27th INTERNATIONAL CONGRESS ON SOUND AND VIBRATION
12 – 16 July 2020, Prague

ICSV27 PRAGUE

Annual Congress of the International Institute of Acoustics and Vibration (IIAV)

AN EARLY HARDWARE PROTOTYPE OF A MINIATURE LOW-COST FLEXIBLE LINK EXPERIMENT

Gergely Takács, Martin Vríčan, Erik Mikuláš and Martin Gulan
Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Bratislava, Slovakia
e-mail: gergely.takacs@stuba.sk

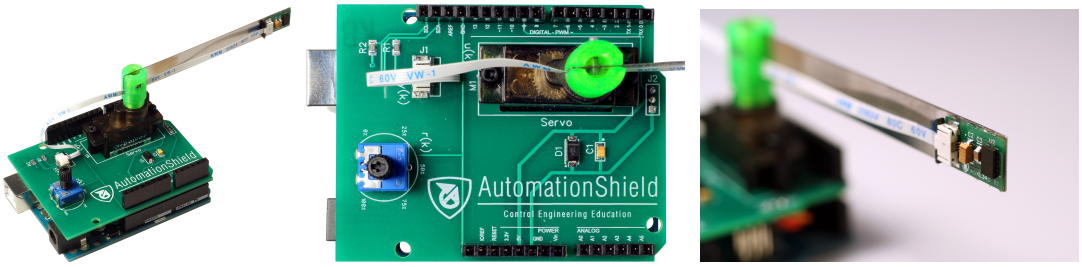
This article presents a reference design for an early prototype of a low-cost miniaturized rotational flexible link experiment. Flexible link systems use a vertically placed motor connected to a horizontal flexible cantilever beam with feedback reading to simulate the vibration response of dynamic systems in motion——such as high-speed slender flexible robotic manipulator arms. We introduce a low-cost hardware design that uses the Arduino microcontroller prototyping boards as a basis and builds the pocket laboratory on top of these devices as an expansion module. The preliminary hardware design uses a micro servo motor as an actuator and a micro-electromechanical system (MEMS) accelerometer as a sensor. The prototype reference design, including editable schematic files, printed circuit board (PCB) layout, 3D models of mechanical components for printing and component list are made open-source and available for replication and improvement by the community. We verify the functionality of our design by a system identification and a closed-loop control example.

Keywords: rotational flexible link, vibration control, open educational resources, educational technology, microcontrollers

1. Introduction

In order to achieve a high directional accuracy, industrial robots and manipulators are designed to be stiff. Unfortunately, increased stiffness is simply achieved by adding mass, thus requiring bulky drives and ultimately resulting in reduced speed and elevated energy consumption [1]. The dynamic response of these manipulators is then well below the first fundamental frequency [2], thus, are often modeled and controlled as stiff structures. Flexible link robots may offer an alternative to traditional designs, and achieve similar manipulation accuracy if vibrations are handled correctly. The price of lighter systems with faster movement speeds and modest energy use is thus a more complicated control system that is focused at active vibration control (AVC). On the other side, increasing manipulation speeds mean, that we cannot consider robots as completely stiff structures anymore [3], and studying their elastic vibration response becomes of utmost importance.

Studying flexible robotic systems in research laboratories and teaching these concepts to engineering students requires experimental laboratory hardware. A single link flexible robotic manipulator can be represented by a vertically placed motor connected to a flexible beam, where the dynamic response of



(a) Isometric view. (b) Top view. (c) Beam tip detail.

Figure 1: Miniature low-cost flexible link device.

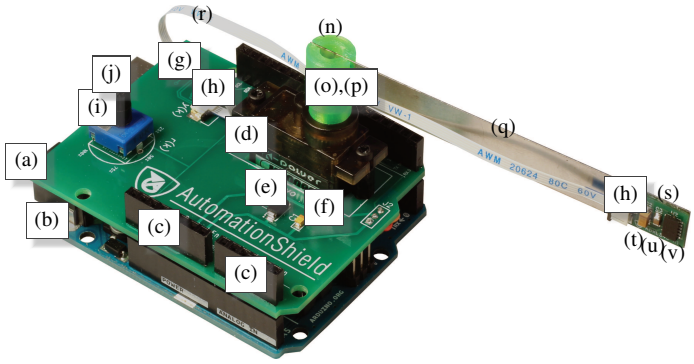


Figure 2: Annotated photograph of the assembled device.

the structure can be measured by e.g. a strain gauge or accelerometer. Such a system is commercially available, and numerous articles available in the research literature feature it, c.f. [1, 3], etc. Such commercial solutions are highly reliable precision instruments, which are even supplied with complete teaching material and enable scientists to compare estimation and control algorithms across individual research teams. Unfortunately turnkey solutions tend to be expensive in the range of tens of thousands of Euros, which is prohibitive for many laboratories. Not only this, but students cannot take hardware home for experimentation on the account of their size, sensitivity and cost.

