Strain feedback gain tuning using neural network for the vibration control in a multilink flexible manipulator

Article · December 2018

DOI: 10.1299/jsmermd.2018.1P1-I08

Paul Waweru Njeri
Gifu University

Minoru Sasaki
Gifu University

Strain feedback gain tuning using neural network for the vibration control in a multilink flexible manipulator

○ Waweru Njeri, Student Member, Gifu University, v3812104@edu.gifu-u.ac.jp
Minoru Sasaki, Member, Gifu University
Kojiro Matsushita, Gifu University

Despite the numerous advantages associated with the flexible manipulators, link vibrations stand in the way to reaping these benefits. This leads to time wastage waiting for vibrations to decay to safe operating levels and the possibility of mechanical failure due to vibration fatigue. This paper presents direct strain feedback vibration control by tuning the feedback gains using artificial neural networks on a 3D flexible manipulator. Online backpropagation was developed in MatLab Simulink and implemented in dSPACE environment for practical experiments. Results show significant reduction in the link vibration relative to the performance of fixed feedback gain.

Key Words: Flexible manipulator, link vibrations, neural networks, strain feedback gain tuning

1 Introduction

In the recent past, robots are becoming smaller and lighter. This is accompanied by reduction in operation cost owing to the small size actuators used and links which are light in weight. However, this is followed by link vibrations especially when the manipulator is operated at high speed and with growing number of links. This stands in the way from reaping the numerous advantages of the flexible manipulator in terms of time wastage, lack of repeatability, lack of precision and possibility of failure due to vibrational fatigue[1, 2].

The theory of Direct Strain Feedback(DSFB) was developed by Z. Luo in [3, 4] and experimented with one-link flexible manipulator. In this control scheme, the strain measured at the root of the link is multiplied by a constant gain and the resulting signal applied as a negative feedback. The author show that the asymptotic stability is guaranteed and that the technique can damp out link vibration by increasing stiffness of the links. Similar work is presented in [5] improving on the original idea by approaching it as a nonlinear problem. The main advantage of DSFB is its simplicity in that the knowledge of the plant model is not required. Loading of the manipulator comes with additional complications which makes the manipulator to vibrate at a slower frequency and for vibrations to be more severe and persist for a longer period of time [6].

In this article, artificial neural network is trained to tune the strain feedback gain to counter strain induced by the change of direction. Multilayer, feedforward neural network utilizing the popular backpropagation with momentum is developed for this task. Backpropagation method is a gradient descent method that establishes the weight in an artificial neural network. Learning is accomplished by successively adjusting the weight based on a set of input patterns and a corresponding set of desired output patterns. During this iterative process, an input pattern is presented to the network and propagated forward to determine the resulting signal at the output units[7,8]. The differences between the actual resulting output signal and the predetermined desired output signal in each output unit represents an error that is backpropagated through the network in order to adjust the weights. The learning process continues until the performance index is less than a preset value[9, 10].

The artificial neural network program is implemented as a Matlab *s-function* developed in *c language* and integrated into an existing dSPACE system. Experiment is carried out from a personal computer connected to a 3D 2 link flexible manipulator. The experimental setup is as shown in Figure 1.

Fig.1: Experiment setup

2 Neural network formulation

The self tuning strain feedback gain configuration is a shown in Figure 2. The plant is excited with the reference joint angle signal modified by the actual joint angle negative feedback and the link strain pre-multiplied by the feedback gain k. The feedback gain k is tuned using artificial neural network depending on the error signal $e(t)$ due to changes in the joint trajectory and the link root strain $\varepsilon(0,t)$.

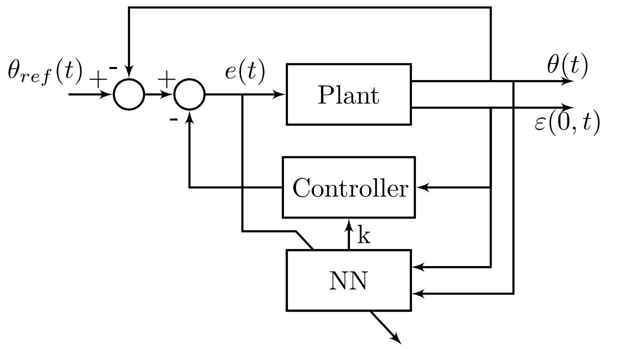


Fig.2: Self tuning strain feedback gain configuration

Based on the backpropagation algorithm at the output layer

$$\Delta w_{jk}(t+1) = -\eta \frac{\partial J}{\partial w_{jk}} + \alpha \Delta w_{jk}(t)$$

