundant device drivers and multitasking. However it cannot support hard real time capabilities since the windows is not designed for the real-time system originally to emphasize the graphic display and to use the multi task in viewed same time. Moreover it is difficult and takes a lot of time to develop the device drivers in windows operating system.[7][8] Basically a real time O/S is suitable to implement a real time control system. The commercial QNX and VxWorks have the hard real time capabilities. However, the price is too high to be used for education and research purposes. Their support of device drives for custom designed controllers are so weak that the expendability is very low [9]. Thus our system is ported to window real time operating system, such as, TwinCAT3. This OS have an advantage of the similar window OS and hard real time is guaranteed on the very low price.

B. EtherCAT

EtherCAT is Real Time Ethernet technology based on the Ethernet. Its objective is to maximize the utilization of the full duplex Ethernet bandwidth [4]. The network communication system employs the Master/Slave principle. The master node sends the packet frames to the slave nodes. The slave node extracts data from and inserts data in to packet frames. Telegram processing principles are utilized “on the fly”. In this way, by performing EtherCAT can use ninety percent of bandwidth without packet collision [5][6]. Also EtherCAT provide distributed clock function for synchronization with each slave node. It can be improve the control capability of parallel robots.

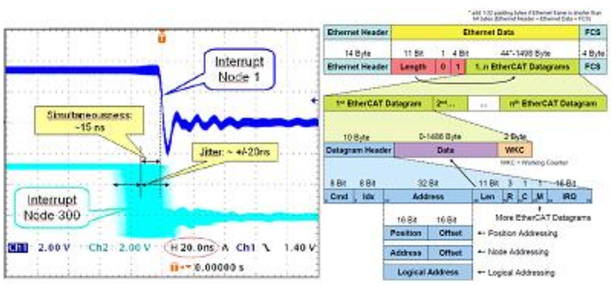

Fig. 2. Accuracy Result of Distributed Clock by EtherCAT (left) and EtherCAT Datagram (right)

C. EtherCAT based control architecture

The Figure 3. illustrate the parallel robot that is one of the industrial robots more used today and is the most successful robots which have been designed. The parallel robot is useful its applications include pick, place and packing products in the food, medical and cosmetic industry, as well as in assembly process of electronic device. The parallel robot is made up of an active joint with joint motor and passive joint which are linked together three independent rigid arms. The actuators are placed for each arm as shown in Fig 3. right side. The upper links and lower links are connected on the ball joints. The moving plate is connected lower link with the ball joint in order to guarantee planar motion along x,y and z axes.

The control scheme is detailed in figure 5. The robot control attempts to achieve motion control using parallel robots under unknown manufacturing tolerances and inaccuracies by migrating the measurement from joint space and tack space. This problem is solved by the on and off calibration of the parallel body. However this is very hard and time consuming work so most field engineer want to avoid this work and automatically find the parameter by program. Thus the control objective of this paper is described as bellows;

Find the control law below condition is satisfied as

$$\lim_{t\to\infty}\begin{bmatrix} x_i^* \\ y_i^* \\ z_i^* \\ t_i^* \end{bmatrix} - \begin{bmatrix} x_i \\ y_i \\ z_i \\ t_i \end{bmatrix} = 0 \longrightarrow \begin{bmatrix} e_i^x \\ e_i^y \\ e_i^z \\ e_i^t \end{bmatrix} = 0 \qquad (1)$$

and

$$\tau(t) \le \alpha < \infty$$

where * upper script mean the command and lower script i mean the i-index parallel robot platform. $\begin{bmatrix} x_i & y_i & z_i \end{bmatrix}$ are a Cartesian space axis, individually x-axis , y-axis, z-axis and time domain. The task space control is very difficult because feedback of the end-effect on the cartesian space is not useful measure on real-time. Thus most control scheme is designed on the joint space of parallel robot. This control block diagram is represented on Figure 4. This work attempts to achieve precise motion control using parallel robot that encounters manufacturing tolerances, assembly errors and thermal deformations without periodic calibration through only joint space measurements.

