Journal of Engineering Sciences, Assiut University, Vol. 35, No. 3 pp. 713-728, May 2007

REINFORCEMENT LEARNING FOR DESIGNING TENDON-DRIVEN ANTHROPOMORPHIC ROBOT HAND

Khaled M. Shaaban; and Abdulmajeed F. Aljunaid
Electrical Engineering Department, Assiut University, Assiut, Egypt.

(Received March 18, 2007 Accepted April 23, 2007)

One of the problems common to tendon-driven anthropomorphic robot hands is the dependency problem. Dependency arises when guiding tendons around joints not through the center of articulation which makes the length of the tendon paths for some joints depend upon the angle of other joints. Instead of the bulky mechanical solutions for this problem, this work proposes handling this problem at the software level. The core of the solution is a mapping-function that associates the desired joint angles to the correct servomotor angles accounting for all the dependencies in the system. The geometrical analysis to get this function is difficult due to the complexity of the paths of the tendons around the phalanxes of the fingers. This work proposes getting this function through learning. Using model based learning requires the equally complex analysis to build the model. Therefore, this work proposes learning by interacting with the real physical system. To evaluate the system a simple setup of an anthropomorphic robot hand was developed and used to evaluate the performance of the proposed technique. The test was done on the index finger.

KEYWORDS: *Anthropomorphic Robot Hand, Reinforcement Learning, Dependency Problem and Tendons.*

1. INTRODUCTION

In the last two decades, there was an increasing interest for developing anthropomorphic robot hands [1-9]. This interest is due to many reasons such as:

- Tele-operation of anthropomorphic robot hand for remote reproduction of the human hand behavior. This gives the flexibility to use robot hand as slave device in dangerous or impossible to reach places. For example robots could be used for space exploration. Robots are sent to space mission transmitting information back to Earth, with no intent of returning home.
- Using robot in a man-oriented environment, where interaction between humans and robots is needed to perform tasks that may be done by humans or robots as well. For example medical robots could use surgical equipments while helping surgeons during operations.
- Prosthetic hands that have the shape and function of a human hand for helping people who lost their hands.

These reasons motivated more than one project around the world like NASA Robonaut Hand [1], Utah/MIT Hand [2], Gifu Hand [9], DLR Hands [6-8], UB Hands

[4, 5, 10] and Ultralight Anthropomorphic Hand [3]. Reviews on anthropomorphic robotic could be found in [11, 12].

The mechanical structure of the robot hand affects the degree of anthropomorphism and dexterity. An important structure aspect is the location of the actuators which may be inside the fingers or remote at the forearm. The advantage of the first selection is that no transfer mechanism is needed to transfer the action from actuators to the joints of the finger. But placing the actuators inside the fingers makes the hand bulky and heavy and also decreases the degree of anthropomorphism. For example a motor in the proximal joint must be powerful enough to lift all the outer motors. The second selection needs a complex transfer mechanism to transfer action from the forearm to the fingers. Although there are other mechanisms, using tendons is the most popular choice for action transfer [2-4], [11].

Tendons are commonly used in robot hands because they are light, clean and small in size allowing more space to add sensors and other devices. Furthermore, tendons are more anthropomorphic which allows for more dexterous motions. A properly pretensioned tendon has minimum backlash which improves the precision of the hand. Using tendons suffer from problems like erosion which results from friction.

Another problem which is of significant to this work is the dependency problem. Dependency occurs from the way the tendons are placed around the joints. Guiding tendons around the joints makes the lengths of the tendon paths for some joints depend upon the angle of other joints. Instead of the usual heavy mechanical solutions [2], [13-18], this work proposes compensating this dependency at the software level (the intelligent control part of the system). Such solution needs a mapping-function that relates the actuators and the finger joint angles. This function accounts for all the dependency relations with other joint angles to facilitate generating the needed actuator commands. To avoid the heavy mathematical modeling, necessary to generate the mapping function this paper further proposes a simple learning scheme.

The rest of the paper is arranged as follow. Section 2 provides a discussion of the dependency problem. Section 3 describes the proposed reinforcement learning method. The experimental results and the hardware setup description are provided in Section 4 followed by a conclusion in Section 5.

2. TENDON DEPENDENCY PROBLEM

In the human hand, tendons routing inside the finger is provided by internally lubricated sheaths connected to the bones as shown in Figure (1). Sheaths connected to the bones maintain the position of the tendons relative to the phalanxes, and thus to the line of action of the finger. This biological configuration allows high adaptability during finger bending, with high efficiency and much reduced cross section. The sheaths are responsible for the smooth action of the tendons. Tendons not connected to a joint, but routed around it, do not pass through the center of articulation. Therefore the lengths of tendon paths vary with the angle of the joints. That is, a tendon not connected to the two bones forming a joint, but guided around this particular joint is affected by its angle. For that reason, the rotation of a joint requires modifying the length of not only the tendons attached to it but also of all tendons guided around it. The large number of tendons and joints increases the complexity of motion. For example, consider the 30 tendons necessary to move the 15 phalanxes of a human hand; the rotation of the wrist joint while maintaining the position of all joints in the hand requires relaxation/tension of 30 muscles. This motion dependency, made constructing an anthropomorphic robot hand a great challenge.

Traditional solution of the dependency problem in robot hands uses pulleys or external sheaths or both to decouple the motion of the joints [2], [13-18]. A successful pulley based mechanism makes changing the angle of a joint requires adjusting the angle of the actuators connected to that joint only. Unfortunately, pulleys make hand bulky and large, and increase the required mounting surfaces. Figure (2) shows the complexity of pulleys system of the Utah/MIT finger.

