5. Wireless Communication Technology for Modular Mechatronic Controllers

Glen Bright[†], Nkgatho S. Tlale[§] and Christopher M. Kumile[‡]

[†] University of KwaZulu-Natal, [§] Council for Scientific and Industrial Research, [‡] Tshwane University of Technology

5.1 Introduction

Modular mechatronics systems comprise of mechatronics sub-systems that are functionally complete and can be independently operated. These sub-systems can be readily fitted and connected to, or in combination with, additional sub-systems]1]. Advantages of modular Mechatronics design of systems include flexibility to changing environmental needs, low-cost due to standardization of the sub-systems, inter-connect ability and inter-changeability of sub-systems with each other, and improved system reliability due to modular architecture [2].

Communication technology allows for the systematic integration of mechatronics sub-systems in order to achieve complex integrated mechatronics system which would be difficult to achieve without the use of information technology (IT). Communication technology allows for the remote, real-time control of mechatronics systems e.g. the use of communication to send control signals from haptic device and feedback signals from robotic manipulator in robotics-aided surgery where the surgeon and patient are in different locations (e.g. countries).

In Figure 5.1, three different levels of communications are illustrated. The choice of communication technology to be used is dependant on the complexity, size and response time requirements of the tasks to be controlled. It is important to match the time delay of the communication system with the time constant of the system/process being controlled. If time delays of the communication system are not an issue, high level protocols can be used to control low level system components/sub-systems. Generally, high level system tasks can be adequately controlled by use of high level communication protocol (e.g. Ethernet), while low level system tasks are adequately controlled with low level communication protocol (e.g. CanBus).

Communication technology in mechatronics systems also allows the achievement of distributed control. However, this results in communication lags due to the distributed architecture. This further introduces problems that are concerned with timing, such as lag effect of zero-order hold (ZOH) and problems with respect to motion control. Problems of time variations can be addressed in control design, e.g., by using robust control so that deviations from nominal timing can be tolerated [3]. This chapter focuses on more on the modelling of such systems, before the controller is implemented i.e. determination of the dynamics of mechatronics systems in order to determine the correct controller reaction time.

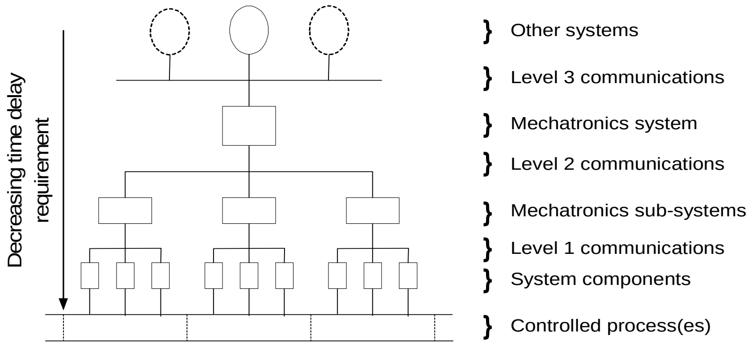


Fig. 5.1 Classification of different of mechatronic and communications levels

5.2 Modular Mechatronics Controllers

Mechatronics controllers are generally embedded controllers that use some model or heuristic rules for the underlying system in order to achieve the optimal control of the systems by monitoring sensor inputs and adjusting actuator outputs. The physical and software components of a mechatronics controller are; signal conditioning hardware unit, signal processing software unit, central processing unit (CPU), power electronics unit, communication circuits and communication software unit.

Figure 5.2 shows a multiple-input multiple-output (MIMO) mechatronics system with sensors, controller and actuators controlling a system/process. The system can be described by its state variables $\mathbf{x}(t)$. The actuators outputs, $\mathbf{u}(t)$ are the inputs of the system, while the outputs of the system, $\mathbf{y}(t)$ are the inputs to sensors. A single-input single-output (SISO) system is comprised only of one sensor and actuator respectively.