The alternative approach to acquiring the necessary hardware for many laboratories is to construct them in-house, e.g. [2, 4]. While the prices of these devices range typically in hundreds of Euros, thus representing a considerable saving, the resulting devices are one-off prototypes with a design that is unavailable to the public. Hence, by manufacturing improvised hardware, researchers and educators lose the advantages of unified platforms that make common teaching materials and experiments possible.

In previous work we have introduced a hybrid concept to create experimental laboratory devices that are both low-cost, miniature and standardized [5, 6]. Our idea is founded on utilizing the widely available Arduino microcontroller unit (MCU) prototyping boards, that can be functionally expanded by printed circuit boards (PCB) connected to the top of electronic header connectors. The reference designs of laboratory devices that are built on these so-called *shields* are then publicly shared, along with a corresponding application programming interface (API) and educational examples.

Thus, in this article, our interest is focused on creating a miniaturized and low-cost version of the flexible rotational link experimental device. Here we present an early hardware prototype (see Fig. 1) along with a provisional API and an identification and closed-loop control example. Our long-term, non-commercial effort to create open hardware and software for control education is called *AutomationShield* (c.f. [7]), while the proposed flexible link device shall be referred to as the *LinkShield* in this article.

2. Hardware Design

The final assembled prototype is featured in Fig. 2 with annotation, which will be explained in detail in the next paragraphs. The schematic drawing of the proposed educational and research aid is shown in Fig. 3, where Fig. 3(a) features the electronic connections of the base shield, while Fig. 3(b) illustrates the accelerometer breakout board. The reader may follow along the upcoming discussion by comparing the notation of identical components in other figures as well (see e.g. Fig. 2–Fig.5 and Tab. 1).

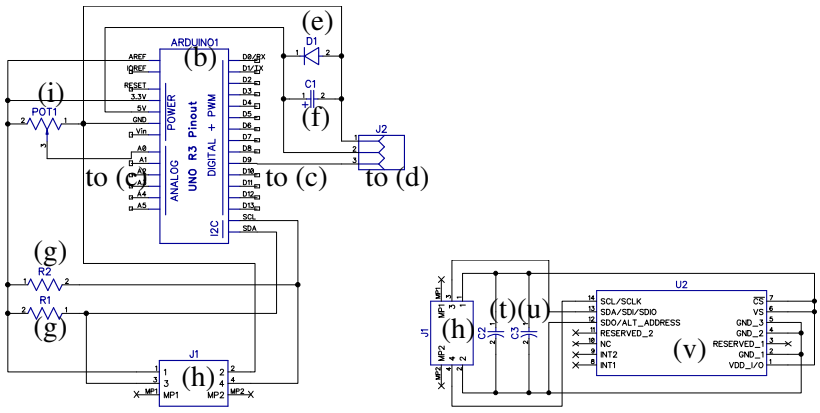
The mechanical base of the LinkShield is a standard two-layer 1.6 mm thick printed circuit board (a) that carries all electronic and mechanical components and acts as a foundation for the device. This is connected to an Arduino R3-layout compatible microcontroller prototyping board (b) by a set of stacking header pins (c). A metal-geared high-speed digital micro servo motor (d) is driven by the `D9` PWM capable pin of Arduino. Its power supply is drawn directly from the board, as the current consumption remains well below the allowable maximum. A diode (e) protects the microcontroller from reverse currents caused by possible back electromotive force (EMF), while transient effects on the servo supply are filtered by a capacitor (f).

To minimize the size of the accelerometer unit, we have included the I2C pull-up resistors (g) on the base board. A miniature connector (h) mounted to the shield supplies power to the acceleration sensor unit, which is connected to the I2C bus of the MCU by the `SCL` and `SDA` pins. The last component located on the base is a potentiometer (i) connected to the A0 analog pin, including a shaft (j), that allows the user to program this input for any purpose, such as providing reference to the feedback control loop.

