

Teleoperation of a surgical robot using force feedback

Dániel Bolgár, Simon Krogh, Filip Marić, Nicolas Silvani
Aalborg University
Department of Electronic Systems
{dbolga16, skrogh13, fmaric16, nsilva16}@student.aau.dk

Abstract—Haptic feedback is a way of transferring information to the user via the sense of touch, usually through the same input device the user gives commands with. This makes it ideal for teleoperating tasks requiring precision in applied force, robotic minimally invasive surgery (MIS) being a prime example. Currently, haptic feedback in teleoperation is subject to numerous constraints on time delay and accuracy. Nonetheless, results show that implementing this type of feedback in teleoperated robotic surgery results in a higher success rate compared to the traditional robotic MIS. In this paper, we focus on improving the haptic feedback on the da Vinci robot at Aalborg University using the existing hardware. The method involves using a state-of-the-art haptic device to control a surgical tool serving as the robot's end-effector. Since the dynamics of the surgical tool are strongly nonlinear, estimation techniques are used to calculate reaction forces on the device. Changes are made to the existing communication protocols in order to reduce time delay.

I. INTRODUCTION

Interest in robotic surgery has increased over the last couple of years. Increased precision provided by surgical robots introduces a decrease in tissue damage, thus reducing the recovery time [1]. Robots used in robotic surgeries have an attached end-effector that is used as a surgical tool. One such tool is the EndoWrist. The main advantage of the EndoWrist lies in its construction, as it is made to be manipulated in a similar manner to the operators wrist. During surgery the operator receives 3D video feedback from the control loop.

The problem with the operator having exclusively visual feedback lies in the fact that the surgeon has to estimate the force applied by observing the color changes of the skin and organs for each maneuver. This constant effort from the operator not only increases the operation time but also leads to errors such as thread breaking during stitching or damaging tissues by applying too much force [2]. It has been shown experimentally that haptic feedback has a considerably positive effect on the reduction of surgical error [3].

The purpose of haptic feedback in Robotic MIS (RMIS) is to restore the sense of touch for the surgeon. To do so, different approaches have been studied such as vibrotactile feedback [4] [5], force feedback [6] or both [7]. However, most of the solution developed introduce new hardware on the robot, such as sensors or processing units. Some studies on force feedback without additional sensors, have shown promising results [8].

Although force feedback still requires a haptic feedback device for the surgeon, the implementation on the robot can be made completely by software.

Direct force feedback method involves calculating the feedback from the resistance affecting the actuators. However, as the tool is highly nonlinear, the output power is lower than the input power, due to the tool's inherent damping. Any forces related to the construction of the robot are not desired in the feedback as the operator would not feel them when holding a tool. In order for the operator to feel as if he was directly holding the tool the control system should be transparent to him.

The haptic feedback could be done as direct force feedback calculated from the resistance affecting the actuators, but as the tool is highly nonlinear, the transparency of the controller would suffer from it. It would be possible to solve this problem by implementing a sensor on the end-effector to measure the force, but due to the demand for high hygiene, the tools have to be sterilized at temperatures over a 100° C which could damage the sensor. Furthermore, each surgical tool has to be discarded after a few uses [9]. This means that the cost of the tool has to stay as low as possible and therefore make the idea of implementing an expensive sensor not ideal. Therefore the force feedback has to be estimated through the actuators, which requires a dynamic model of the tool. From this model the forces related to the actuation of the tool can be estimated and the external forces applied at the end-effector can be computed. The response time and the frequency of the force feedback control loop have to be considered as any of those could break the transparency of the controller if too high. The frequency of the control loop is directly related to the frequency of the communication between the different components of the system. It is widely discussed what the minimum refresh rate of the feedback loop should be but seems to be somewhere between 300 Hz and 1000 Hz depending of the hardness of the object [10].

In this paper, an attempt is made to implement a force feedback on a setup emulating the essential parts of the da Vinci Robot without implementing new hardware. To do so, a dynamic model is derived and a control strategy is proposed. The aimed frequency for the feedback loop is 550 Hz, to reach that goal analysis of the communication protocol is provided. In

section II, we will take an overview of our proposed control system as a whole, briefly presenting each of the components and their interaction. Section III will cover the methods used to create a dynamic model of the EndoWrist and proposed methods of translating the estimated force to actual force fed back to the operator. Section IV contains descriptions of the modification made in order to improve the refresh rate of the communication between devices. Finally, we present the experimental results in Section V and draw a short conclusion in Section VI.