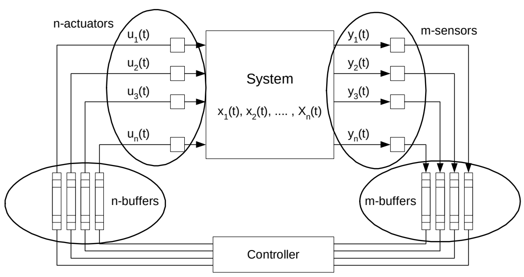


Fig. 5.2 Mechatronic controller on a process level

The use of communication technology requires communication nodes at sensors, controllers and actuators nodes. The communication network is used to transmit the information between sensors, controllers and actuators. The transmission of information over a communication network takes some time, which leads to communication lags. In many cases, the communication lags are varying in a random fashion.

Communication lags depend on the configuration of the network and the scheduling policy used. Factors like the network load, priorities of the other ongoing communications, and electrical disturbances affect the communication lags [5]. Communication lag at time *k* can be sub-divided into three sections:

- Communication lags between the sensor and the controller, τ_k^{sc},
- Computational lag in the controller, τ_k^{c}, and
- Communication lag between the controller and the actuator, τ_k^{ca}

The control delay, τ_k, can then be defined as the time from when a sensor receives a signal to when it is used in the actuator as the control signal. Thus:

$$\tau_k = \tau_k^{sc} + \tau_k^{c} + \tau_k^{ca} \quad 5.1$$

The Nyquist theorem states that the controller reaction time must be at least half of the smallest time constant of the system to ensure proper system control [5].

Depending on how the sensors, actuators, and controller nodes are synchronised, several setups can be considered. Event-triggered controllers send information as soon as it is available to the nodes (i.e. sensors and controllers). The information is triggered by some programmed activity (e.g. the signal being monitored achieving some specified level). Time-triggered systems transmit the information based on some time model using the clock of the system. The node is able to start its activity at a pre-defined time (e.g. node’s activities can be periodic). By implementing communication technology, a distributed MIMO system can be achieved. Instead of centralizing intelligence in one controller, intelligence can be distributed on different sensors and actuators in the system.

Control and processing of data can only be done when data is available at the node. Vacant sampling occurs when data does not arrive at the communication node on time. To limit the level of vacant sampling, the buffers must be longer than the worst case communication lag [6]. Buffers can be used between the sensors, controllers, and actuators. This minimizes the variation of the communication lags between the communicating nodes and produce constant communication lags. The disadvantage of using buffers is that the control delay can become longer than necessary. This can lead to decreased controller performance.

5.3 Communications Technology

Communications technology is used in mechatronic systems to achieve distributed, real-time control systems. The choice of the communication technology implemented in a mechatronic system is dependent on the following:

- The spatial distances between the units of the system
- The amount of information to be transferred via the communication technology (or bandwidth)
- The response time that is required from the communication technology

For long distances (>10 meters, level 3 communication), communication protocols such as Transmission Control Protocol/Internet Protocol (TCP/IP), ProfiBus, DeviceNet, FieldBus, Modbus, etc are used. These technologies are normally used to connect one mechatronic system to a network of other mechatronic systems. For medium distances (<10 meters, > 1 meter, level 2 communication) communication protocols such as Controller Area Network (CAN), RS232, RS485, RS422, GBIP, etc are used. These protocols are also used in instrumentation devices.

For short distances (<1 meters, level 1 communication) communication protocols such as CAN, Inter-Integrated Circuit (I^2C), Serial Peripheral Interface (SPI), embedded TCP/IP, etc. are used. These protocols are normally used to connect one microprocessor to others.

5.4 Model Based Mechatronic Controllers

The difference between a model-based controller and a knowledge-based controller is that the latter uses rules or heuristics, while the former uses some model of the system in order to achieve control of the system.

