Development of an integrated system for force measurement on end effectors of serial manipulators (IRB1410) using LabVIEW and data dashboard.

Vasanth kumar.CH¹, Raghavendra Rao.V² and Ann C V Medona³

¹Department of AI and DS, EIE, Saveetha Engineering College, Chennai, Tamil Nadu, India
²Department of Mechanical Engineering, SRM institute of Science and Technology, Chennai, Tamil Nadu, India
³Department of AI and DS, Saveetha Engineering College, Chennai, Tamil Nadu, India

E-mail: sreevasanthkumarchoda@gmail.com, Raghavev@srmist.edu.in.

Abstract. A data acquisition method to obtain force measurement on an end effector of a serial manipulator is carried out using LabVIEW Data Dashboard to obtain results wirelessly developing an integrated, system between the serial manipulator, a computer and a remote device. A strut array was manufactured with a 4x4 grid and 8 strain gauges were attached on the parallel sides of the 4 rectangular struts based on strain analysis run theoretically using ANSYS to locate position of maximum strain. Two full bridge circuits from the array were then connected to a NI 9219 analog I/O module set up on a NI cDAQ 9174 chassis to measure the force on the struts whenever a load is applied. Tabular results of strain were obtained which varied linearly with uniformly appreciating loads, graphically displayed using LabVIEW GUI. LabVIEW is further connected wirelessly to the NI Data Dashboard app on a mobile via a shared variable engine which generates the results as a shared variable which can then be displayed on other remote devices. Therefore we can say that an integrated wireless system to remote monitor force applied on an end effector of the serial manipulator IRB1410 has been developed successfully.

Keywords- LabVIEW Data Dashboard, Shared Variable, End Effector, Strut array.

1. Introduction.

With further progress in robotics research, scientists are concerned more and more with the field of compliance robotics and wireless and seamless connectivity to other machines and Central Processing Unit (CPU) mainframes to better control robots/serial manipulator and end effectors remotely and accurately to develop systems that are more flexible and less error prone by eliminating redundancies and to improve worker safety. The best way to improve on this is to find a method to wirelessly provide and receive sensor-based intelligence from the robot in real time and to connect it in a closed feedback loop to continuously update and eliminate errors. Remote access of equipment has always been a priority, an example of which can be seen in the paper by Jagadeesh et al. [1] and which is why wireless sensor networks can play a big role in this field. Robot arm systems have always been complex as evidenced in the paper by Chandra Mouli.et.al. [2] Which describes the construction and analysis of a
robotic arm. Remote monitoring and control of such a system has a lot of difficulties. In this case setting up sensor arrays such as Santoso et al. [3] has done in a robotic arm can greatly improve efficiency. The setting up of DAQ and a wireless sensor network for which several previous studies have been carried out [4-11] is an important part of realizing this idea for compliance. Even microcontrollers such as Arduino and Xbee radio modules can be utilized to facilitate wireless sensor networks for remote viewing [5-6], [9-11] which is more cost effective but more error prone as compared to DAQ systems on which several studies have been undertaken [3, 7-8]. The main objective of the project is to run an analysis and receive values wirelessly from an array of sensors present on a robotic gripper with the help of an example metallic strut array to increase compliance methods and to present these results directly on the cellular device of the user instantaneously using web publishing tools or data dashboard and by presenting these results graphically using the GUI interface LabVIEW.

2. Materials and Methods

2.1 Materials and construction

The strut array was constructed using the Aluminium alloy Al H11 which has specifications as shown in table 1.

| Young's Modulus MPa | Poisson’s Ratio | Bulk Modulus MPa | Shear Modulus MPa |
|---------------------|----------------|------------------|-------------------|
| 69000               | 0.33           | 67647            | 25940             |

Strain gauges were affixed on the struts with the following specifications:
1. Gauge Factor: 2:1 ±1%
2. Gauge Resistance: 350 ± 0:3
3. Gauge Length: 3mm

The basic design of the strut array system is made using SolidWorks. It is a 4 x 4 array with each strut having equivalent dimensions of 7 x 5 x 35mm. Each strut is spaced 20mm from each other to form a perfect symmetrical rectangular array. Each strut has two strain gauges attached on parallel sides. The design of a single strut is as shown in Figure 1 (ANSYS). The actual strut system is shown in figure 2. Jumper wires were used for the connection wiring.