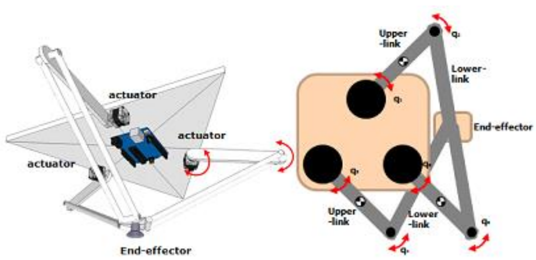

Fig. 3. The Parts of parallel robot (left) and configuration (right)

Master Controller is made up of industrial PC platform and EtherCAT Master Stack. The system is implemented window OS such as window 7-32bit, with window TwinCAT3 . The parallel robot control scheme and safety mechanism is running on TwinCAT3 platform for real-time and periodic computing work. The monitoring and user interface is implemented window side. The data is shared between window and TwinCAT3 with inner interface socket communication. The controller's application program defines two data module structure: data modules of the control instruction and state feedback.

III. PARALLEL ROBOT CONTROL

The parallel robot path planning block is the generation block of motion on the cartesian space trajectory or regulation. This block make the x,y,z and time trajectory that is input of the inverse-kinematics block. The inverse-kinematic block generate the joint space motor position trajectories.

By using the natural orthogonal complement method, the dynamic model of parallel robot can be derived in joint space

$$M(q)\ddot{q} + b(q,\dot{q}) + g(q) = \tau(t) \tag{2}$$

where $M(q) \in R^{3\times3}$ is the parallel robot positive definite inertia matrix, $b(q,\dot{q}) \in R^{3\times3}$ is a vector of coriolis and centripetal forces, $g(q) \in R^{3\times1}$ is the gravity term while is the generalized torque vector acting on the generalized coordinate vector.

$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \in R^{3\times1}$ is the vector of parallel robot joint space position, i.e., encoder value

$\dot{q} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \in R^{3\times1}$ is the vector of velocity of parallel robot joint space

$\ddot{q} = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} \in R^{3\times1}$ is the vector of accelerations of parallel robot space.

Most practical system have the physical measurement uncertainty so taking the dynamical inaccuracies into consideration, one can write the following relations

$$M(q) = M_n(q) + \Delta M(q)$$

$$g(q) = g_n(q) + \Delta g(q),\ b(q,\dot{q}) = b_n(q,\dot{q}) + \Delta b(q,\dot{q}) \tag{3}$$

where $M_n(q), b_n(q,\dot{q}), g_n(q)$ are respectively the nominal value inertia matrix, nominal vector of coriolis and centripetal forces and the nominal gravity term. These value are measured by the calibration tool or 3D cad software. Δ symbol stands for the deviation between these terms and the actual ones.

Rewriting (2)

$$M_n(q) + b_n(q,\dot{q}) + g_n(q) + \Delta M(q) + \Delta b(q,\dot{q}) + \Delta g(q) = \tau(t)$$

$$M_n(q) + b_n(q,\dot{q}) + g_n(q) + d(t) = \tau(t) \tag{4}$$

where $d(t) = \Delta M(q) + \Delta b(q,\dot{q}) + \Delta g(q)$ is disturbance term that is canceled by the disturbance observer which is described as follows

The dynamic uncertainties such as the un-modeled dynamics are considered as disturbances, then a disturbance observer is designed to estimate such disturbances and used to generate compensation term.

$$\hat{d}(t) = \tau(t) - \tau^*(t) \tag{5}$$

where $\hat{d}(t)$ is disturbance estimation value and $\tau^*(t) = M_n(q) + b_n(q,\dot{q}) + g_n(q)$ is a calculation value with nominal value.

Moreover, it can be estimated through the following low pass filter for avoiding noise sensitivity,

$$\hat{d}(t) = \frac{h}{s+h}\left[\tau(t) - \tau^*(t)\right] \tag{6}$$

where h is the single observer gain which controls how fast the estimated signal converges to the actual disturbance.

Then, the control law is represented as follow

$$\tau(t) = \frac{1}{M_n}\left[b_n(q,\dot{q}) + g_n(q) - ke(t) + \hat{d}(t)\right] \tag{7}$$

where k is the control gain for reducing convergence time and regulation error.