For a SISO mechatronic system, let x(t) (= [$x(t_1)$, $x(t_2)$,…, $x(t_n)$]) be a state variable that can be used to fully describe a continuous-time varying system defined by, f(t). This infinite dimensional system can be described by the equation 5.2:

$$f(t) = g(x(t)) \quad 5.2$$

Let the controlled process be defined by:

$$\dot{x}(t) = Ax(t) + Bu(t) + v(t) \quad 5.3$$

where A is the state constant
B is the input constant
u(t) describes the input

and

v(t) is white noise with zero mean and has covariance R_v.

The control loop of this mechatronic system can be formulated by sampling. Assuming that the sampling period, *h*, is constant and greater than the delay from the sensor to the actuator (i.e. $h > \tau_k^{sc} + \tau_k^{ca}$). Integrating equation 5.3 over a sampling interval results in equation 5.4.

$$x_{k+1} = \Phi x_k + \Gamma_0\left(\tau_k^{sc}, \tau_k^{ca}\right)u_k + \Gamma_1\left(\tau_k^{sc}, \tau_k^{ca}\right)u_{k-1} + v_k \qquad 5.4$$

where $x_k = x(kh)$

$$\Phi = e^{Ah} \qquad 5.5$$

$$\Gamma_0\left(\tau_k^{sc}, \tau_k^{ca}\right) = B\int_0^{h-\tau_k^{sc}-\tau_k^{ca}} e^{As}ds \qquad 5.6$$

$$\Gamma_1\left(\tau_k^{sc}, \tau_k^{ca}\right) = B\int_{h-\tau_k^{sc}-\tau_k^{ca}}^{h} e^{As}ds \qquad 5.7$$

and a variance of

$$-v_k = R_1 = E\left[v_k^2\right] = R_v\int_0^h \left(e^{A(h-s)}\right)^2 ds \qquad 5.8$$

This result is the same as those found in [7] and [8]. These are standard results for the sampling of systems with time-delays, where the infinite-dimensional continuous-time system is reformulated to the time-varying, finite-dimensional, discrete-time system. The output equation is

$$y_k = Cx_k + w_k \qquad 5.9$$

where C is the output constant and w_k is a random process of white noise which is uncorrelated to v_k.. The mean of w_k is zero and its co-variance is R_2.

A linear controller for this system can be written as

$$x_{k+1}^c = \Phi^c(\tau_k)x_k^c + \Gamma^c(\tau_k)y_k \qquad 5.10$$

$$u_k = C^c(\tau_k)x_k^c + D^c(\tau_k)y_k \quad 5.11$$

where τ_k in Φ^c, Γ^c, C^c and D^c means that the controller knows the network delays completely or partly i.e. $\{\tau_0,\ldots,\tau_k, h_o,\ldots, h_{k-1}\}$ are known to the controller before u_k is calculated. This can be achieved by time-stamping of network messages and time synchronisation of the communicating nodes.

Substituting u_k in x_{k+1} and y_k in u_k and re-arranging, the closed-loop system can be written as

$$z_{k+1} = \Phi(\tau_k)z_k + \Gamma(\tau_k)e_k \quad 5.12$$

where

$$z_k = \begin{bmatrix} x_k \\ x_k^c \\ u_{k-1} \end{bmatrix} \quad 5.13$$

$$\Phi(\tau_k) = \begin{bmatrix} \Phi + \Gamma_0(\tau_k)D^c(\tau_k)C & \Gamma_0(\tau_k)C^c(\tau_k) & \Gamma_1(\tau_k) \\ \Gamma^c(\tau_k)C & \Phi^c(\tau_k) & 0 \\ D^c(\tau_k)C & C^c(\tau_k) & 0 \end{bmatrix} \quad 5.14$$

$$e_k = \begin{bmatrix} v_k \\ w_k \end{bmatrix} \quad 5.15$$

and the variance R of e_k is

$$R = E(e_k e_k^T) = \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix} \quad 5.16$$

The form of $\Phi(\tau_k)$ and $\Gamma(\tau_k)$ is determined by the process, the communication network, and the controller structure. τ_k is a random process uncorrelated with $\{e_k\}$ and can be a vector consisting of the delay from sensor to controller, τ_k^{sc}, as well as the delay from controller to actuator, τ_k^{ca}. It is assumed that τ_k has a known distribution pattern, and that τ_k is independent from different k.