Let us now move to the mounting of the servo motor and the beam. The servo is inserted into a pre-fabricated slot on the PCB and raised by 10 mm using a pair of spacers (k), fixed with polyamide screws (l) from the top and nuts (m) from the bottom. A 3D printed slotted cylindrical hub (n) connects the servo shaft with the beam. We fabricated this custom mechanical component by a Prusa i3 MK3/S 3D printer in PETG filament. Printing time is only 21 minutes; requiring a mere 1.1 g of filament that renders its cost to less than 0.04€ including the 0.07 kWh electricity consumption of the printer. The model for this part has been designed in Autodesk Fusion 360 and is included in the documentation [8]. The hub is held in place by a machine screw connecting the servo shaft that comes with most metal-geared servos as standard, while the slot holding the beam is tightened and fixed by a M2$\times$8 machine screw (o) and corresponding nut (p). The flexible cantilever beam (q) measuring 85$\times$10$\times$0.3 mm and with a ϕ2 mm mounting hole placed 5 mm from the edge is laser-cut from AISI 301 (S30100) stainless steel. Although any thin metal or possibly even plastic may be used for the beam, we have elected this particular material because of its increased elasticity.

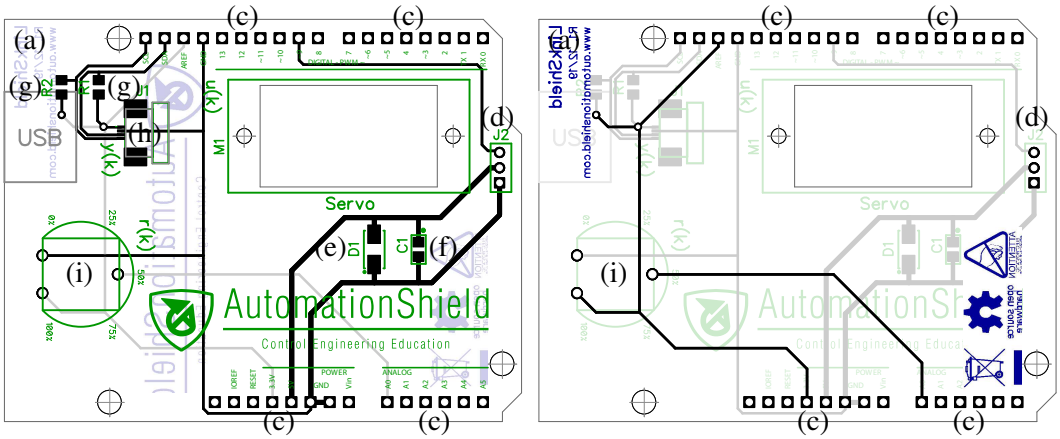
The tip of the beam is equipped by the accelerometer unit, which is connected to the base board by a 4-lead flexible flat cable (FFC) with a 0.5 mm pitch (r). This cable transfers power and communication to the accelerometer by a connector identical to the one located on the base board. The accelerometer unit is based on a single layer 0.6 mm thick printed circuit board (s). The PCB is glued firmly to the tip by Suxun B-7000 adhesive. Power to the accelerometer chip is filtered by a pair of 10 µF (t) and 100 nF (u) capacitors. Finally, the breakout PCB contains the Analog Devices ADXL345 3-axis configurable gain digital accelerometer unit (v). The tip mass may be modified by attaching neodymium magnets to the beam, thereby simulating a changing end-effector load. Such a concept may be then used to explain and research adaptive or robust control concepts.

Both the base unit and the accelerometer breakout board have been hand-soldered in-house using customary tools found on most electronics workbenches. The assembly of the surface mount devices (SMD) requires a fair amount of practice, but even placing the accelerometer with a 14 terminal land grid array (LGA) package is feasible without special equipment. If the proposed device is to be used in an educational setting, the assembly itself may be a valuable didactic experience.



(a) Base shield. (b) Accelerometer breakout.

Figure 3: Electronic schematics of the base board and the breakout module.