and the hidden layer

$$\Delta w_{ij}(t+1) \quad = \quad -\eta \frac{\partial J}{\partial w_{ij}} + \alpha \Delta w_{ij}(t)$$

where

$$J \triangleq \frac{a}{2}|e(t)|^2 + \frac{b}{2}|\varepsilon(0,t)|^2 \tag{1}$$

Defining δ_k

$$\delta_k \quad = \quad -\frac{\partial J}{\partial Net_k}$$

where

$$Net_k = \sum_j w_{jk} O_j + b_j$$

using chain rule

$$\begin{aligned}
\delta_k &= \delta_{k1} + \delta_{k2} \\
\delta_{k1} &= \frac{\partial J}{\partial e(t)} \frac{\partial e(t)}{\partial k} \frac{\partial k}{\partial Net_k} \\
&= ae(t) \frac{\partial e(t)}{\partial k} && (2) \\
\delta_{k2} &= \frac{\partial J}{\partial \varepsilon(0,t)} \frac{\partial \varepsilon(0,t)}{\partial k} \frac{\partial k}{\partial Net_k} \\
&= be(t) \frac{\partial \varepsilon(0,t)}{\partial k} && (3)
\end{aligned}$$

letting the two partial derivatives(jacobi) in 2 and 3 be equal to unity, we have

$$\delta_k = a|e(t)| + b|\varepsilon(0,t)| \tag{4}$$

For the hidden layer having tansigmoid activation function we have

$$\begin{aligned}
\delta_j &= \sum_k \frac{\partial J}{\partial Net_k} \frac{\partial Net_k}{\partial O_j} \frac{\partial O_j}{\partial Net_j} \\
&= -\sum_k \delta_k w_{jk} f'(Net_j) \\
&= -\sum_k \delta_k w_{jk} (1 - O_j^2)
\end{aligned}$$

Thus the update rules for the weights w_{ij}, w_{jk} and biases b_{ij}, b_{jk} are given by

$$\begin{aligned}
w_{ij}(t+1) &= w_{ij}(t) + \eta \delta_j O_j + \alpha \Delta w_{ij}(t) \\
w_{jk}(t+1) &= w_{jk}(t) + \eta \delta_k O_k + \alpha \Delta w_{jk}(t)
\end{aligned}$$

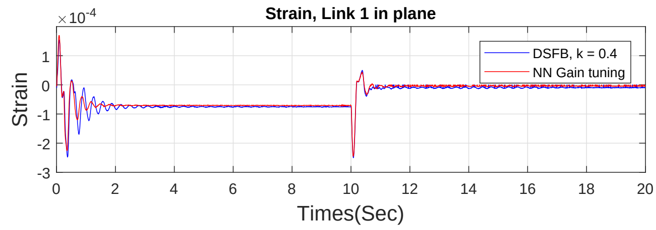
where η and α are the learning and the momentum rates respectively.

3 Results and discussion

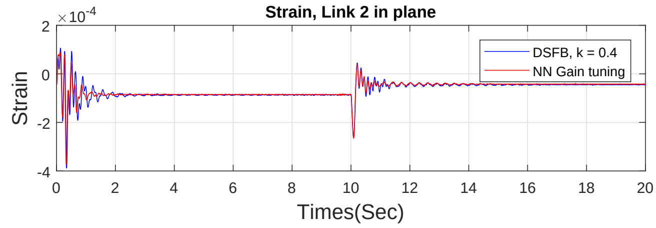
The manipulator presented in this paper has 2-flexible links and it has a load at the distal end. It is sitting on a rotary joint giving three degrees of freedom. The three joints are driven by three servomotors controlled from a personal computer running the Matlab™ and dSPACE™ applications. The neural network has 6 inputs , 12 hidden neurons and 1 output neuron corresponding to the strain feedback gain k.

The inputs to the network are $\theta(t)$, $e(t)$, $\varepsilon(0,t)$, $\theta(t-1)$, $e(t-1)$, $\varepsilon(0,t-1)$. Tansigmoid activation function is employed in the hidden layer and linear function in the output layer. In this experiment, we used $a = 0.05$, $b = 0.2$ and learning rate $\eta = 2 \times 10^{-4}$ and the sampling rate of 0.002. The experiment involved moving the three joints at an angle of 20 degrees using a step signal lasting for 10 seconds followed by returning the links to their original position for 10 more seconds. Strain measurement is achieved by attaching strain gauges at the root of respective links.