If the sampling period, h, is not constant, this results in sampling interval jitter.

Equation 5.4 then becomes:

$$x_{k+1} = \Phi x_k + \Gamma_0(h_k, \tau_k^{sc}, \tau_k^{ca})u_k + \Gamma_1(h_k, \tau_k^{sc}, \tau_k^{ca})u_{k-1} + \Gamma_v(h_k)v_k \quad 5.17$$

For a MIMO system with *m* sensors and *n* actuators, the system equations are:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{v}(t) \qquad 5.18$$

where **A** and **B** are now matrices.

The longest sensor to controller delay is defined as:

$$\bar{\tau}_k^{sc} = \max\left(\tau_k^{sc1}, \tau_k^{sc2}, ..., \tau_k^{scm}\right) \qquad 5.19$$

and assuming that the old time delays are known to the controller and that that sampling period is greater than the delay from the sensor to actuator; i.e.

$$\max\left(\tau_k^{sc1}, \tau_k^{sc2}, ..., \tau_k^{scm}\right) + \max\left(\tau_k^{ca1}, \tau_k^{ca2}, ..., \tau_k^{san}\right) < h \qquad 5.20$$

Equation 5.16 can be sampled into:

$$\mathbf{x}_{k+1} = \mathbf{\Phi}\mathbf{x}_k + \mathbf{\Gamma_0}\left(\bar{\tau}_k^{sc}, \tau_k^{ca1}, ..., \tau_k^{can}\right)\mathbf{u}_k + \mathbf{\Gamma_1}\left(\bar{\tau}_k^{sc}, \tau_k^{ca1}, ..., \tau_k^{can}\right)\mathbf{u}_{k-1} + \mathbf{v}_k \qquad 5.21$$

where $x_k = x(kh)$,

$$\mathbf{\Phi} = e^{\mathbf{A}h} \qquad 5.22$$

$$\mathbf{\Gamma_0}\left(\bar{\tau}_k^{sc}, \tau_k^{ca1}, ..., \tau_k^{can}\right) = \left[\Gamma_0^1\left(\bar{\tau}_k^{sc}, \tau_k^{ca1}\right) \quad ... \quad \Gamma_0^n\left(\bar{\tau}_k^{sc}, \tau_k^{can}\right)\right] \qquad 5.23$$

$$\mathbf{\Gamma_1}\left(\bar{\tau}_k^{sc}, \tau_k^{ca1}, ..., \tau_k^{can}\right) = \left[\Gamma_1^1\left(\bar{\tau}_k^{sc}, \tau_k^{ca1}\right) \quad ... \quad \Gamma_1^n\left(\bar{\tau}_k^{sc}, \tau_k^{can}\right)\right] \qquad 5.24$$

$$\mathbf{\Gamma_0^i}\left(\bar{\tau}_k^{sc}, \tau_k^{cai}\right) = B \int_0^{h-\bar{\tau}_k^{sc}-\tau_k^{cai}} e^{As} ds \qquad 5.25$$

$$\Gamma_1^i\left(\bar{\tau}_k^{sc}, \tau_k^{cai}\right) = B \int_{h-\bar{\tau}_k^{sc}-\tau_k^{ca}}^{h} e^{As} ds \qquad 5.26$$

and the variance of state noise

$$\nu_{\mathbf{k}} = \mathbf{R}_1 = \mathrm{E}\left[\mathbf{v}_k \mathbf{v}_k^T\right] = \int_0^h e^{\mathbf{A}(h-s)} R_v \left(e^{\mathbf{A}(h-s)}\right)^T ds \qquad 5.27$$