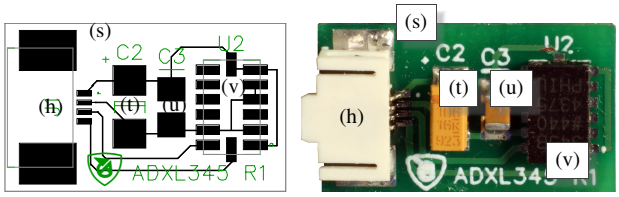


(a) Top layer of base shield. (b) Bottom layer of base shield.

Figure 4: Printed circuit board of the base shield, 1:1 scale.

We have kept global accessibility and ease of component purchasing in mind when designing the device. The motor, diode, capacitors, connectors and cable may be easily exchanged to a variety of alternatives. The PCB can be manufactured by online services, and 3D printing is easily available nowadays. The only required component is the accelerometer, however, this is also common and effortlessly obtainable as a discrete part. The electronic schematics (Fig. 3), PCB layout (Fig. 4,Fig. 5) and the 3D printed hub are all available as editable files [8, 7].

Table 1 summarizes all the components necessary to manufacture a LinkShield device. The most expensive item on the list is the digital metal-gear micro servo motor. The reader may notice the relatively low price estimate for the accelerometer chip, which is justified by the possibility of salvaging third-party ADXL345 breakout boards that are often cheaper than the component price at low volumes. The PCB manufacturing cost is calculated for an order of 10 boards each. The price list excludes postage, shipping, labor and other material costs. The prices for individual components are shown for low quantity purchases at various suppliers and exclude high volume purchasing deals. According to this, the price of a single device is 21.85 €. Even though overhead is excluded from this estimate, we believe that, in case the LinkShield is mass-produced, the large volume purchases and professional SMD assembly offsets these expenses and the unit price can be around ∼20–25 €. Such a price level makes the LinkShield an affordable teaching option for all laboratories and even students.



(a) Breakout PCB. (b) Breakout photo.

Figure 5: Printed circuit board of the breakout board, 1:2.5 scale.

Table 1: Component list and cost calculation in Euros.

Name	Part no., value	Desc.	Mark	Pcs.	Price	Total
Servo	digital, high-speed metal gear micro-servo (e.g. Savox SH-0257MG)	M1	(d)	1	16.76	16.76
Accelerometer	Analog Devices ADXL345	U2	(v)	1	0.89	0.89
Connector	0.5 mm pitch, 4-lead FFC/FPC (e.g. 52745-0497)	J1, J2	(h)	2	0.30	0.59
Potentiometer	10 kΩ, 250 mW (e.g. ACP CA14NV12,5-10KA2020)	POT1	(i)	1	0.10	0.10
Resistor	0805, 10 kΩ (e.g. ROYAL OHM 0805S8J0103T5E)	R1, R2	(g)	2	0.01	0.02
Capacitor	0805, tantalum, 4.7 µF (e.g. AVX TAJP475K016RNJV)	C1	(f)	1	0.15	0.15
Capacitor	1206, tantalum, 10 µF (e.g. T491A106M016AT)	C2	(u)	1	0.22	0.22
Capacitor	0805, ceramic, 100 nF (e.g. C0805C104M5RACTU)	C3	(t)	1	0.01	0.01
Diode	DO214AC (e.g. Vishay BYG20J, 1.5 A, 600 V)	D1	(e)	1	0.17	0.17
Cable	0.5 mm pitch, 4-lead FFC	-	(r)	1	0.12	0.12
PCB (shield)	2 layer, FR4, 1.6 mm thick, green mask	-	(a)	1	0.45	0.45
PCB (breakout)	1 layer, FR4, 0.6 mm thick (or less), green mask	-	(s)	1	0.45	0.45
Screw	M2×8, steel	-	(o)	1	0.02	0.02
Nut	M2, steel	-	(p)	1	0.01	0.01
Spacer	hexagonal; polyamide; M2; 10 mm	-	(k)	2	0.15	0.30
Screw	M2×5, Phillips, polyamide	-	(l)	2	0.13	0.25
Nut	M2, polyamide	-	(m)	2	0.06	0.12
Shaft	Potentiometer shaft, (e.g. ACP CA9MA9005)	-	(j)	1	0.10	0.10
Header	6×1, female, 2.54 mm pitch	-	(c)	1	0.06	0.06
Header	8×1, female, 2.54 mm pitch	-	(c)	2	0.09	0.18
Header	10×1, female, 2.54 mm pitch	-	(c)	1	0.09	0.09
Hub	1.1 g green PETG filament, 21 m to print, 0.07 kWh electricity	-	(n)	1	0.04	0.04
Magnets	ϕ9×2 mm, N50, ∼ 13 N (e.g. Omo Magnets N50D00960020)	-	-	3	0.12	0.36
Beam	85×10×0.3 mm, ϕ2 mm hole 5 mm from edge, AISI 301 (S30100)	-	(q)	1	0.40	0.40
					Total:	**21.85€**