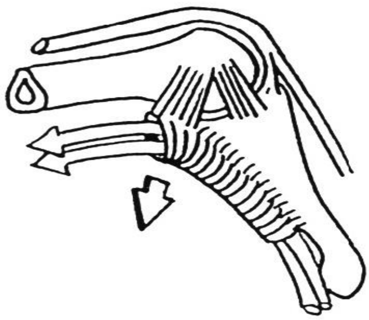


Figure (1) A scheme of biological sheath routing of tendons

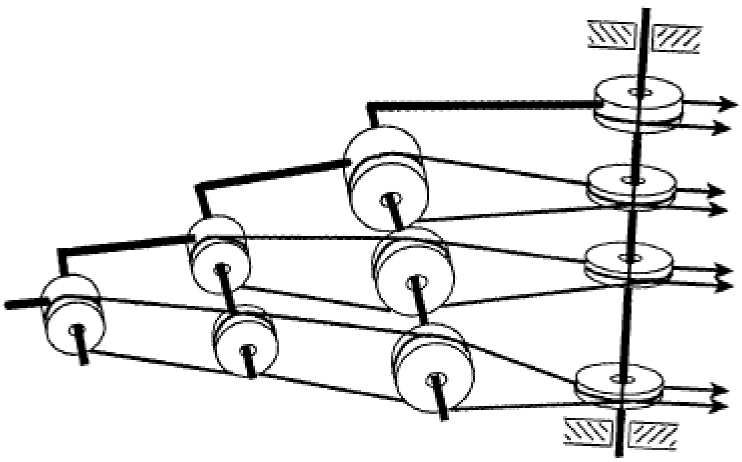


Figure (2) The link structure and tendons routing of the Utah/MIT finger [17]

The second solution is using mechanical sheaths to guide the tendons. The mechanical sheath is simply a flexible tube having a fixed length with a wire running inside it (like the hand brake wire of a bicycle). The difference of length between the wire and the tube, represents the action (the motion) required at the remote end of the sheath. Unlike biological sheath, the mechanical one does not change its length with the change of the joint angles not directly connected to it. That is, the mechanical sheath solution does not suffer from the dependency problem. Unfortunately, the main draw back of this mechanism is that the mechanical sheath should be kept free to move in order to be able to maintain its fixed length. That is, it should not be fixed to the structure of the hand from the joint all the way to the actuator. Such a problem makes the sheath wires exposed and free to move around joints especially around the wrist joint. In many applications (like medical and aerospace application) this open frame or exposed configuration are not tolerated. The mechanical sheaths were used in LRP five fingers artificial hand [19].

This work proposes tendons routing through internal sheaths, more like the biological hand. These sheaths are channels machined inside the fingers and through the palm. The channels are lubricated to decrease friction. Clearly as with biological hand, the proposed system suffers from dependency problem.

The proposed system suggests compensating for the path length changes at the software level. That is, it suggests using a mathematical model to generate the necessary action to all tendons involved in any given motion.

In general the number of joints is not necessary equal to the number of actuators. Some or all of the joints could be operated using two independent actuators. Therefore, let the number of the joints to be n and the number of actuators to be m. The vector $\boldsymbol{\theta}$ represents the angles of the joints $\boldsymbol{\theta} = [\theta_1 \theta_2 \cdots \theta_n]^T$ and the vector $\boldsymbol{\varphi}$ represents the angles of actuators $\boldsymbol{\varphi} = [\phi_1 \phi_2 \ \cdots \phi_m]^T$. Then the relation between $\boldsymbol{\varphi}$ and $\boldsymbol{\theta}$ could be written as:

$$\boldsymbol{\varphi} = \zeta(\boldsymbol{\theta}) \tag{1}$$

where, ζ is a mapping function that represents all the complexity and dependencies of the transfer mechanism. With careful mechanical design we could guarantee linearity of the mapping relation. Under this condition, the model is a set of linear equations typically represented in a matrix form that is called the structure matrix, $\mathbf{A}$ [13, 17, 18].

$$\boldsymbol{\varphi} = \mathbf{A}\boldsymbol{\theta} \tag{2}$$

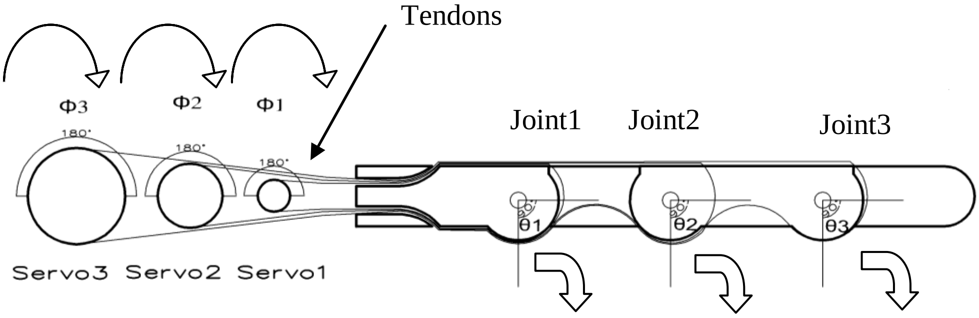


Figure (3) Scheme of tendon internal sheaths in the finger and servomotors

For example, consider the single finger (three joints) and its actuators drawn in Figure (3). One actuator is used per joint. In order to rotate joint1 to an angle θ_1, the three actuators must rotate to a specific angles ϕ_1, ϕ_2 and ϕ_3. Otherwise the tension of the tendons of joint2 and joint3 will oppose the rotation of joint1. For this configuration the structure matrix, $\mathbf{A}$ of Equation (2), is 3×3 square matrix.

If the matrix $\mathbf{A}$ is found then the operation of the finger is simple. If a certain values of $\boldsymbol{\theta}$ are needed the corresponding angle for all three actuators could be evaluated using Equation (2). Unfortunately getting $\mathbf{A}$ requires complex geometrical analysis of the paths of the tendons.

3. PROPOSED SOLUTION

As an alternative to the complex analysis needed to get the structure matrix $\mathbf{A}$, this work proposes learning. Constructing a model for the learning process requires the same complex analysis needed for getting the structure matrix. Therefore, the model based learning is not beneficial and learning by interacting with the real physical system is the most practical alternative. This class of learning is usually called Reinforcement Learning.