The output equation is then:

$$\mathbf{y}_k = \mathbf{C}x_k + \mathbf{w}_k \qquad 5.28$$

5.4Wireless Mechatronics Controller for the Camera Platform

In this project, the aim was to replace the tether connection between a camera platform and the operator platform, see Fig. 5.3) used in the film industry. The operator platform consisted of two hand-wheels (with their respective encoders) that controlled the roll and the yaw orientations. The joystick was then used to control the pitch orientation. The hand-wheels had adjustable viscous dampers to maintain the feel of a typical hand-wheel for conventional camera platforms. The hand-wheels, joystick and associated electronics were mounted on a separate box (operator side) from the camera platform (camera platform side). The hand-wheels and the joystick were used to determine the direction of turn, positional control, speed and acceleration [9].

After acquiring the signal from the sensors, the servo-controller implemented a PID routine to control the position, velocity and acceleration of the camera platform motors within the required response time. The developed wireless communication system was required to send the position, speed, acceleration and time-stamp information to the camera platform's servomotor controller. Figure 5.4 shows the data flow through the mechatronic control system.

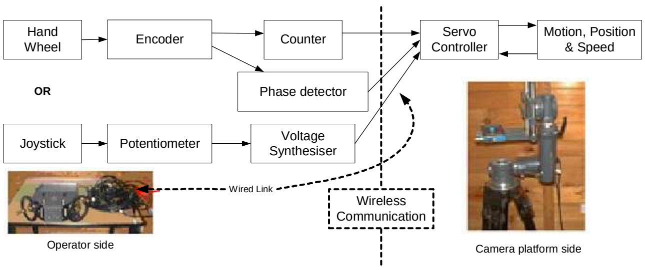


Fig. 5.3 A camera platform with tether connection

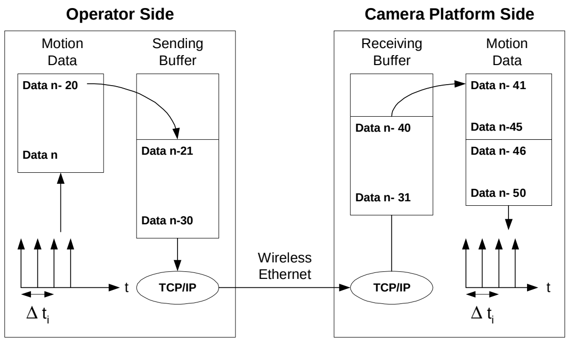


Fig. 5.4. Flow of data through the wireless mechatronic controller for camera platform

5.4.1 Requirements on the Wireless Mechatronics Controller

Wireless communication was required to transmit the control commands only, without the vision information from the camera. The average sight reaction time of trained camera operators of t_o = 0.2466 seconds, was used as a benchmark for the minimum response time of the integrated system [10]. The developed hard-wired system was determined to have satisfactory lag/response time of t_{sh} = 0.062 seconds (i.e. $t_o >> t_{sh}$). For the developed wireless system, t_o must be less than 0.20 seconds (using a safety factor of 20%) in order to achieve satisfactory control of the camera platform.

The total number of bytes per second required to be handled by the wireless communication system, in order to effectively control the camera platform was determined to be 20.00156 Kbytes per $1/60^{th}$ of a second (i.e. sampling rate of 60 Hz). This included 25 percent more bytes for address, time-stamping and error-checking the data.

Wireless Ethernet implemented at a carrier frequency of 2.4 GHz (or 300 000 Kbytes/sec) and IEEE 802.11b standard was used on two single board computers (the sender and the receiver) was implemented as a solution. Analysis of network utilization results in [11]:

$$U = \sum_i \frac{C_i}{T_i} = 4.00098 \times 10^{-3} \qquad 5.29$$

where: i is the number of periodic transmissions on the bus,

C_i is the transfer time for this message, and

T_i is the period for sending of message (i).