Figure 1. Design of the strut in ANSYS
Figure 2. Physical model of strut with mounted sensor array.

2.2 Connecting to DAQ and electronic circuitry:

The sensor array was connected to the analog I/O module NI 9219 which was fixed on the chassis cDAQ 9174 using two full bridge circuits which were connected to two channels on the I/O module. There are 4 channels on the NI 9219 as shown in figure 3. The full bridge circuit connections are as shown in figure 4 with each terminal in a channel connected to a respective connection from the full bridge circuit which comprises 4 strain gauges.

Figure 3. NI 9219 Channels for interfacing analog inputs
The next step is to set up the automatic VI block diagram on LabVIEW to read the values that are inputted from the DAQ module to LabVIEW GUI as shown in figure 5, which is then stored in an excel document and the values can be shown in strain i.e, mm/mm format by choosing the strain module from the DAQ assistant. This block diagram that runs to read the values dumped by the DAQ module should be stored as a read only shared variable in the Shared Variable Engine (SVE) to be deployed in real time to be used in wireless transfer.

2.3 Wireless transfer method using data dashboard:

1. The first step is to open the app NI Data dashboard on the cellular device which has an opening panel as shown in figure 6.
2. Click on the ‘+’ sign in the upper corner to open a new dashboard.
3. Next place an indicator or control such as a waveform chart and numeric indicator in the dashboard by selecting from the indicators and controls palette as shown in figure 7.
4. The next step is to deploy the shared variable in LabVIEW in real time. A trial shared variable program which shows a graph of random numbers is shown in figure 8. The difference between this trial program and the actual deployment is that in the trial program the shared variable is in write mode which means the random number function can change the values stored in the SVE whereas in the original program, the shared variable is in read only mode which only reads the values coming from the DAQ and stores it in SVE and continuously updates it.
5. Now the next step that was taken was to set up an internet hotspot connecting the cellular/remote device and the computer.
6. Next select the data link icon present under the placed indicator in data dashboard which opens the connect dialog box.
7. In the connect dialog box select the shared variables option and enter the computer’s IP address which establishes the connection between LabVIEW and dashboard and connects to the project library.
8. Open the library which has the required shared variable and select it.
9. Finally press the play button on the upper right-hand corner of the panel on the dashboard to view the results remotely on the mobile as shown in figure 9.

![Figure 6. Data Dashboard panel on an android platform](image)

3. Results and Discussions

3.1 Standardized testing with cantilever beam

Before we analyse the strut array we have to first determine if there are any errors in the sensors (strain gauges) we use using a standardized testing base comparing theoretical and experimental data and calculating error percentage. If the error percentage is within 10%, then the strain gauges are good to use and can be used for further testing. Values of the strain value are obtained by a quarter bridge circuit where the strain gauge is placed at the centres of the cantilever which is 181 mm from the support. The weight put at the end of the cantilever is 750 grams or 7.5 N downwards for which the strain values are as shown in table 2.

![Figure 7. (a) Indicators and (b) Controls palette In the android platform](image)
Figure 8. Shared Variable with the data dash board using LabVIEW.

Figure 9. Real time results on dashboard after establishing connection

Table 2. Experimental strain values using cantilever beam as base

| S No. | Strain Values (mm/mm)         |
|-------|-------------------------------|
| 1     | $0.12361 \times 10^{-3}$      |
| 2     | $0.12361 \times 10^{-3}$      |
| 3     | $0.12361 \times 10^{-3}$      |
| 4     | $0.12361 \times 10^{-3}$      |
| 5     | $0.12361 \times 10^{-3}$      |

Average = $0.12361 \times 10^{-3}$ \hspace{1cm} (1)

The theoretical strain at the same point for cantilever beam for the same load of 7.5N comes at $0.134 \times 10^{-3}$ mm/mm.

To calculate the error % see equation 1.

$$
\left[\frac{(0.000134 - 0.000123)}{0.000134}\times 100 \right] = 8.2\% \hspace{1cm} (2)
$$

Since the error percent is less than 10% it is safe for us to proceed with this model to calculate the strain on the given strut.