Reinforcement Learning (RL) is defined as the problem of an agent that learns a task through a direct interaction with the environment. The agent senses the environment then selects an action. Depending on the effect of the action, the environment rewards the agent. The agent's general goal is to maximize the amount of reward. That is, the agent's goal is to learn a better policy that maps state to action [20-22].

The proposed scheme to solve the tendons dependency problem fits the above RL definition except of the part of the reward. The reward in the proposed method is abstract and not measurable; it is to minimize the $\boldsymbol{\varphi}$ to $\boldsymbol{\Theta}$ mapping error. That is, the proposed system goal is to learn the optimal policy (matrix $\mathbf{A}$) that maps desired joint angles to actuator angles. The state in the dependency problem is the desired joint angles only. The current values of the joint angles are omitted from the system state because the actuators are servomotors with their own feedback loop. The internal servo mechanism of the actuators enables them to execute the commands (the desired positions) automatically.

To understand the proposed scheme, consider n DOF system in which every two opposite tendons are controlled by one actuator. Consequently, the number of actuator is n and the system could be described by $n \times n$ structure matrix, $\mathbf{A}$ given in Equation (2). The reverse representation of this equation is also useful to the proposed learning scheme and could be written as:

$$\boldsymbol{\Theta} = \mathbf{B}\ \boldsymbol{\varphi} \tag{3}$$

Where $$\mathbf{B} = \mathbf{A}^{-1} = \begin{pmatrix} b_{11} & b_{12} & \cdots & \cdots & b_{1n} \\ b_{21} & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ b_{n1} & \cdots & \cdots & \cdots & b_{nn} \end{pmatrix}$$

The learning is performed in cycles that we distinguish by the superscript k . In each cycle, the target joint angles, $\boldsymbol{\theta}^k = [\theta_1\theta_2 \cdots \theta_n]^T$ of all the joints are arbitrary selected. At the same step, k , the estimated structure matrix is denoted $\mathbf{A}^k$. Using Equation (2) a command vector to the actuator, $\boldsymbol{\varphi}^k = [\phi_1\phi_2 \ \cdots \phi_n]^T$ is generated using:

$$\boldsymbol{\varphi}^k = \mathbf{A}^k \cdot \boldsymbol{\theta}^k \tag{4}$$

After executing the command the actual reading of the joints angles, $\boldsymbol{\theta}'^k$ is sensed back. Clearly, if there is an error in the structure matrix, $\mathbf{A}^k$, there will be an error between $\boldsymbol{\theta}^k$ and $\boldsymbol{\theta}'^k$.

$$\mathbf{E}^k = \boldsymbol{\theta}^k - \boldsymbol{\theta}'^k = \begin{pmatrix} \theta_1 - \theta'_1 \\ \theta_2 - \theta'_2 \\ \vdots \\ \vdots \\ \theta_n - \theta'_n \end{pmatrix} = \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ \vdots \\ e_n \end{pmatrix} \tag{5}$$

The update of the control policy is performed by minimizing an objective function in the form:

$$\xi = \frac{1}{2}\mathbf{E}^T\mathbf{E} \tag{6}$$

Minimizing this function implies reducing the difference between $\boldsymbol{\theta}^k$ and $\boldsymbol{\theta}'^k$. The minimization is performed in iterative manner to update the elements of system model, $\mathbf{B}^k$. The update is based on the first order gradient descent method [23]. That is:

$$b_{ij} = b_{ij} - \alpha \frac{\partial \xi}{\partial b_{ij}} \tag{7}$$

where

$$\frac{\partial \xi}{\partial b_{ij}} = \frac{\partial \theta_i}{\partial b_{ij}} \cdot \frac{\partial \xi}{\partial \theta_i} = e_i \cdot \phi_j \tag{8}$$

α : Training constant.

Therefore the update function of Equation (7) becomes

$$b_{ij} = b_{ij} - \alpha \cdot e_i \cdot \phi_j \tag{9}$$

Equation (9) represents the changes in the values of the modeling matrix B^k in order to minimize the error. Due to practical consideration another scaling factor is added to the equation to minimize the swing in the parameters during learning:

$$b_{ij} = b_{ij} - \frac{\alpha \cdot e_i \cdot \phi_j}{\sum_{i=1}^{n} \phi_i} \tag{10}$$

After updating the matrix B^k, the structure matrix is simply updated using $\mathbf{A}^k = \mathbf{B}^{k-1}$.

The modification in each step minimizes the local immediate error. The correction is intended to minimize the error for the particular input used at step k. A large number of these local policy improvements are likely to lead to a global improvement. As with other learning algorithm there is no guarantee of convergence. The convergence is affected by the selection of the initial matrix $\mathbf{A}^1$ and Learning rate α^k. In summary, the algorithm is shown in Figure (4).

Figure (4) The Learning Algorithm

Initialize:

$\mathbf{A}^1$ =Identity matrix, $\mathbf{B}^1$ = Identity matrix;

Train:

For $k \leftarrow 1 \cdots m$

    $\boldsymbol{\theta}^k = random(0 - 90^o)$. Since the range of angles for each joint is 0-90.

    $\boldsymbol{\varphi}^k = \mathbf{A}^k \cdot \boldsymbol{\theta}^k$

    Send $\boldsymbol{\varphi}^k$ to the actuators.

    Read $\boldsymbol{\theta}'^k$ from joints angles sensors.

    $\mathbf{E}^k = \alpha^k \cdot (\boldsymbol{\theta}^k - \boldsymbol{\theta}'^k)$

    For $i \leftarrow 1 \cdots n$

        For $j \leftarrow 1 \cdots n$

$$b_{ij} = b_{ij} - \frac{\alpha \cdot e_i \cdot \phi_j}{\sum_{i=1}^{n} \phi_i}$$

        End

    End

$\mathbf{A}^k = \mathbf{B}^{k-1}$;

End

4. EXPERIMENTAL RESULTS

A simple setup of an anthropomorphic robot hand was developed to test the performance of the proposed algorithm. This hand consists of four identical fingers and opposable thumb. Each of the four fingers consists of three joints with three degrees of freedom. The thumb consists of four joints with four degrees of freedom. All fingers are connected to the palm to form a structure similar to a human hand. Every joint is controlled by two tendons attached to one servomotor so that when the servomotor rotates the two tendons operate in opposite directions. One of the two tendons relaxes the same amount of length as the other tendon tenses. Each tendons pair has a certain amount of tension to make the fingers rigid in their position.