The utilisation is a measure of how much load there is on the bus. A utilisation of 4.00098 x 10^{-3} indicates the bus load is very low. If the utilisation is greater than 1 the bus has overload. However, in practice if T_i increases there is a high network utilization being experienced.

5.5 Modelling of the Camera Platform

In order for the camera operator to control the motion of the camera intuitively, a controller that could combine and resolve the various motions of the camera joint motors into separately controllable hand motions along the world coordinates axes was required. Such a control scheme is termed resolved motion control. This means that the several joint motors would run simultaneous at different time-varying rates in order to achieve desired coordinated hand motion along any world coordinate axis. This would enable the camera operator to specify the direction and speed of motion along any arbitrary oriented path for the camera to follow.

A typical camera platform used in the film industry can be described as a three degree of freedom (DOF), spatial revolute manipulator. Using the Denavit-Hartenberg technique, the coordinate axes of the three rotational axis of the camera platform were positioned and the kinematics parameters of the manipulator were determined [12]. The homogenous transformation matrix that describes the orientation and position of the camera was determined as:

$$ {}^{\mathbf{base}}\mathbf{T}_{\mathbf{hand}}(t) = \begin{bmatrix} n_x(t) & s_x(t) & a_x(t) & p_x(t) \\ n_y(t) & s_y(t) & a_y(t) & p_y(t) \\ n_z(t) & s_z(t) & a_z(t) & p_z(t) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{n}(t) & \mathbf{s}(t) & \mathbf{a}(t) & \mathbf{p}(t) \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad 5.30 $$

$$ {}^{0}\mathbf{T}_3 = \begin{bmatrix} C_1C_2C_3 - S_1S_3 & C_1C_2S_3 - S_1C_3 & C_1S_2 & C_1C_2(c+a) - dS_1 \\ S_1C_2C_3 + C_1S_3 & S_1C_2S_3 + C_1S_3 & S_1S_2 & S_1(cC_1 + aS_2) + dC_1 \\ -S_2C_3 & -S_2S_3 & C_1 & -S_2(c+a) \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad 5.31 $$

where **p** is the position vector of the camera,
n, **s**, **a** are the unit vectors along the principal axes of the coordinate frame describing the orientation of the camera i.e. orientation of the hand coordinate system.

The non-linear equation that describes the camera platform is then:

$$ \mathbf{x}(t) = \mathbf{f}(\mathbf{q}(t)) = \mathbf{f}(\theta_1(t), \theta_2(t), \theta_3(t)) \qquad 5.32 $$

where $\mathbf{x}(t)$ are the world coordinates ($x, y, z, \alpha, \beta, \gamma$) and
$\mathbf{q}(t)$ are the generalized coordinates ($\theta_1, \theta_2, \theta_3$).

Taking the first derivative of equation 5.32 then gives:

$$\frac{d\mathbf{x}(t)}{dt} = \dot{\mathbf{x}}(t) = \begin{bmatrix} \mathbf{v}(t) \\ \Omega(t) \end{bmatrix} = \mathbf{N}(\mathbf{q})\dot{\mathbf{q}}(t) \qquad 5.33$$

where $\mathbf{N}(\mathbf{q})$ is the Jacobian matrix with respect to $\mathbf{q}(t)$, i.e. $N_{ij} = \partial f_i/\partial q_i$
The acceleration of the camera platform can also e determined to be:

$$\begin{bmatrix} \dot{\mathbf{v}}(t) \\ \dot{\Omega}(t) \end{bmatrix} = \dot{\mathbf{N}}\left(\mathbf{q}, \dot{\mathbf{q}}(t)\right)\dot{\mathbf{q}}(t) + \mathbf{N}(\mathbf{q})\ddot{\mathbf{q}}(t) \qquad 5.34$$