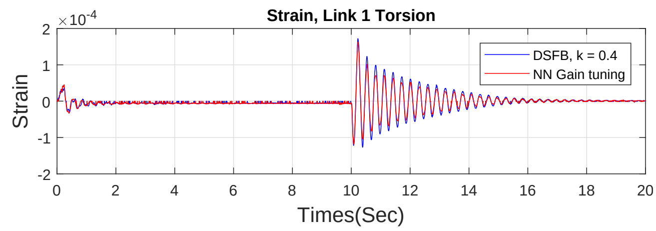
Figure 3 show the link strain for the manipulator with neural network tuning and fixed gain k = 0.4.



(a) Link 1, Torsion



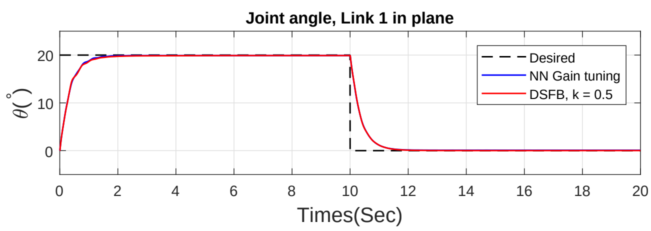
(b) Link 1, in plane



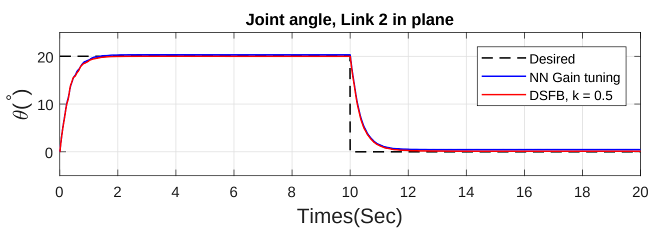
(c) Link 2, in plane

Fig.3: Link strain

Figure 4 show the joint trajectories for the manipulator with neural network tuning and fixed gain k = 0.4.



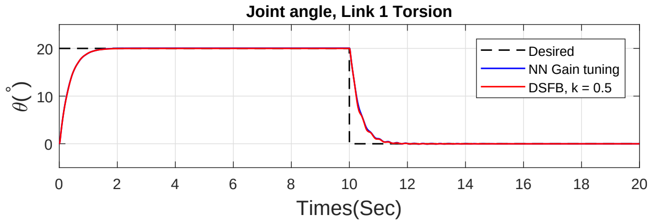
(a) Link 1, Torsion



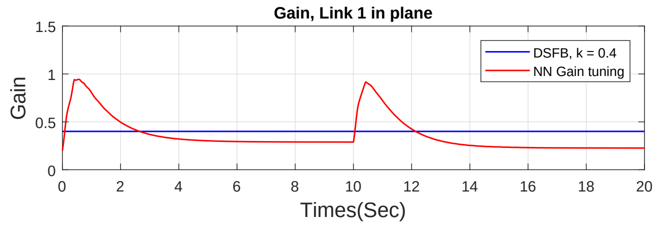
(b) Link 1, in plane

Figure 5 show the link strain for the manipulator with neural network tuning and fixed gain k = 0.4.

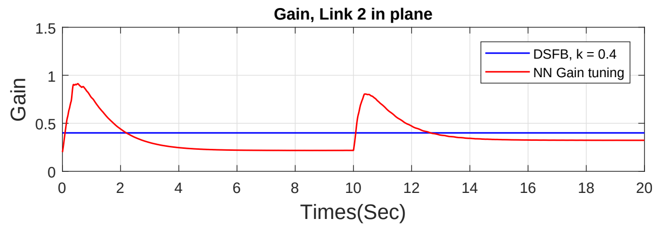
Figure 6 show the link strain for the manipulator with neural network tuning and fixed gain k = 0.4.



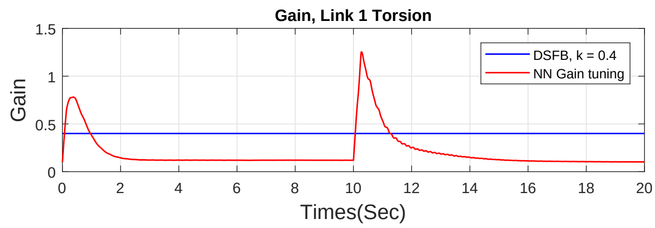
(c) Link 2, in plane
Fig.4: Joint trajectories