3. Application Programming Interface

The application programming interface (API) serves as an abstraction layer to render experimentation with the LinkShield more straightforward for students and researchers alike. The API is currently available for the Arduino IDE in C/C++ and is a part of the AutomationShield library [7]. Every hardware-specific functionality is included in the `LinkShield` object, that is initialized automatically for the user when including the `LinkShield.h` header. The hardware is started by calling

```
LinkShield.begin();
```

which initializes the default Arduino servo motor library and attaches the motor to the `D9` pin. To maintain both 3.3 V system compatibility and analog resolution, the reference is set to external. Finally, the ADXL345 accelerometer is initialized with a 8 G range and 3200 Hz data rate, producing a 1600 Hz bandwidth. The sensor can be optionally calibrated by running the `calibrate()` method to remove sensor bias in the direction of interest. The acceleration sensor can be read at any time instant by calling

```
float y = LinkShield.sensorRead();
```

which returns a floating-point number providing acceleration data $y(k) = \ddot{q}(k)$ in m·s^{-2}. Moreover, the servo motor can be commanded to the internally kept position of $u(k)$ degrees by the

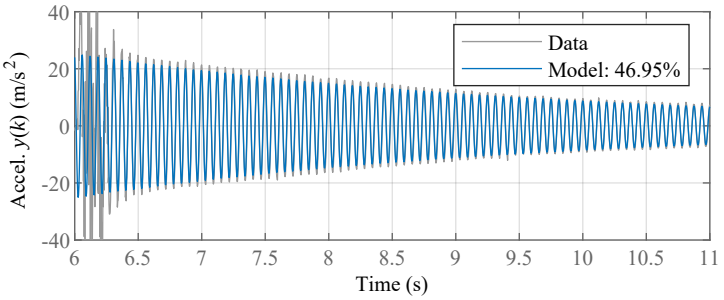


Figure 6: Modeled and measured tip acceleration including servo movement transients.

```
LinkShield.actuatorWrite(float u);
```

method. The board contains a programmable potentiometer as well; its state is returned in the range of 0–100% by the `referenceRead()` method.

4. Examples

Let us demonstrate the functionality of the proposed hardware by typical classroom examples. We shall first identify a simple dynamic model of the tip position $q(t)$ (m) in dependence of the servo angle $u(t)$ (deg). Furthermore, let us assume that the load carried by the end-effector consists of three stacked magnets weighing 2.84 g ($\sim$0.95 g each).

Assuming the manipulated load and the beam can be represented by a single dominant resonant mode, we have

$$\ddot{q}(t) + 2\zeta\omega\dot{q}(t) + \omega^2 q(t) = f(t), \tag{1}$$

where ω (rad·s^{-1}) is natural angular frequency, ζ (-) is the damping ratio and $f(t)$ is the force acting on the system. We may represent this force by assuming $f(t) = c\omega^2 u(t)$, where c (m·deg^{-1}) is the actuator constant. After performing a Laplace transform, we arrive at the well-known transfer function

$$P(s) = \frac{Q(s)}{U(s)} = \frac{c\omega^2}{s^2 + 2\zeta\omega s + \omega^2}. \tag{2}$$

Do not forget that we may only measure an acceleration signal $y(t) = \ddot{q}(t)$, however, may still use the same model structure for grey-box identification, then compensate for the discrepancy later. The servo angle was changed 90° while sampling the acceleration signal at a $T_\mathrm{s} = 0.003$ s rate, where the identification experiment loaded to the MCU is available in the `LinkShield_Identification.ino` source code.

The transfer function has been identified using the MATLAB System Identification Toolbox (R2019a). First, a data section with free vibration has been selected, then a continuous two-pole and no-zero transfer function has been obtained with a $\sim$92% fit to the estimation data in 13 iterations and 37 function evaluations using the instrument variable (IV) initialization method. The procedure is a part of the API and can be found under the `LinkShield_Identification_TF.m` file.