For redundant manipulators, the inverse $\mathbf{N}^{-1}(\mathbf{q})$ cannot be determined. It can be found by minimising an error criterion formed by adjoining previous equation with a Lagrange multiplier to a cost criterion, C:

$$C = \frac{1}{2}\dot{q}^T A \dot{q} + \lambda^T\left[\dot{x} - N(q)\dot{q}\right] \qquad 5.35$$

where λ is a Lagrange multiplier vector and
A is an *m* x *m* symmetric, positive definite matrix.
After minimizing with respect to λ, the angular velocities of the joints can be determined to be:

$$\dot{\mathbf{q}}(t) = \mathbf{A}^{-1}\mathbf{N}^T(\mathbf{q})\left[\mathbf{N}(\mathbf{q})\mathbf{A}^{-1}\mathbf{N}^T(\mathbf{q})\right]^{-1}\dot{\mathbf{x}}(t) \qquad 5.36$$

Elements of the Jacobian matrix were determined to be:

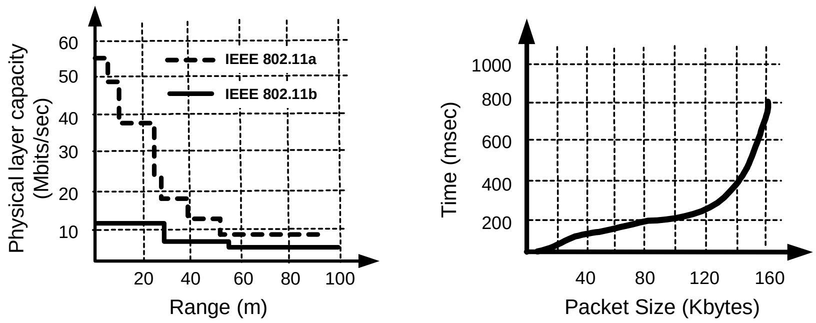
$$\mathbf{N}_1 = \begin{bmatrix} (C_1 + C_2)(c + a) - dS_1 \\ S_1(cC_1 + aS_2) + dC_1 \\ -S_1(c + a) \\ 0 \\ 0 \\ 1 \end{bmatrix}, \mathbf{N}_2 = \begin{bmatrix} (C_1 + C_2)(c + a) - dS_1 \\ S_1(cC_1 + aS_2) + dC_1 \\ -S_1(c + a) \\ S_1 \\ C_1 \\ 0 \end{bmatrix} \text{ and}$$

$$\mathbf{N}_3 = \begin{bmatrix} cS_1S_2^2 + acS_1S_2C_1 \\ cC_1S_2^2 + acC_1C_2S_2 \\ cS_1S_2C_1^2 + cC_1C_2S_1S_2 \\ C_1C_2 \\ S_1S_2 \\ -S_2 \end{bmatrix}$$

5.6 Results

A test involving round-trip times of different data packets over the wireless communication network was taken. Round-trip time is the time a data packet takes to be received by the transmitter from the time of transmission i.e. time from transmitter to receiver and back to transmitter [13]. This test emulates when the time-stamps are used or during error checking of the system. This test indicated that the performance of the wireless communication system was inferior to the one shown in Figure 5.5(a). Figure 5.5(b) then shows the results of this test.

The results in Figure 5.5(b) were obtained under ideal conditions. There was a line of sight between the transmitter and the receiver. As soon as the line of sight was lost, the system became unreliable. A response time of 0.72 seconds for a data packet of 120 Kbytes indicated that the system would not have had a satisfactory performance with data throughput of 160.0125 Kbytes/second as determined in the previous section. In order to improve the system reliability and performance, techniques for reducing the minimum data transfer rate acceptable to optimally control the camera platform had to be used. A mechatronic design of the integrated camera platform was implemented in order to achieve this.



(a) (b)

Fig. 5.5. (a) Physical Layer Capacity of Belkin Wireless Ethernet [14]
(b) Round-times for data packets of different sizes implementing TCP/IP with line of sight .