II. SYSTEM OVERVIEW

A da Vinci robot with connected EndoWrist has four arms with 6 - 7 actuated Degrees Of Freedom (DOF) each. Each arm has its own drivers and an embedded system controlling it. The surgeon sends the commands to the embedded system by controlling the console.

As mentioned in Section I, the EndoWrist is highly nonlinear and constitute most of the challenge in the modelization of one arm of the robot, thus, if the force feedback can effectively be applied to the tool, it can be extended to the arm. As each arm is independent, the system designed for one arm can be extended to the full arm, the same applies to the tool. The setup used in the present project only controls one EndoWrist which has four DOF and is further described in Section II-B. To control the EndoWrist, the embedded system of the robot is used. On a da Vinci robot the surgeon performs the operation from a console that communicates with the robot. The manipulators for the robot do not implement an interface for haptic feedback, thus, a haptic device, a Geomagic Touch (GT), is used in this setup instead of the device used on the official console. This device is further described in Section II-A.

As force estimation requires computational power and an interface is required to exchange information between GT and embedded system, a computer is added to the system and connected to both devices using Ethernet cables. The entire setup is represented in Fig. 1

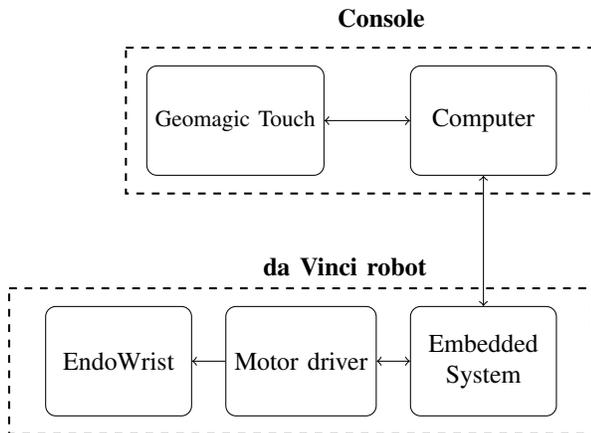
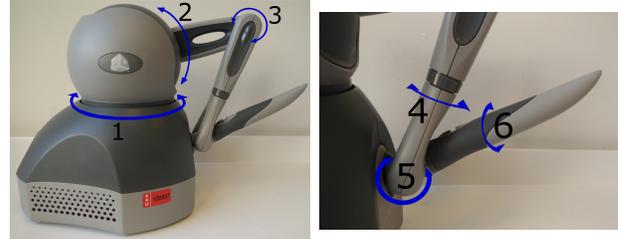


Fig. 1: Block diagram representing the system.

A. Geomagic touch

The Geomagic Touch is a haptic feedback device, which has the ability to actuate its joints in such a way that the user feels resistance when moving the pen.



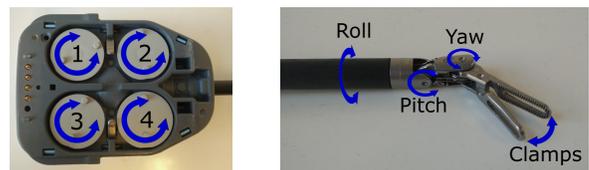
(a) Overview of the Geomagic Touch's first three joints. (b) Overview of the Geomagic Touch's last three joints

Fig. 2: Overview of all the Geomagic Touch's joints.

On Fig. 2, it can be seen that the Geomagic Touch has six DOF, where the first three can be actuated, see Fig. 2a. This means that the device has the ability to generate force feedback with three translational DOF, in this case corresponding to the EndoWrist's roll, pitch and yaw movements.

B. EndoWrist

An EndoWrist, see Fig. 3, is a surgical tool for the da Vinci robot which can be manipulated in a similar manner as a human wrist. It provides the surgeon the ability to operate with the robot as the operator would without it. To replicate the movements of a human wrist, the tool is composed of a system of cables and pulleys. This construction imitates the human tendons however it also introduces nonlinearities in the tool, and thus, a challenge for controlling or modeling it.



(a) Actuator plates, which can manipulate the end effector position (b) End-effector of the EndoWrist

Fig. 3: The EndoWrist and its end-effector

In real operation, each arm has six to seven actuated DOF in total, however, the EndoWrist itself, when disconnected from the robot, only has four DOF. Each DOF is actuated through a plate, see Fig. 3a. The four DOF are roll, pitch, yaw and clamp, see Fig. 3b. The nonlinearities of the tool are analysed when building the model in Section III.