3.2 Theoretical strain analysis on system of struts using ANSYS

A stress and strain analysis is run on the struts to obtain the optimum location to position the strain gauges on the respective struts. A theoretical strain table with respect to length of strut (table 3) and graph (figure 10) are obtained. It is taken for a load of 10N on each strut on ANSYS. The results
are obtained as shown in figure 11 (ANSYS) after the boundary conditions are set and figure is appropriately meshed in ANSYS.

![Figure 11](image)

**Figure 10.** Length of strut versus strain

![Figure 11](image)

**Figure 11.** Strain distribution on the strut after analysis

**Table 3.** Theoretical strain values

| Sl no | Length[mm] | Value[mm/mm] |
|-------|------------|--------------|
| 1     | 0          | 4.8875 ×10^{-3} |
| 2     | 0.72917    | 4.4589 ×10^{-3} |
| 3     | 1.4583     | 4.2378 ×10^{-3} |
| 4     | 2.1875     | 4.3403 ×10^{-3} |
| 5     | 2.9167     | 4.4221 ×10^{-3} |
| 6     | 3.6458     | 4.4357 ×10^{-3} |
| 7     | 4.375      | 4.4428 ×10^{-3} |
| 8     | 5.1042     | 4.4188 ×10^{-3} |
| 9     | 5.8333     | 4.3865 ×10^{-3} |
| 10    | 6.5625     | 4.3444 ×10^{-3} |

The max strain is experienced at 4.375mm of the strut from the base, hence the strain gauge is placed at this distance from the base of strut.

3.3 Experimental analysis using NI-cDAQ 9174 and NI9219

Using the NI9219, strain values were recorded according to the procedure in experimental setup using a strain module on the DAQ assistant (figure 5) for 5 different weights ranging from 1kg to 5kg thereby getting a range of strain values for each load without too much variation from average value as
shown in table 4. These results were then displayed in graphical format on LabVIEW GUI which was then displayed in real time on Data Dashboard as shown in figure 9.

Table 4. Experimental strain values from DAQ

| LOAD(Kg) | RANGE OF STRAIN VALUES FOR THE GIVEN LOAD AS PER NI-cDAQ9174 |
|----------|--------------------------------------------------------------|
| 1.00     | 0.000336, 0.000338, 0.000342, 0.000338                      |
| 2.00     | 0.000357, 0.000357, 0.000357, 0.000357                      |
| 3.00     | 0.000450, 0.000452, 0.000449, 0.000451                      |
| 4.00     | 0.000542, 0.000543, 0.000544, 0.000544                      |
| 5.00     | 0.000546, 0.000545, 0.000545, 0.000547                      |

The results that have been measured provide a correct estimate on the forces applied on the serial manipulator IRB 1410 by using strain gauge and a DAQ and remotely view and monitor it. This study further enhances our understanding of remote monitoring of robots used in industries, i.e., compliance robotics. A number of sensor arrays in a variety of configurations and types can be used in this setup to completely monitor a working robotic arm/serial manipulator from afar and many further enhancements to this idea can also be made. A closed feedback system can be put in place so that the gripper/robot itself can be controlled based on the results obtained from the various sensor arrays. A method to implement pseudo force to test out the robot must be forthcoming in order to safely manipulate and run tests on it without actually loading the robot. A web publishing method can be established in order to view and control all data displayed on the GUI in any network by creating a Hyper Text Transfer protocol (HTTP) which should let the user view and control the data with the help of certain plug-ins on any PC or device which has an internet connection. Different materials can be used for testing such as Teflon to replace the Aluminium H11 alloy to test further effectiveness. Strain or other sensor tests for bending and vibration impacts can also be carried out apart from buckling alone.

4. Conclusions

Theoretical strain analysis was carried out on the strut and strain gauges were also tested using standardised cantilever beam with an error percentage less than 10% thus the strut system and the strain gauges are all within acceptable limits and will show the desired results. The strain values were obtained from DAQ. Accurate strain results were obtained for the sensor array via NI 9219 Analog module and were displayed in the GUI, i.e. LabVIEW. Moreover, the results were displayed in real time in a remote device such as a mobile phone using the NI Data Dashboard app thus facilitating remote control. Thus, all the requirements to prove this concept of wireless data transfer and control on a remote device from a sensor array affixed on a robotic gripper has been affirmed.