The stability is proved by the lyapunov function that is omitted in this work.

The dynamic model (2) has the following properties that will be used in controller designing[]

Property 1: The matrix $M(q)$ is a symmetric and positive-definite matrix, which satisfies $\|M(q)\| \le a$ and $\lambda_{\max}(M(q)) \le b$ for some constants $a, b > 0$, where $\lambda_{\max}(\ )$ represent maximum eigenvalue of the matrix.

Property 2: The matrix $b(q,\dot{q})$ satisfies $\|b(q,\dot{q})\| \le c$ for some constant $c > 0$

Property 3: The vector $g(q)$ satisfies $\|g(q)\| \le d$ for some constant $d > 0$

Property 4: The matrix $\dot{M}(q) - 2b(q,\dot{q})$ is skew-symmetric matrix.

IV. Experimental Result

Let us now show the design and implementation process of the proposed design on real parallel robot systems. For this purpose, we use two pick and place motion trajectory tracking.

Figure 4 show the developed parallel robot delta type. The robot system is controlled by EtherCAT master and slave interface module. The robot employs direct drive motors by Mistubuish Ltd. The motor is equipped with encoder to detect the rotation angle. The maximum torque, rated speed, and weight of the motor are 14 Nm, 3.75rps and 12kg, respectively.

The moving plate of the parallel robot is moved along the specified trajectory to and from both of the ends. The round-trip cycle time is set 0.43 sec. A short wait(20msec) is inserted at both ends of the trajectory in order to test a pick and place operation. It is assumed that the external forces and moment are not applied to the moving plate during the motion, and the viscous frictions are small enough to be ignored. Figure 5 illustrates a rough scheme of the pick and place motion. The dotted line is represents the trajectory along which the endpoint of the robot should track. There are no specification on the corner radius, travel distance L and height h of the trajectory. In the evaluation experiment, the travel distance and height are specified as L = 0.27m and h =0.025m, respectively.

Figure 5 and 6 show the system responses, control effort and error signal with reference trajectory. It can be seen that while the system remain stable for both uncertainty levels, tracking performance deteriorates and larger disturbances.

Figure 6 shows the tracking error using adaptive nonlinear scheme. It is seen that the initial transient error and the transients at the parameter changes are quite large. If the adaptive parameter is large, the control effort also is large. Thus the designer chooses the control parameter and adaptation parameter for considering under the control effort limit and transient response time specification.