Figure (5) Parts of the Hand (small pieces of Perspex)

The tendons used are made of steel wires and coated with a thin layer of Teflon. CNC machine is used to cut Perspex sheet (Acrylic glass) into small pieces as shown in Figure (5). These small pieces are glued together to form the hand structure as shown in Figure (6). The weight of the hand is 0.3 kg and its length is 18 cm and its width is 8.5 cm. The hand is actuated by HS-475HB servomotors from Hitec Inc. Potentiometers are used as joint angle sensors. The hand is controlled by a network of microcontrollers. The network consists of one PIC16F876 microcontroller used as a master and five PIC16F877 microcontrollers (one for each finger) used as slaves. The microcontrollers (master and slaves) communicate together via the I2C protocol [24].

Figure (7) shows the system's block diagram. The master communicates with the Personal Computer (PC) via RS232 serial port. The PC that executes the learning algorithm sends the commands to the master microcontroller. Then the master dispatches the commands to the desired slave. The slave applies the commands to the servomotors. It waits for 0.8 second (the time's required for servomotors to execute commands in the worst case), then it reads the joint angles sensors. The slave sends the sensors readings back to the master who sends them to the PC and this concludes a commands cycle.

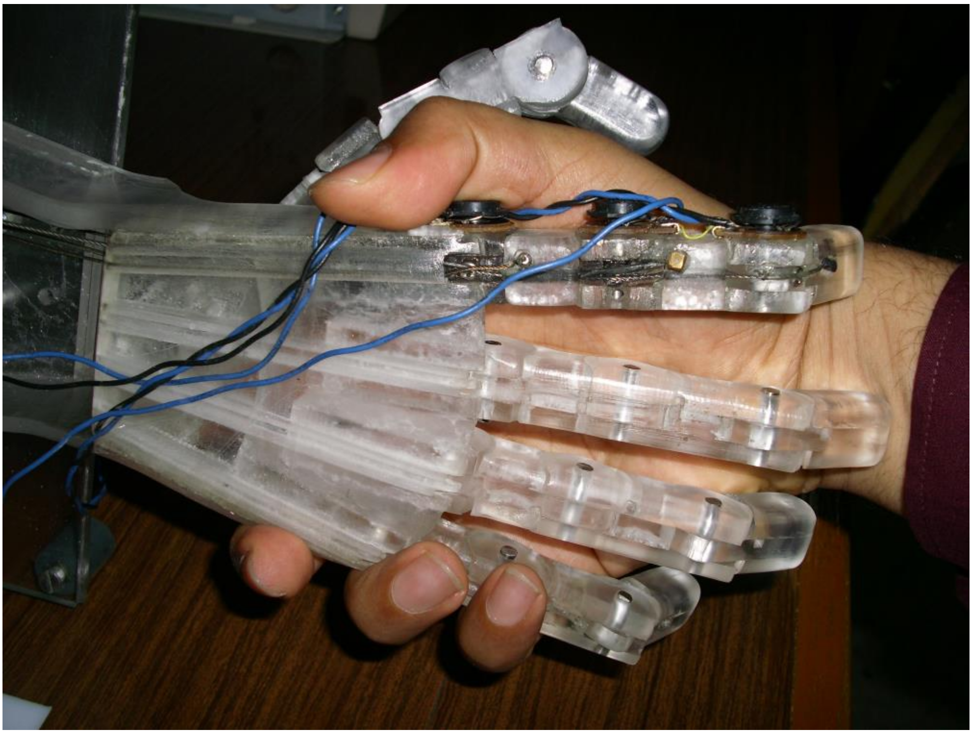


Figure (6) Hand mechanical structure

Figure (8) shows the coefficients of the structure matrix $\mathbf{A}$ as they converge to their final values with the progress of learning. The first plot shows each coefficient alone in separate subplot and the second plot shows all the coefficients combined. To illustrate the effect of learning on the average mapping error, the average error over consecutive periods of 50 cycles is shown in Figure (9). The results show that the average error drops with the increase of learning cycles for all joints. For proximal joint, it begins with about 27º for non-trained system, drops to about 7.7º for the second period and to about 3º for period number 15 (after 700 learning cycles). For middle joint, it begins with about 29º for non-learned system, drops to about 9º for the second period and to about 5.4º for period number 15. For distal joint, it begins with about 30º for non-learned system, drops to about 9.8º for the second period and to about 5.4° for period number 15. The entire learning process of 1000 iteration takes about 15 minutes. The learning factor is 0.01.