III. FORCE ESTIMATION

In order to have a representation of the reaction force on the EndoWrist, estimation is needed. As stated in Section II, the force cannot be measured on-line using sensors and thus we have to rely on mathematical models as functions of actuator measurements.

A. Mathematical model

The main challenge faced in making a mathematical model lies in the fact that the pulley system of the EndoWrist is nonlinear, and thus its full dynamics cannot be modeled in a straightforward manner. The nonlinearity of the EndoWrist dynamics emerges from friction and elasticity of the wires controlling the end-effector, which causes multiple pulleys to move as a result of actuating only one. In other words, to have an accurate representation of Cartesian force an intricate model is required [11].

Another method of approaching this problem lies in creating multiple mathematical models pertaining to forces output by actions performed with the EndoWrist. In this manner, the feedback vector is transformed from Cartesian space to a task space in which the chosen actions form a basis. Each element of the new feedback vector corresponds to an actuated axis of the Geomagic Touch. For the purpose of this system, we choose to feedback the yaw force generated by the grip action of the clamp, the force generated by the roll actuator and force exerted by the clamps pitch movement, as seen in 3.

B. System Identification

Ideally, a mathematical model derived from classical mechanics would be used to describe the dynamics involved in the EndoWrist's movements. However, deriving this model precisely enough for grey-box identification has been proven difficult and time consuming due to the nonlinear nature of the dynamics.

Models for yaw and pitch forces are derived using black-box identification algorithms, which only provide a general model structure. A straightforward approach would involve choosing a nonlinear model structure for identification. On the other hand, stability analysis of nonlinear models is difficult and due to the nature of the system, only general trends in force need to be represented. For this reason it was decided to identify a linear state-space model which is then used as a part of Hammerstein-Wiener [12] nonlinear model as seen in Fig. 4.

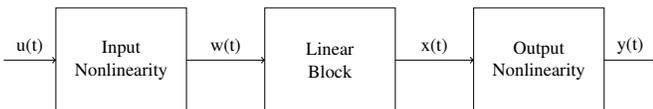


Fig. 4: Block diagram of a Hammerstein-Wiener model.

1) *Roll torque*: The roll torque, which is determined by the roll actuator and directly rotates the entire tool, making it independent to the rest of the system. We can model the roll torque as a state that only depends on input in the state-space model of the system (1).

$$\mathbf{x}(k+1) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \mathbf{A}_{pitch} & 0 \\ 0 & 0 & \mathbf{A}_{yaw} \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} \mathbf{B}_{roll} & 0 & 0 \\ 0 & \mathbf{B}_{pitch} & 0 \\ 0 & 0 & \mathbf{B}_{yaw} \end{bmatrix} \mathbf{u}(k) \quad (1)$$

$$\mathbf{y}(k+1) = \begin{bmatrix} \mathbf{C}_{roll} & 0 & 0 \\ 0 & \mathbf{C}_{pitch} & 0 \\ 0 & 0 & \mathbf{C}_{yaw} \end{bmatrix} \mathbf{x}(k) \quad (2)$$

2) *Pitch and yaw forces*: The pitch and yaw force state-space models were determined using subspace identification [13]. This algorithm combines concepts from system theory, linear algebra and statistics in order to provide a state-space model of the system, which makes it useful for MIMO system identification.

Actuator effort and velocity were used as inputs during identification. The order of the identified model was picked using Hankel singular values [14], which determine the amount of data dynamics a model can describe at a given order. For both the pitch and yaw forces, it was determined that 6th order models are sufficient.

Deadzone nonlinearities were estimated for the inputs and output of the models. The resulting Hammerstein-Wiener models were validated on previously unused data.

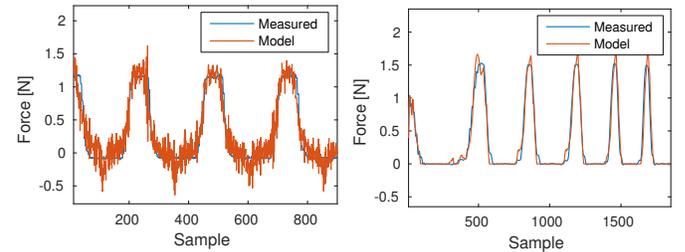


Fig. 5: Comparison of pitch (left) and yaw (right) model to measurements.