5.6.1 Performance of the system

The total response time of the PID servo-controller and DC servomotor was determined by exciting the system with a ramp input of about 3000 Kbytes/sec. This was twice the anticipated input signal as determined previously. Figure 5.6 shows the response times of the system and error. The error increased with increasing input signal. The system exhibited a degree of second order response to a ramp input. It can be seen that the response times were in milliseconds. The maximum response in lag occurred at 140 Kbytes/second and was equal to 25 ms. This was satisfactory for the considered application.

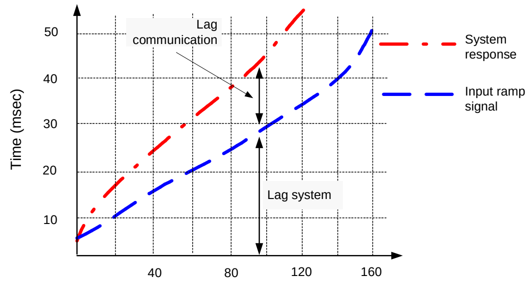


Fig. 5.6 Performance of the wireless mechatronic controller

5.7 Conclusion

The concept of wireless modular mechatronic system comprising of mechatronic sub-systems has been presented. Communication technology has provided a wireless solution to achieve an integrated mechatronic system. Wireless technology has allowed for a remote, real-time control of a mechatronic system. A ~~system,~~ camera platform, was successfully controlled using wireless communication technology. Communication nodes were used to facilitate the transfer of information from the modules/sub-systems. The dynamics of the mechatronic control system was determined in order to determine the correct controller reaction time.

The wireless communication technology in the mechatronic controller provided distributed control. Delays of distributed control systems were minimized through the introduction of communication technology. Problems associated with timing, such as lag effect of ZOH and those with respect to motion control were addressed. Problems of time variations were also addressed in the control design by using robust control so that deviations from nominal timing could be tolerated.

References

1. Virk GS (2003) CLAWAR modularity for robotic systems, Int. Journal of Robotics Research, **22** (3-4); 265-277
2. Bi Z (2002) On adaptive robot systems for manufacturing applications, PhD Thesis, Dept. of Mechanical Engineering, University of Saskatchewan, Saskatoon
3. Chen D-J (2001) Architecture for systematic development of mechatronics software systems', Licentiate Thesis, ISSN 1400 -1179, Mechatronics Lab, Department of Machine Design, Royal Institute of Technology, KTH, Stockholm
4. Ray A (1987) Performance evaluation of medium access control protocols for distributed digital avionics, Trans. ASME Journal of Dynamic Systems, Measurement and Control, **109**; 370–377
5. Nise NS (2008) Control Systems Engineering, 5th Edn, John Wiley & Sons
6. Luck R, Ray A (1990) An observer-based compensator for distributed delays, Automatica, **26** (5); 903–908
7. Åström KJ, Wittenmark B (1997) Computer-controlled systems, 3rd Edn, Prentice Hall
8. Neilson J (1998) Real-time control systems with delays, PhD Thesis, Department of Automatic Control, Lund Institute of Technology
9. Tlale NS, Potgieter J, Bright G (2003) Wireless mechatronics controller for tele-operated camera platforms, Mechatronics and Machine Vision 2003: Future Trends, Ed. Billingsley J, Research Studies Press Ltd
10. Chudler (2008) Test your reaction time, faculty.washington.edu/chudler/java/reacttime.html (visited 13 February 2008)
11. Tindell, K, Hansson H (1995) Real time systems and fixed priority scheduling, Technical Report, Department of Computer Systems, Uppsala University
12 Hartenberg RS, Denavit J (1964) Kinematic synthesis of linkages, McGraw-Hill
13. Comer DE (2000) Internetworking with TCP/IP: principles, protocols, and architectures, Vol 1, 4th Edn, Prentice-Hall.
14. Dornan A (2002), 'Emerging technology: wireless LAN standards, two new wireless LAN standards are cleared for takeoff. Should you climb aboard?', Network Magazine.