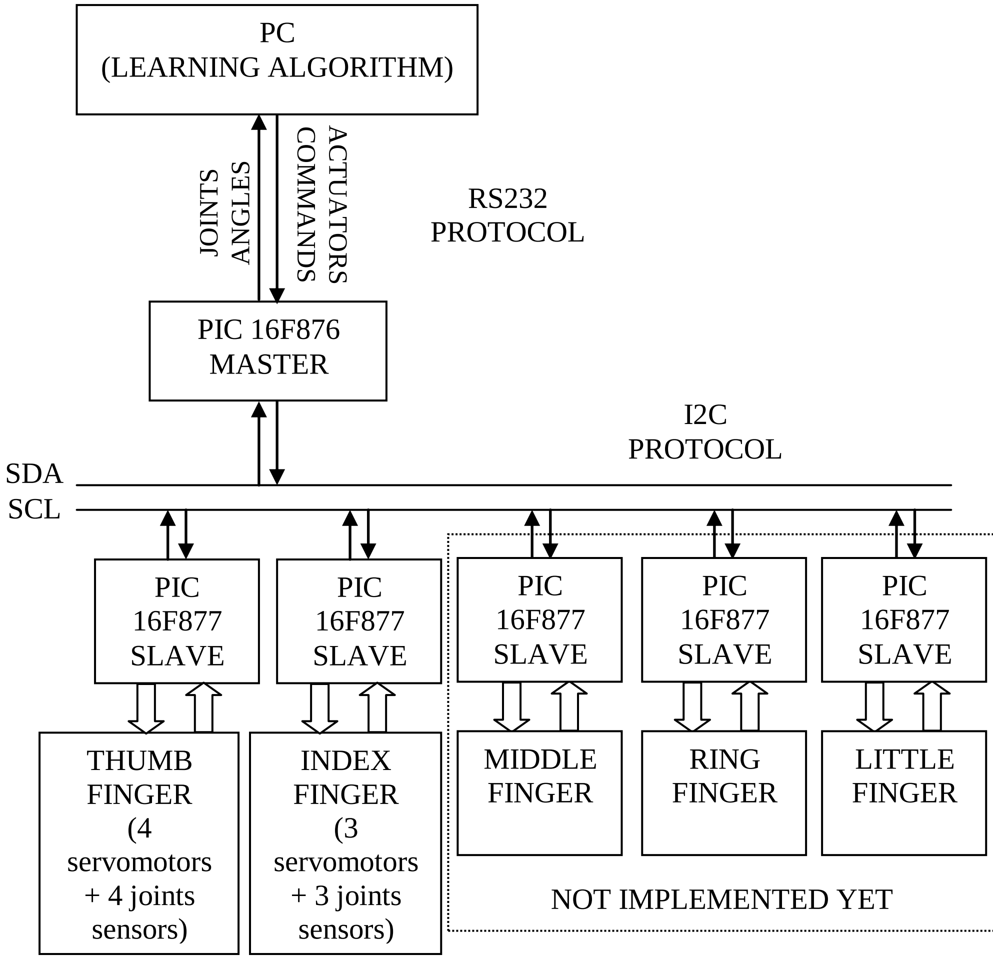


Figure (7) Hardware implementation block diagram

The residual error after training is due to three main reasons:

1- Due the limited quality of the machining of the parts, the tendon paths through internal sheaths and on the surfaces of the hand are not perfectly regular. For example a smaller shift in the location of the axis of articulation may have a nonlinear effect in the length of a tendon path. This effect could not be eliminated using the proposed linear compensation system.
2- The utilized servo motors use a proportional error controller. Such simple control strategy leaves a considerable residual error especially when a large part of the actuator output force is to overcome friction.
3- A minimum backlash is observed in the system due to servomotors internal plastic gears and the simple tendon-actuator coupling.