References

[1] Jagadeesh, R., Ranganatha, R., and Siddaraju, C. (2018). “Design and implementation of an architecture for a remotely operated plc laboratory using labview.” 8, 419–426.
[2] Chandra Mouli, C. (2013). “Design and implementation of robot arm control using labview and arm controller.” 6, 80–84.
[3] Santoso, D., Maryanto, S., Nadhir, A., and Sugiharto, T. (2017). “A simple and low cost data acquisition system with multi-nodes facility for geophone array sensors.” 12, 2109–2114.
[4] Engelbrecht, N. and Penzhorn, W. T. (2005). “Secure authentication protocols used for low power wireless sensor networks.” Proceedings of the IEEE International Symposium on Industrial Electronics, 2005. ISIE 2005, Vol. 4. 1777–1782.
[5] Harikrishnan, R. (2015). “An integrated xbee arduino and differential evolution approach for localization in wireless sensor networks.” Procedia Computer Science, 48, 447 – 453. International Conference on Computer, Communication and Convergence (ICCC 2015).
[6] Kabir, A. F. M. S., Shorif, M. A., Li, H., and Yu, Q. (2014). “A study of secured wireless sensor networks with xbee and arduino.” The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014). 492–496.

[7] Liu, Q., Xu, Z., and Li, Z. (2017). “Implementation of hardware tcp/ip stack for daq systems with flexible data channel.” 53.

[8] Mukaro, R. and Francis Carelse, X. (2000). “A low-cost microcontroller-based data acquisition system for solar radiation monitoring.” 48, 1232 – 1238.

[9] Sung, W. T., Chen, J. H., Hsiao, C. L., and Lin, J. S. (2014). “Multi-sensors data fusion based on arduino board and xbee module technology.” 2014 International Symposium on Computer, Consumer and Control, 422–425.

[10] Yussoff, Y., Abidin, H. Z., Rahman, R. A., and Yahaya, F. H. (2010). “Development of a pic-based wireless sensor node utilizing xbee technology.” 2010 2nd IEEE International Conference on Information Management and Engineering. 116–120.

[11] Nikhade, S. and Agashe, A. (2015). “Wireless sensor network communication terminal based on embedded linux and xbee.” 1468–1473.

[12] Segura, F.; Bartolucci, V.; Andújar, J.M. Hardware/Software Data Acquisition System for Real Time Cell Temperature Monitoring in Air-Cooled Polymer Electrolyte Fuel Cells. Sensors 2017, 17, 1600.

[13] Kim, H.S., Han, J.S and Lee.Y.H. (2012). “Scalable network joining mechanism in wireless sensor networks.” 2012 IEEE optical Conference on Wireless Sensors and Sensor Networks. 45–48.

[14] Kulakov, A., Davcev, D and Trajkovski, G. (2005). “Application of wavelet neural networks in wireless sensor networks.”

[15] Escobar, R.F.; Adam-Medina, M.; Garcia-Beltrán, C.D.; Olivares-Peregrino, V.H.; Juárez-Romero, D.; Guerrero-Ramírez, G.V. Monitoring and Control Interface Based on Virtual Sensors. Sensors 2014, 14, 20645-20666.

[16] Hercog, D.; Gergič, B. A Flexible Microcontroller-Based Data Acquisition Device. Sensors 2014, 14, 9755-9775.

[17] Chen, C.-Y.; Liu, C.-Y.; Kuo, C.-C.; Yang, C.-F. Web-Based Remote Control of a Building’s Electrical Power, Green Power Generation and Environmental System Using a Distributive Microcontroller. Micromachines 2017, 8, 241.

[18] Shahzad, A.; Landry, R.; Lee, M.; Xiong, N.; Lee, J.; Lee, C. A New Cellular Architecture for Information Retrieval from Sensor Networks through Embedded Service and Security Protocols. Sensors 2016, 16, 821.

[19] Shahzad, A.; Lee, M.; Xiong, N.N.; Jeong, G.; Lee, Y.-K.; Choi, J.-Y.; Mahesar, A.W.; Ahmad, I. A Secure, Intelligent, and Smart-Sensing Approach for Industrial System Automation and Transmission over Unsecured Wireless Networks. Sensors 2016, 16, 322.

[20] Chaos, D.; Chacón, J.; Lopez-Orozco, J.A.; Dormido, S. Virtual and Remote Robotic Laboratory Using EJS, MATLAB and LabVIEW. Sensors 2013, 13, 2595-2612.