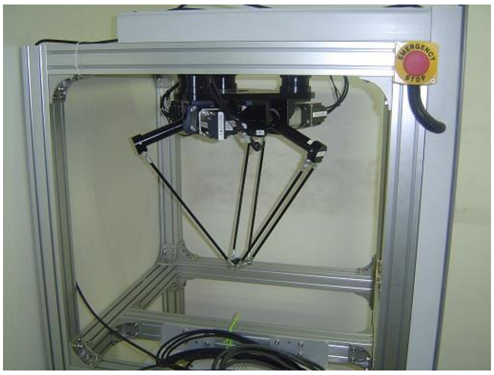

Fig. 4. The individual joint trajectory of parallel robot system

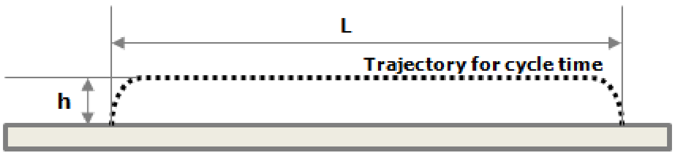

Fig. 5. The specification of the adept pick and place motion

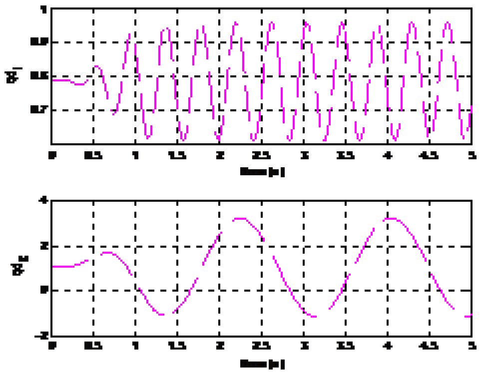

Fig. 6. The individual joint trajectory of parallel robot system

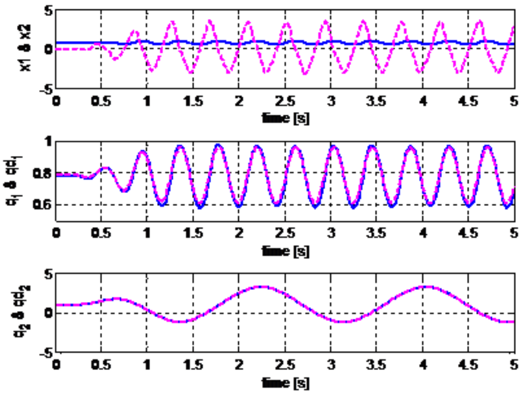

Fig. 7. The trajectory following response and error

V. Conclusion

This paper proposed the control scheme of the parallel robot system. For the high speed and precision motion control, we equipped the widow real time operating system, e.x. window TwinCAT3, and 100Mbps industrial communication network such as EtherCAT. The comparison between the nomal PC-based control system and real time based control system with EtherCAT results will be obtained on the same parallel robot hardware. Also we will prove the feasibility and performance of the proposed control frame work without any change of the parallel robot hardware. This is very important on the parallel robot motion real time tracking, e.g. fast cycle. Three servo control algorithm is implemented on the EtherCAT mater PC with real time-OS. The porting PD and PID controller with velocity feedback ,forward term and dynamic compensation term. The gain of these controller is fixed and obtained by Off-line simulation. Thus after running many pick and place condition, the error remains some boundary and tracking result is not satisfactory. Therefore the adaptation or on-line gain tuning scheme is necessary for small error bound and quickness of response of all condition or parallel robot system.

ACKNOWLEDGMENT

This work was supported by the Industrial Strategic technology development program, 10041965, Development of a parallel link robot with 6kg payload and 0.43 sec cycling time, funded by the Ministry of Knowledge Economy (MKE), Korea.

REFERENCES

[1] Yangmin Li. “Design and application of a new 3-DOF translational parallel manipulator”, Advanced Robotics and Its Social Impacts, 2007. ARSO 2007. IEEE Workshop on, 9-11 Dec., pp. 1-6(2007)

[2] Khalil, I.S.M.; Golubovic, E.; Sabanovic, A.; ,”High precision motion control of parallel robots with imperfections and manufacturing tolerances,” Emerging Technologies and Factory Automation (ETFA), 2010 IEEE conference on , 13-16 Sept.,vol., no., pp. 1-7(2010)

[3] J.D. Decotignie, "Ethernet-based real-time and industrial communications," Proceedings of IEEE, vol. 93, pp 1102-1117, 2005

[4] Il-Seuk Song, “Implementation and analysis of the embedded master for EtherCAT”, Control Automation and Systems (ICCAS), 2010 International Conference on, 27-30 Oct., pp. 2418–2422.(2010)

[5] Rostan, M. “EtherCAT enabled advanced control architecture”, Advanced Semiconductor Manufacturing Conference (ASMC), 2010 IEEE/SEMI, 11-13 July, pp. 39-44(2010)

[6] Prytz, G. “A performance analysis of EtherCAT and PROFINET IRT”, Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on, 15-18 Sept., pp. 408-415(2008)

[7] E. Tapia, S. Intille and K. Larson, " EUROM ICR0’91 WORKSHOP ON REAL-TIME SYSTEMS," UNIVERSITE PARIS XI PAR I S-0 RSAY, FRANCE, 12-14, June(1991)

[8] C. H. Lee and C. Mavroidis, "PC based control ofand mechatronic systems under MS-windows NT workstation," IEEE/ASME Trans. on Mechatronics,vol.6, no. 3, pp. 311-321,(2001)

[9] W. F. Lages and E. M. Hemerly, "Linux as a software platform for mobile robots," submitted to IEEE Transactions on Software Engineering(200)