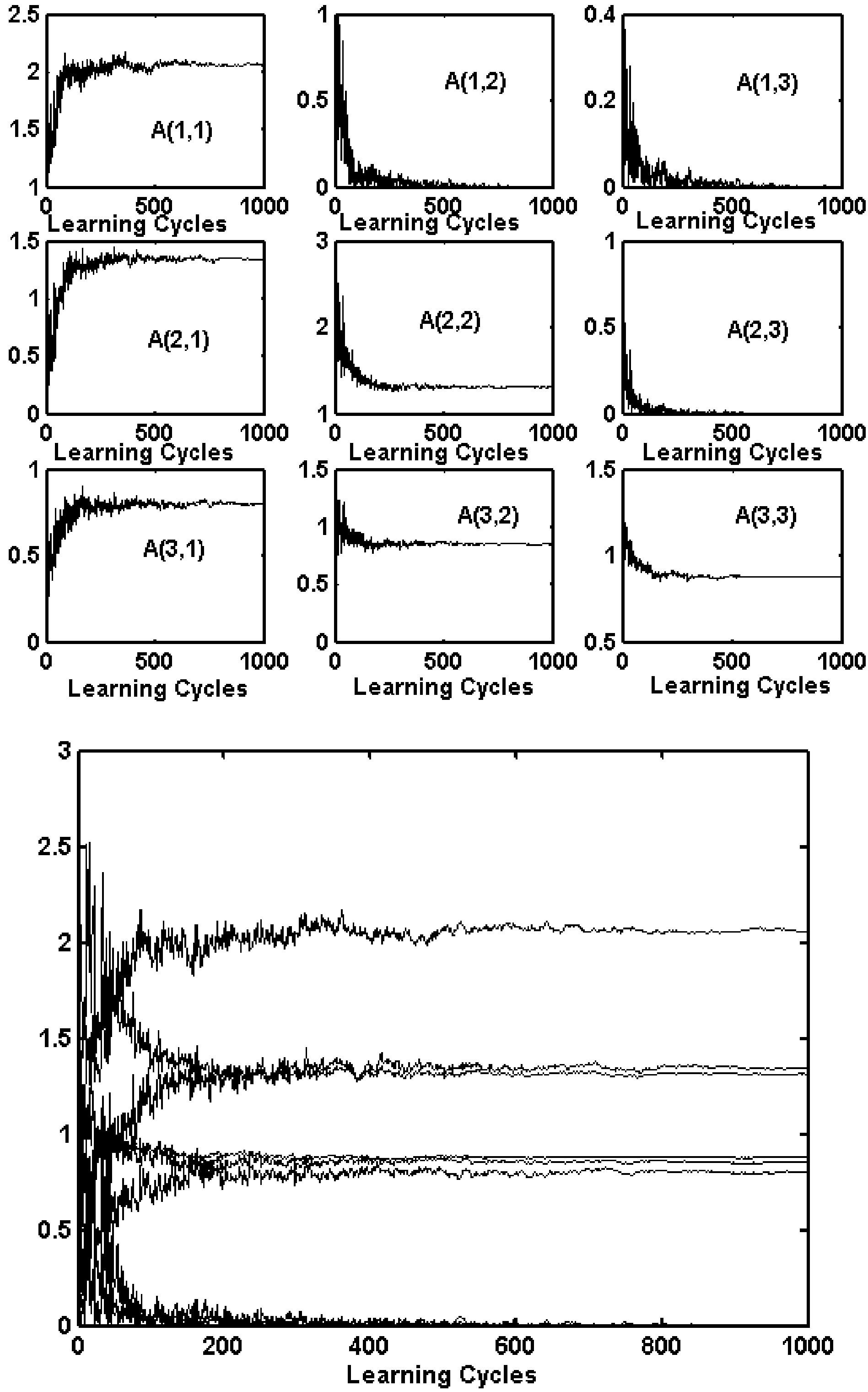
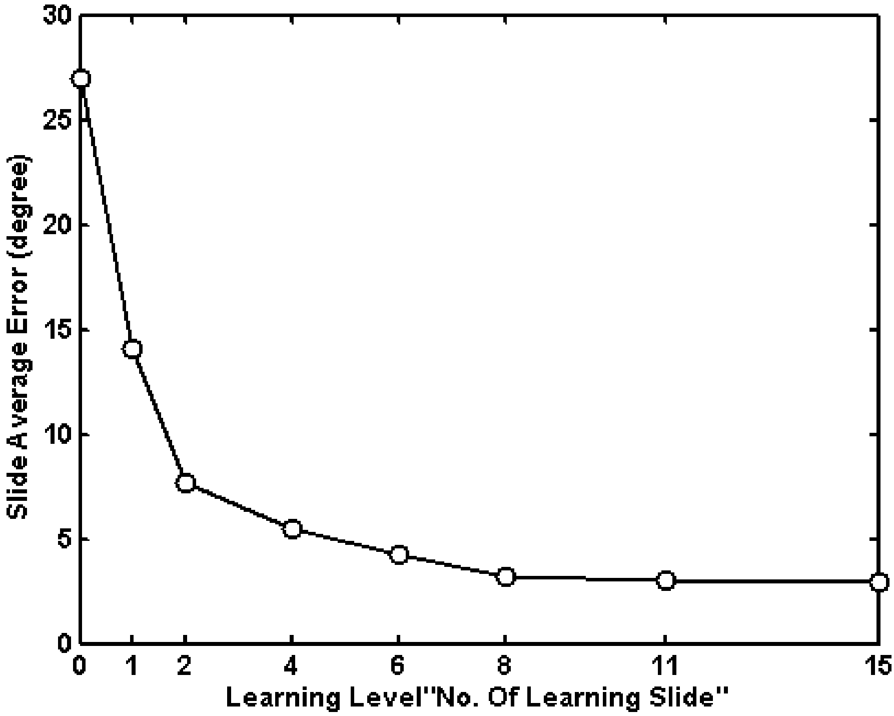
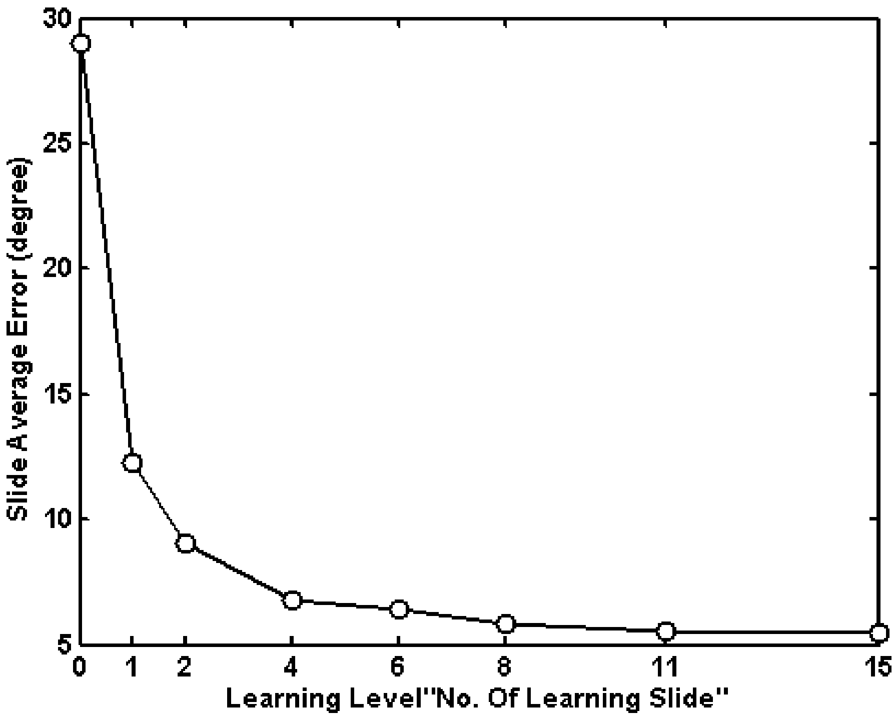


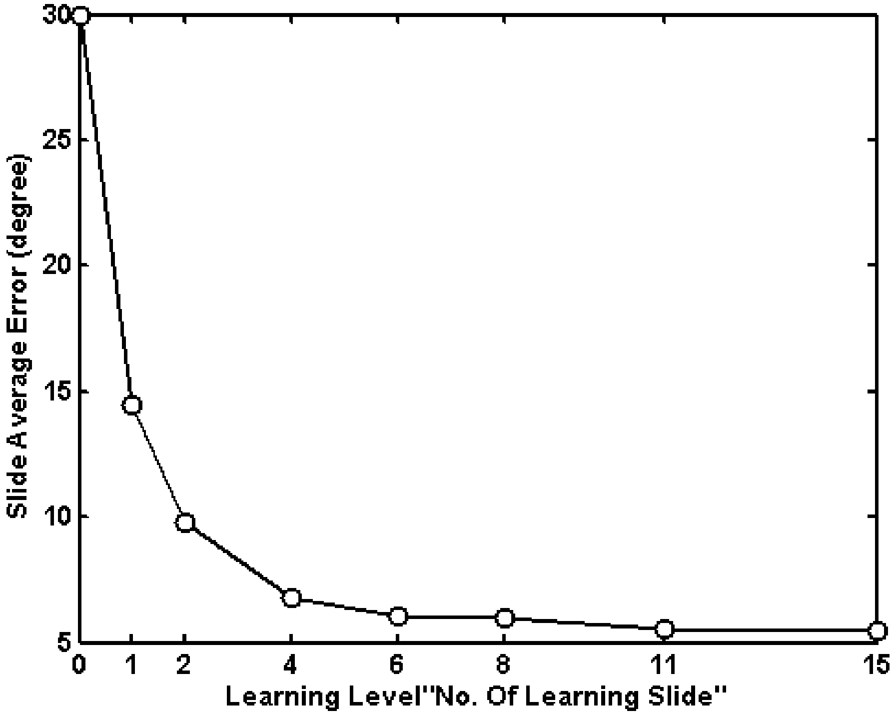
Figure (8) Coefficients of structure matrix **A** during learning