IV. COMMUNICATION

The embedded system controls the motors for one EndoWrist. The desired positions of the motors and the list of the enabled motors are sent to the board from the computer using an Ethernet cable. To perform force estimation, the computer needs to receive the list of motors currently actuated as well as the position, velocity and effort for each of them.

As mentioned in Section I the frequency aimed for the force feedback loop is 550 Hz. However this loop not only includes communication between the embedded system and the computer but also computation time for force estimation and communication between the GT and the computer. Thus, the

communications with the computer must be faster than 550 Hz. The drivers for the GT have a refresh rate of 1000 Hz. From experimentation it was found that the embedded system's built-in UDP library cannot handle refresh rates higher than 1000 Hz. Thus the resulting feedback loop have a refresh rate of 500 Hz and not 550 Hz as it was aimed. This study aims at reaching the maximum frequency of the embedded system which is 1000 Hz.

The original system implements a stream of Javascript Object Notation (JSON) [15] files using TCP. However this communication setup can not reach a frequency higher than 100 Hz, which does not match the goal of 1000 Hz. Thus, this section focuses on the modifications applied to the communication in order to make it reach the requirements.

In order to get a faster communication it was decided to use UDP instead of TCP as it does not retransmit any packets or implement any features to improve long distance communications. In our system, retransmission of packets would lead to retransmitting obsolete data instead of transmitting new ones. Furthermore, improvements of long distance communications would be superfluous since the two devices are directly connected.

In addition to the transport protocol, another factor that influence the speed of the communication is the size of the packets. To maximize the number of packets sent, the size of those packets must be minimized while keeping the computation time as low as possible. As stated before, the packets sent to the computer contain position, velocity, effort and a boolean value for each motor. The JSON used in the original system creates a human readable file and thus, use one character per digit in a number. In this setup the numbers can go up to 23 characters. The size of the numbers, combined to the additional characters required for the JSON leads to packets of 346 bytes in a worst case scenario, those packets are described in Fig. 6. To reduce the size of those packets, it was decided to interpret the binary representation as characters instead of the human readable format. As the numerical values are stored as floats following the IEEE 754 standard [16], each of them require four bytes. Also, a constant structure was define to remove the need of control sequences. The new packets are described in Fig. 7 and have a constant size of 49 bytes. Thus, the size of the packets was reduced by 86%.

To investigate the quality of the communication as a function of frequency three parameters were measured: the delay between two packets received, the jitter and the error rate. Since the computation time on the computer and on the embedded system are very small compared to the frequency of the communication, i.e. inferior to 3 μ s.

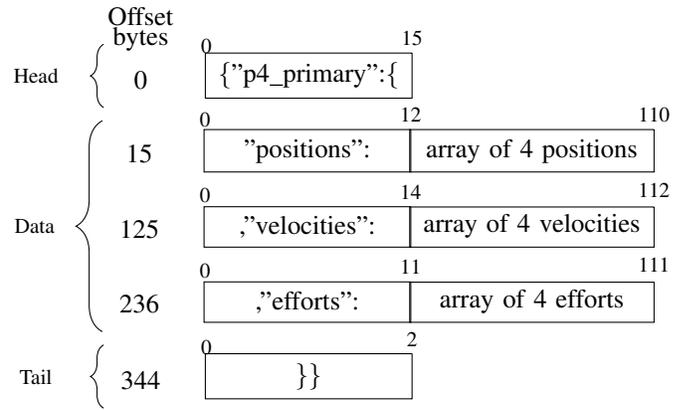


Fig. 6: Packet using JSON

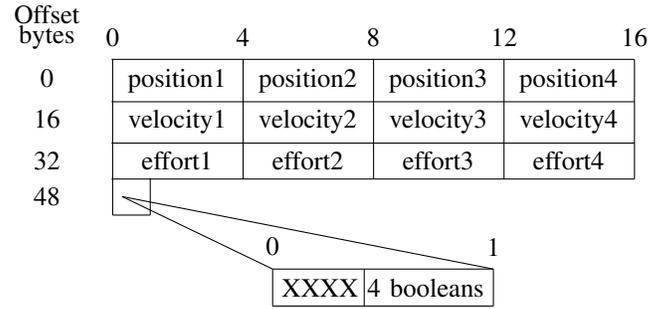


Fig. 7: Packet using the binary representation

V. RESULTS

A. Communication

As shown in TABLE I, when the frequency increases from 100 Hz to 500 Hz, so do the jitter and packet loss. However when the refresh rate is increased to its maximum value the jitter sharply decreases while the packet loss increases significantly.