According to this, the angular natural frequency of the system with the given load is $\omega = 100.4$ rad·s^{-1} (16 Hz) and the damping ratio is $\zeta = 0.0027$. Remember that the numerator assumes an acceleration signal, thus, by modeling simple harmonic motion we may safely consider the position signal to be out-of-phase with acceleration and scaled down by ω^2. The resulting actuator constant is then c =-9.2718E-4 m·deg^{-1}. The comparison of the identified transfer function to test data is shown in Fig. 6, assuming we model the acceleration signal.

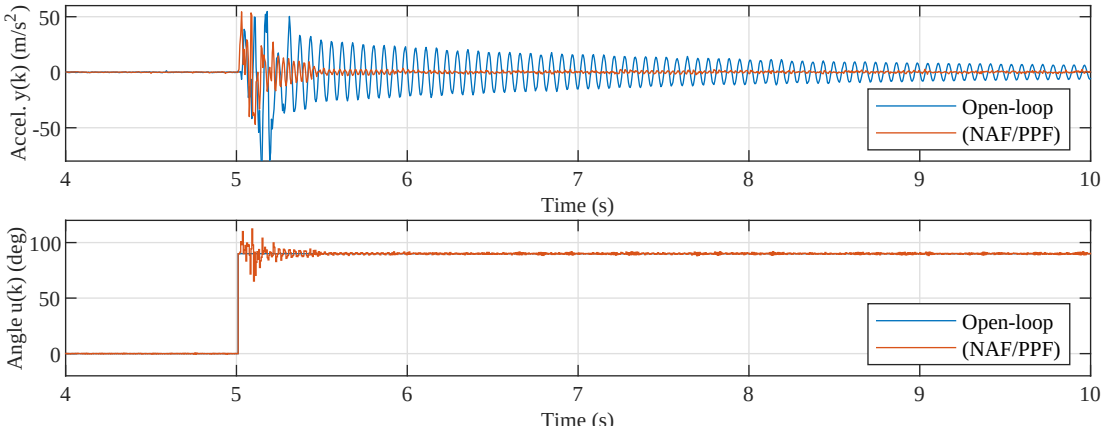


Figure 7: Beam tip vibrations and corresponding servo motor angles in open loop and under NAF control.

Let us now turn to the feedback control of the beam tip, or in other words, the stabilization of the end-effector of the simulated robotic arm. The manipulation angle is maintained by a feedback loop inside the servo motor to $r(t)$ (deg). This reference can be then modified by a linked tip controller of $u(t)$, so that the resulting overall angle is $u_{\mathrm{r}}(t) = r(t) + u(t)$. Let us the consider positive position feedback control (PPF) of the dominant structural mode, where $q(t)$ (m) is the position signal and $u(t)$ (deg) is the resulting servo angle. The PPF controller is given in the time domain by [9, 10]

$$\ddot{u}(t) + 2\zeta_{\mathrm{c}}\omega_{\mathrm{c}}\dot{u}(t) + \omega_{\mathrm{c}}^2 u(t) = g q(t), \tag{3}$$

where g (deg·m^{-1}) is the tunable controller gain, ω_{c} (rad·s^{-1}) is the angular frequency of the controller that usually equals to the angular natural frequency of the controlled mode and ζ_{c} (-) is the tunable controller damping ratio affecting the "sharpness" of the controller response.

As we are controlling a single dominant mode with a prevalent harmonic response, we may assume that the acceleration signal recorded by the system is out-of-phase to the position with an amplitude that is $\omega^2 = \omega_{\mathrm{c}}^2$ smaller. The PFF controller is then transformed to

$$\ddot{u}(t) + 2\zeta_{\mathrm{c}}\omega_{\mathrm{c}}\dot{u}(t) + \omega_{\mathrm{c}}^2 u(t) = -g\omega_{\mathrm{c}}^2 q(t), \tag{4}$$

which is effectively a negative acceleration feedback (NAF) controller. Let us not forget, that besides the orientation of the feedback, the scaling of the acceleration signal is eventually combined with the tunable gain g, thus is ultimately lost in the digital realization. The transfer function of the controller is