(a) Proximal joint



(b) Middle joint



(c) Distal joint

Figure (9) The effect of learning on slide average error (over 50 learning cycles) before learning and during different levels of learning

5. CONCLUSIONS

In tendons-driven anthropomorphic robot hand, routing tendon around joints causes an operation dependency problem. Instead of the bulky mechanical solutions this work proposes solving the problem at the software level. The foundation of the proposed solution is finding the correct mapping between the joint angles and the actuator angles. This mapping should account for all the dependencies in the system. Using this mapping the operation of the hand is a straight forward arithmetic substitution. Under carefully selected design constrains the mapping is simply a set of linear equations. These equations are represented in a matrix form that is called the structure matrix. Due to the complexity of tendon paths through the structure, the analytical methods for getting the structure matrix are complex. This work proposes finding the matrix using reinforcement learning. In this learning scheme the controller interact with the real physical system to iteratively enhance this matrix.

To test the proposed method a simple anthropomorphic robot hand was developed. The test was done on the index finger. The test results show that the coefficients of the structure matrix converge to their optimal values as the learning progress. Also the experimental testes show that the average operation error reduces with the increase of the number of training cycles.

REFERENCES

1. C.S. Lovchik and M.A. Diftler, “The Robonaut Hand: a Dexterous Robot Hand for Space”, in Proc. IEEE Int. Conf. on Robotics and Automation ’99, ICRA 99, Michigan, pp. 907-912 vol. 2, May 1999.
2. S.C. Jacobsen et al., "Design of the Utah/MIT dexterous hand", In Proc. IEEE International Conference on Robotics and Automation, ICRA’86, pp. 1520-1532, 1986.
3. S. Schultz, C. Pylatiuk and G. Bretthauer , “A New Ultralight Robotic Hand”, in Proc. IEEE Int. Conf. on Robotics and Automation ’01, ICRA 01, Seoul, Korea, pp. 2437-2441 vol. 3, May 21-26,2001.
4. L. Biagiotti, F. Lotti, C. Melchiorri, P. Tiezzi, and G. Vassura., "Ubh3: an anthropomorphic hand with simplified endo-skeletal structure and soft continuous fingerpads", In Proc. IEEE Int. Conf. on Robotics and Automation, 2004.
5. L. Biagiotti, F. Lotti, G. Palli, P. Tiezzi, G. Vassura, C. Melchiorri," Development of UB Hand 3: Early Results", ICRA’05, IEEE Int. Conf. on Robotics and Automation, Barcelona, Spain, pp. 4488-4493, April 18 –22, 2005.
6. J. Butterfass, M. Grebenstein, H. Liu, G. Hirzinger, "DLR-Hand II: Next generation of a dexterous robot hand," In Proceedings of the 2001 IEEE International conference on Robotics & Automation, pp. 109-114, 2001.
7. J. Butterfaß, M. Fischer, M. Grebenstein, S. Haidacher and G. irzinger, "Design And Experiences With DLR Hand II" ,ISORA028, World Automation Congress Tenth International Symposium on Robotics with Applications. Seville, Spain, pp. 105-110 vol. 15, June 28th-July 1st, 2004.
8. X.H. Gao, M.H. Jin, L. Jiang, Z.W. Xie, P. He, "The HIT/DLR dexterous hand: work in progress", Proceedings of the 2003 IEEE International Conference on Robotics & Automation, Taipei, Taiwan, pp. 3164-3168, 2003.
9. Tetsuya Mouri, Haruhisa Kawasaki, Keisuke Yoshikawa, Jun Takai, and Satoshi Ito, "Anthropomorphic Robot Hand: Gifu Hand III", Proc. of Int. Conf. ICCAS2002, October 16 - 19, Muju Resort, Jeonbuk, Korea, pp.1288-1293, 2002.
10. F. Lotti and G. Vassura. "A novel approach to mechanical design of articulated finger for robotic hands", In Proc. IEEE/RSJ IROS Int. Conf. on Intelligent Robot and Systems, pp. 1687-1692 vol. 2,2002.
11. Luigi Biagiotti, "ADVANCED ROBOTIC HANDS:DESIGN AND CONTROL ASPECTS," Ph.D. Thesis, DEIS - DIEM, University of Bologna, 2000-2002.
12. Zheng Li, "USING ROBOTIC HAND TECHNOLOGY FOR THE REHABILITATION OF RECOVERING STROKE PATIENTS WITH LOSS OF HAND POWER," Master Thesis, Faculty of North Carolina State University, November 3, 2003.
13. A. Nahvi, J.M. Hollerbach, Y. Xu, and I.W. Hunter, "An investigation of the Transmission System of a Tendon Driven Robot Hand" Proceedings of

the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems apos;94. apos; Advanced Robotic Systems and the Real World apos;, IROS apos;94., pp. 202 - 208 vol.1, 12-16 Sep 1994.