(a) Link 1, Torsion



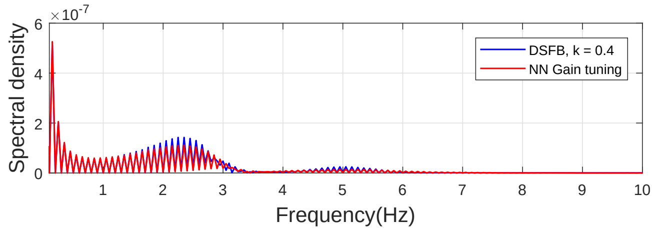
(b) Link 1, in plane



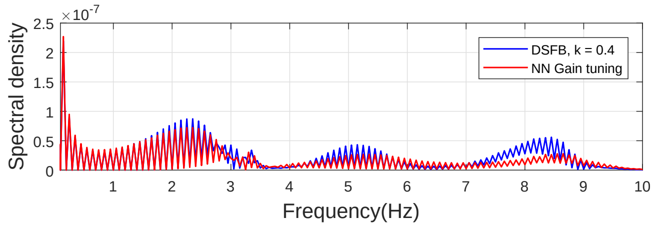
(c) Link 2, in plane
Fig.5: Feedback gains

4 Conclusion

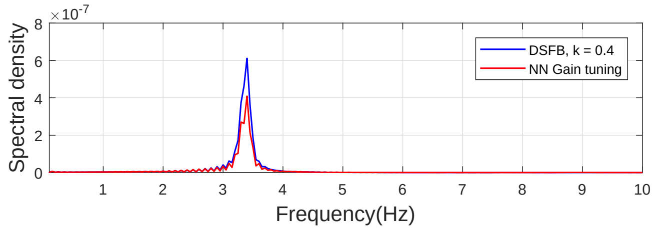
This paper presented a self tuning strain feedback gain controller for a 3D 2 link flexible manipulator using neural network. Neural network successfully learned the suitable gain for the flexible manipulator. Practical results shows that artificial neural network tuner achieves better performance than that of achieved using fixed gain.



(a) Link 1, Torsion



(b) Link 1, in plane



(c) Link 2, in plane
Fig.6: Performance index

References

[1] B. Subudhi and A. Morris, "Dynamic modelling, simulation and control of a manipulator with flexible links and joints," *Robotics and Autonomous Systems*, vol. 41, no. 4, pp. 257 – 270, 2002. `http://www.sciencedirect.com/science/article/pii/S0921889002002956`.

[2] S. Tso, T. Yang, W. Xu, and Z. Sun, "Vibration control for a flexible-link robot arm with deflection feedback," *International Journal of Non-Linear Mechanics*, vol. 38, no. 1, pp. 51 – 62, 2003. `http://www.sciencedirect.com/science/article/pii/S0020746201000403`.

[3] Z.-H. Luo, "Direct strain feedback control of flexible robot arms: new theoretical and experimental results," *IEEE Transactions on Automatic Control*, vol. 38, pp. 1610–1622, Nov 1993. `http://doi.org/10.1109/9.262031`.

[4] Z.-H. Luo and Y. Sakawa, "Gain adaptive direct strain feedback control of flexible robot arms," in *TENCON '93. Proceedings. Computer, Communication, Control and Power Engineering.1993 IEEE Region 10 Conference on*, vol. 4, pp. 199–202 vol.4, Oct 1993.

[5] S. S. Ge, T. H. Lee, and G. Zhu, "Improving regulation of a single-link flexible manipulator with strain feedback," *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 179–185, Feb 1998.

[6] A. Morris and A. Madani, "Static and dynamic modelling of a two-flexible-link robot manipulator," *Robotica*, vol. 14, no. 3, p. 289300, 1996.

[7] S. Aoshima, K. Takeda, T. Yamada, and T. Yabuta, "Autotuning of feedback gains using a neural network for a small tunnelling robot," in *Intelligent Robots and Systems '91. 'Intelligence for Mechanical Systems, Proceedings IROS '91. IEEE/RSJ International Workshop on*, pp. 221–226 vol.1, Nov 1991.

[8] K. Takahashi, "Remarks on self-tuning feedback controller using the clifford multi-layer neural network," in *2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA)*, pp. 1–6, Sept 2015.

[9] Minoru Sasaki, Toshimi Shimizu, Yoshihiro Inoue, and Wayne J. Book, "Self-tuning vibration control of a rotational flexible timoshenko arm using neural networks," *Advances in Acoustics and Vibration*, vol. 2012, p. 7, 2012.

[10] A. Otsuka and F. Nagata, "Auto gain-tuning for trajectory following control based on neural network," in *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*, pp. 128–133, Dec 2014.