Frequency (Hz)	delay (ms)	Jitter (μ s)	Packet loss (%)
99	10.1	4.66E-2	0
474	2.1	5.51E-2	0.2
638	1.6	1.16E-2	1.2

TABLE I: Measurements of the UDP performances

The goal of 1000 Hz could not be reached when running the entire system. However, as the original communication could not exceed the refresh rate of 100Hz a significant improvement is to be noted.

B. Force feedback

As seen in Fig. 8, the yaw force fed back to the user by the Geomagic Touch dynamically corresponds to both the current increase and the position error. We have found that applying

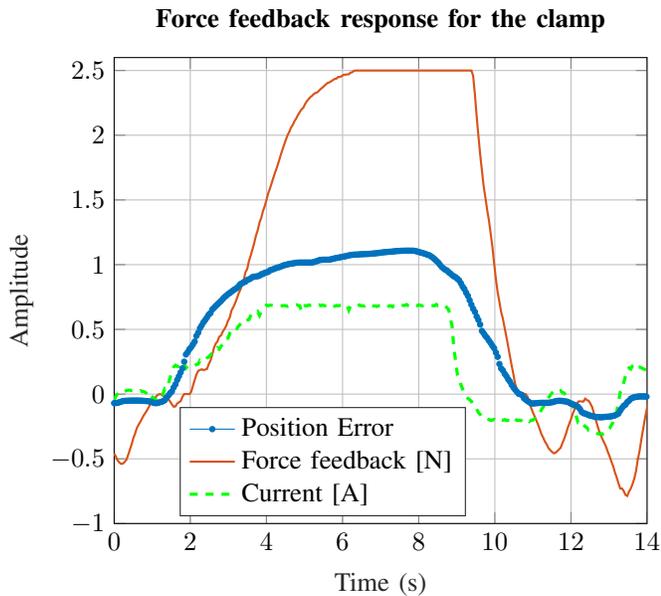


Fig. 8: Measurements of the response of the force feedback for the clamp

a gain to the feedback provides the user with a better sense of the force exerted by the EndoWrist.

VI. DISCUSSION

When increasing the refresh rate, increasing jitter and packet loss were expected as these parameters are highly correlated to the network congestion [17]. The drop in jitter when reaching for higher frequency can be explained by the way the communication driver was designed. In order to reach high frequencies, a trade-off was made by setting a deadline to receive a packet, if the packet does not meet the deadline, it is discarded. Thus, when the jitter increases, more packets are discarded, increasing the packet loss. The maximum refresh rate reached does not meet the goal previously set. However, compared to the original system, the refresh rate has been increased by more than six folds. Furthermore, the goal of 1 kHz was reached when only the communication with the embedded system was enabled, thus it is expected that optimizing the program could lead to reaching the goal even with the entire system running. To further increase the refresh rate, compression of the data was considered and it is believed that implementing a fast compression algorithm such as the one described in [18] could reduce the time required to transmit a packet.

By choosing UDP as a transport protocol, every network reliability feature was removed from the connection which matches the demands of our system in term of bandwidth. However, safety needs to be considered for such a system. As such a feature was implemented on both sides of the communication in order to detect packet loss and connection timeout. The detection of those events allows to stop moving

the end-effector and to notify the operator. In the future, additional steps such as protection against external cyber-attacks and handling of packet losses should be taken in order to improve the overall safety of the system.

Due to the EndoWrists structure, we have chosen to model the dynamics of the tool in a task space consisting of roll torque and pitch and yaw forces. As we have chosen a data-based approach towards force modeling, all imperfections in the data acquisition process can affect the result. Roll torque estimation was simple as the data showed its linear dependence on the actuator effort.

The yaw (grip) force model has shown an average 77% fit to validation data. While the errors in the model output exist, they usually involve the estimate being slightly lower than the actual force. This is mostly due to the saturation nonlinearity implemented in the model output, which prevents the linear part of the model from overshooting the estimate.

Unlike the yaw model, data acquisition for the pitch model was more difficult, as the EndoWrists structure the affected measurements. This results in additional nonlinearities in the measurements, since force wasn't always applied to an angle perpendicular to the load cell. As a consequence, the model underestimates the applied force.

An attempt was made to implement state estimation the correct the force estimate. The steady-state Kalman filter was implemented, with position error and velocity measurements used for state estimation. Simulation results have shown that such us system would not improve the systems, as the current models do not capture the dynamics adequately.