14. Kaneko, M. Yamashita, T. Tanie, K. "Basic Considerations on Transmission Characteristics for Tendon Drive Robots", 'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on Advanced Robotics, pp. 827-832 vol.1, 19-22 Jun 1991.
15. Kaneko, M. Wada, M. Maekawa, H. Tanie, K. "A New Consideration on Tendon-Tension Control System of Robot Hands", Proceedings., 1991 IEEE International Conference on Robotics and Automation, Sacramento, CA, USA, pp.1028-1033 vol.2, 9-11 Apr 1991.
16. A.Caffaz, G.Panin, "Control Strategies for Tendon Driven Devices", ICMT '99, Pusan (Korea), October 1999.
17. Yeong-Jeong. Ou "The design of Tendon-Driven Manipulators with Isotropic Transmission Characteristics ", Ph.D. Thesis, University of Maryland 1994.
18. Ludvig,L. "Some design problems of tendon-driven, multi-fingered, robot grippers", Innovative Actuators for Mechatronic Systems, IEE Colloquium on, pp. 12/1-12/4, 18 Oct. 1995.
19. http://www.caip.rutgers.edu/~bouzit/lrp/hand.html 30/1/2007
20. Daniele LOIACONO, "Evolving Rules with XCSF: Analysis of Generalization and Performance", Master thesis, Politecnico di Milano, 2003-2004.
21. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning:An Introduction", A Bradford Book, The MIT Press, Cambridge, Massachusetts, London, England,2005.
22. Moriarty, D., Schultz, A., & Grefenstette, J. "Evolutionary algorithms for reinforcement learning", Journal of Artificial Intelligence Research, pp. 241-276 Vol. 11, 1999.
23. Edwin K. P. Chong & Stanislaw H. Zak, "An Introduction to Optimization", A Wiley-Interscience Publication, JOHN WILEY & SONS, INC., New York / Chichester / Weinheim / Brisbane / Singapore / Toronto, pp. 113-134, 2001.
24. **www.microchip.com**, "I2C™ Master Mode Overview and Use of the PICmicro® MSSP I2C Interface with a 24xx01x EEPROM", v 0.40., 22/10/2002.

التعلم بالتعزيز لتصميم يد الروبوت التي تعمل بالأوتار والشبيهة بالإنسان

من المشاكل الشائعة في يد الروبوت التي تعمل بالأوتار مشكلة اعتماد حركة بعض المفاصل على أوتار مفاصل أخرى حيث أن مرور أوتار المفاصل التي في المقدمة حول المفاصل التي تليها متجهة نحو المشغلات يجعل حركة كل مفصل معتمدة ليس فقط على الأوتار التي تحرك المفصل ولكن أيضا على الأوتار التي تمر حول المفصل. أي أن طول مسار الأوتار لبعض المفاصل يعتمد على زوايا مفاصل

أخرى. و الحلول الميكانيكية السابقة لهذه المشكلة تعتمد على استخدام إما البكر لفك الترابط بين حركة المفاصل أو الأغماد الميكانيكية ذات الطول الثابت والتي لا تعاني من المشكلة. ومن عيوب استخدام البكر التعقيد و زيادة الحجم و الوزن. أما الأغماد الميكانيكية فتترك حرة غير مثبتة في هيكل الروبوت لكي تتمكن من المحافظة على طولها وهذا غير مقبول لتطبيقات معينه مثل استخدامه في غرف العمليات وأبحاث الفضاء. كما أن هذا الحل لا يشبه المقابل في اليد البشرية ففي الأغماد البيولوجية يكون طول مسار الأوتار متغير أثناء حركة اليد بعكس الأغماد الميكانيكية ويتم حل مشكلة الاعتمادية بين المفاصل على مستوى التحكم المركزي في المخ.

لذلك فإن هذا البحث يقترح حل المشكلة بطريقة اقرب للحل البيولوجي وهو الاعتماد على نظم التحكم الذكية لحل المشكلة و نواة الحل المقترح هو إيجاد مصفوفة الهيكل التي تحول زوايا المفاصل المطلوبة إلى زوايا المحركات الصحيحة آخذة بعين الاعتبار الاعتمادية بين المفاصل ولكن التحليل الهندسي لمسارات الأوتار لإيجاد هذه المصفوفة عملية صعبة نظرا لتعقيد مسارات الأوتار داخل الأصابع وحول المفاصل وكثرة درجات حرية الحركة والتي قد تصل إلى اثنين وعشرين حركة حرة. لذلك يقترح البحث حل المشكلة عن طريق التعلم وحيث أن استخدام التعلم الذي يعتمد على وجود نموذج رياضي للنظام يتطلب نفس التعقيد من التحليل والحساب فإن البحث يقترح استخدام التعلم عن طريق التفاعل مع النظام الحقيقي والذي يسمى التعلم بالتعزيز .وفيه يتم استخدام الفرق (الخطأ) بين زوايا المفاصل المراد تحقيقها والزوايا الحقيقية التي تم إنجازها لتعديل مصفوفة الهيكل في الاتجاه الذي يقلل من هذا الخطأ بالنسبة للتجربة الحالية أملا في أن تؤدي هذه التعديلات المحلية إلى التقارب العام نحو القيم الصحيحة لمصفوفة الهيكل. وكبقية أنظمة التعليم الأخرى التقارب العام نحو القيم الصحيحة لمصفوفة الهيكل غير مضمون. ولغرض التقييم والتحقق العملي فقد تم تطوير يد روبوت بسيطة شبيهة بالإنسان وتعمل بالأوتار لفحص أداء خوارزمية التعلم المقترحة. وتم تسيير الأوتار في أغماد محفورة داخل الأصابع وعلى سطح اليد لتشبه وتقوم تقريبا بنفس الوظيفة التي تقوم بها الأغماد البيولوجية في اليد البشرية. كما تم بناء شبكة متحكمات مايكروئية للتحكم في يد الروبوت تتكون هذه الشبكة من سيد وخمسة خدم وتتصل هذه الشبكة من خلال السيد مع كمبيوتر شخصي ينفذ خوارزمية التعلم.