$$G(s) = \frac{U(s)}{Q(s)} = -g\,\omega_{\mathrm{c}}^2\,\frac{1}{s^2 + 2\zeta_{\mathrm{c}}\omega_{\mathrm{c}} + \omega_{\mathrm{c}}^2}, \tag{5}$$

which after selecting $\omega_{\mathrm{c}} = \omega = 100.4$ rad·s^{-1}, $\zeta_{\mathrm{c}} = 0.04$ (-) and the gain $g = 2$ deg·m^{-1} results in the $T_{\mathrm{s}} = 0.005$ s sampling discrete-time transfer function

$$G(z) = -2\,\frac{0.1218z + 0.1202}{z^2 - 1.719z + 0.9606}.$$

The computation and simulation of the NAF/PPC controller is a part of the example collection and is listed in `LinkShield_PPF.m`, while the digital realization of the discrete-time transfer function is listed in the `Linkshield_PPF.ino` source that is a part of the API. A representative experiment is shown in Fig. 7, where an open-loop response is compared to the closed-loop response obtained with the NAF/PPF controller presented above. As one may observe from the experimental data, the controller damps the tip vibrations very effectively whilst ultimately converging to the desired servo angle.

5. Conclusion

We have presented a reference design for a rotational flexible link laboratory device that is low-cost and miniaturized. The design is made open-source in the hope that the global community of control researchers and educators will find it useful, while allowing the possibility of a crowd-sourced improvement of the available hardware, software and examples. In further work we would like to make several changes to the hardware, namely to use a motor that is controlled in MCU based on an angle reading at the hub and to replace the accelerometer with an inertial measurement unit to produce angular displacement instead of an acceleration signal. As of software, we aim to include an API for Simulink and possibly MATLAB; and to expand the range of examples included with the library.

Acknowledgements

The authors gratefully acknowledge the contribution of the Slovak Research and Development Agency (APVV) under the contracts APVV-18-0023, APVV-17-0214 and APVV-14-0399. The authors appreciate the financial support provided by the Cultural and Educational Grant Agency (KEGA) of the Ministry of Education of Slovak Republic under the contract 005STU-4/2018.

REFERENCES

1. Skosarev, E. and Kolyubin, S. Case study on energy-aware position control for flexible link mechatronic systems, *2019 IEEE International Conference on Mechatronics*, Ilmenau, Germany, pp. 617–621, (2019).
2. Zhang, X., Sørensen, R., Iversen, M. R. and Li, H. Computationally efficient dynamic modeling of robot manipulators with multiple flexible-links using acceleration-based discrete time transfer matrix method, *Robotics and Computer-Integrated Manufacturing*, **49**, 181–193, (2018).
3. Ettefagh, M. H., Naraghi, M. and Mahzoon, M. Experimental control of a flexible link by the method of controlled Lagrangian, *Journal of Theoretical and Applied Vibration and Acoustics*, **4** (1), 81–98, (2018).
4. Ikizoğlu, S. and Gürışık, O. LQR based optimal control for single-joint flexible link robot, *6th International Conference on Control Engineering Information Technology*, Istanbul, Turkey, pp. 1–6, (2018).
5. Takács, G., Gulan, M., Bavlna, J., Köplinger, R., Kováč, M., Mikuláš, E., Zarghoon, S. and Salíni, R. HeatShield: a low-cost didactic device for control education simulating 3D printer heater blocks, *2019 IEEE Global Engineering Education Conference*, Dubai, United Arab Emirates, pp. 374–383, (2019).
6. Takács, G., Konkoly, T. and Gulan, M. Optoshield: A low-cost tool for control and mechatronics education, *12th Asian Control Conference*, Kitakyushu, Japan, pp. 1001–1006, (2019).
7. Takács, G. *et al*., (2018-2020), *AutomationShield: Control Systems Engineering Education*. Online, [cited 17.1.2020]; Available from `http://www.automationshield.com`.
8. Gulan, M., Vríčan, M. and Takács, G., (2020), *LinkShield*. Online, [cited 1.16.2020]; GitHub Wiki page. Available from `https://github.com/gergelytakacs/AutomationShield/wiki/LinkShield`.
9. Goh, C. J. and Caughey, T. K. On the stability problem caused by finite actuator dynamics in the collocated control of large space structures, *International Journal of Control*, **41** (3), 787–802, (1985).
10. Preumont, A., *Vibration Control of Active Structures*, Kluwer Academic Publishers, 2. edn. (2002).