In the future, an improved model could be utilized with state estimation to provide state feedback control of the outputted force. The force reference could be directly mapped to the Geomagic Touch movement axes, providing a greater degree of control to the system.

VII. CONCLUSION

Force feedback has been implemented using the data that can already be measured on a surgical robot. The refresh rate of the communication has been increased by reducing the size of the data sent and by implementing a new communication protocol. A model that estimates the force has been built. The system designed can be used as a basis for future implementation of force feedback on a surgical robot.

REFERENCES

- [1] P. C. Giulianotti, A. Coratti, M. Angelini, F. Sbrana, S. Cecconi, T. Balestracci, and G. Caravaglios, "Robotics in general surgery: personal experience in a large community hospital," *Archives of surgery*, vol. 138, no. 7, pp. 777–784, 2003.
- [2] C. Lee, Y. H. Park, C. Yoon, S. Noh, C. Lee, Y. Kim, H. C. Kim, H. H. Kim, and S. Kim, "A grip force model for the da vinci end-effector to predict a compensation force," *Medical & biological engineering & computing*, vol. 53, no. 3, pp. 253–261, 2015.

- [3] J. C. Gwilliam, M. Mahvash, B. Vagvolgyi, A. Vacharat, D. D. Yuh, and A. M. Okamura, "Effects of haptic and graphical force feedback on teleoperated palpation," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 677–682.
- [4] M. O. Culjat, C.-H. King, M. L. Franco, C. E. Lewis, J. W. Bisley, E. P. Dutton, and W. S. Grundfest, "A tactile feedback system for robotic surgery," in *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2008, pp. 1930–1934.
- [5] S. Peddamatham, W. Peine, and H. Z. Tan, "Assessment of vibrotactile feedback in a needle-insertion task using a surgical robot," in *2008 Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. IEEE, 2008, pp. 93–99.
- [6] T. Ortmaier, B. Deml, B. Kübler, G. Passig, D. Reintsema, and U. Seibold, "Robot assisted force feedback surgery," in *Advances in telerobotics*. Springer, 2007, pp. 361–379.
- [7] A. R. Peon, C. Pacchierotti, and D. Prattichizzo, "Vibrotactile stimuli for augmented haptic feedback in robot-assisted surgery," in *World Haptics Conference (WHC), 2013*. IEEE, 2013, pp. 473–478.
- [8] A. M. Okamura, "Haptic feedback in robot-assisted minimally invasive surgery," *Current opinion in urology*, vol. 19, no. 1, p. 102, 2009.
- [9] *EndoWrist/Single-Site Instrument & Accessory Catalog*, http://www.intuitivesurgical.com/products/871145_Instrument_Accessory_%20Catalog.pdf, Intuitive Surgery, 2014.
- [10] T. R. Coles, D. Meglan, and N. W. John, "The role of haptics in medical training simulators: a survey of the state of the art," *IEEE Transactions on Haptics*, vol. 4, no. 1, pp. 51–66, 2011.
- [11] C. Y. Kim, M. C. Lee, R. B. Wicker, and S.-M. Yoon, "Dynamic modeling of coupled tendon-driven system for surgical robot instrument," *International Journal of Precision Engineering and Manufacturing*, vol. 15, no. 10, pp. 2077–2084, 2014.
- [12] Y. Zhu, "Estimation of an n - l - n hammerstein-wiener model," *Automatica*, vol. 38, no. 9, pp. 1607–1614, 2002.
- [13] P. Van Overschee and B. De Moor, *Subspace identification for linear systems: Theory—Implementation—Applications*. Springer Science & Business Media, 2012.
- [14] W. Gawronski and J.-N. Juang, "Model reduction in limited time and frequency intervals," *International Journal of Systems Science*, vol. 21, no. 2, pp. 349–376, 1990.
- [15] E. T. Bray, "The javascript object notation (json) data interchange format," *RFC 7159*, Google, Inc., 2014.
- [16] I. S. Committee *et al.*, "754-2008 ieee standard for floating-point arithmetic," *IEEE Computer Society Std*, vol. 2008, 2008.
- [17] Cisco. Understanding jitter in packet voice networks (cisco ios platforms). <http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/18902-jitter-packet-voice.pdf>. Document ID: 18902, Accessed: 16-12-2016.
- [18] W. N. Ross, "An extremely fast ziv-lempel data compression algorithm," *Proceedings of Data Compression Conference*, pp. 362–371, 1991.