Iranian Journal of Medical Physics
Vol. 12, No. 4, Autumn 2015, 251-261
Received: May 17, 2015; Accepted: October 6, 2015

Original Article

A Hybrid Neural Network Approach for Kinematic Modeling of a Novel 6-UPS Parallel Human-Like Mastication Robot

Hadi Kalani[1,2], Alireza Akbarzadeh[1,2*], Sahar Moghimi[1,3]

Abstract

Introduction
we aimed to introduce a 6-universal-prismatic-spherical (UPS) parallel mechanism for the human jaw motion and theoretically evaluate its kinematic problem. We proposed a strategy to provide a fast and accurate solution to the kinematic problem. The proposed strategy could accelerate the process of solution-finding for the direct kinematic problem by reducing the number of required iterations in order to reach the desired accuracy level.

Materials and Methods
To overcome the direct kinematic problem, an artificial neural network and third-order Newton-Raphson algorithm were combined to provide an improved hybrid method. In this method, approximate solution was presented for the direct kinematic problem by the neural network. This solution could be considered as the initial guess for the third-order Newton-Raphson algorithm to provide an answer with the desired level of accuracy.

Results
The results showed that the proposed combination could help find a approximate solution and reduce the execution time for the direct kinematic problem, The results showed that muscular actuations showed periodic behaviors, and the maximum length variation of temporalis muscle was larger than that of masseter and pterygoid muscles. By reducing the processing time for solving the direct kinematic problem, more time could be devoted to control calculations.. In this method, for relatively high levels of accuracy, the number of iterations and computational time decreased by 90% and 34%, respectively, compared to the conventional Newton method.

Conclusion
The present analysis could allow researchers to characterize and study the mastication process by specifying different chewing patterns (e.g., muscle displacements).

Keywords: Kinematic Problem, Mastication Robot, Neural Networks, Newton-Raphson Method

1- Center of Excellence on Soft Computing and Intelligent Information Processing (SCIIP)
2- Department of Mechanical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran
*Corresponding author: Tel: +98 51 38805011, E-mail: *ali_akbarzadeh@um.ac.ir*
3- Department of Electrical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

1. Introduction

Mastication is a complex process which is driven by a complex assembly of more than twenty muscles working as an ensemble. These muscles are placed on both sides of the mandible. To investigate the dynamic behavior of chewing in humans, a robotic device is required. Various researchers have investigated the mastication system, its modeling, and robotics [1-7]. To understand the natural trajectory of mastication, the jaw movements must be tracked. For the first time, Posselt recorded the human jaw motions in the mid 1950s [1], describing an envelope of maximum mandibular movements. Moreover, Pap and Bronlund reported the maximum displacement and angular movements of the mandible [2]. Röhrle et al. also measured the three-dimensional (3D) coordinates of the incisor point (IP) and three angular motions of the mandible [3]. In addition, Koolstra and Eijden developed a biomechanical model of the jaw [4].

Many researchers have investigated the kinematics of parallel manipulators. In general, kinematic problems can be regarded as either direct or inverse. Based on the literature, the inverse kinematic problem (IKP) of a parallel robot, unlike a serial robot, is often simpler to solve than the corresponding direct kinematic problem (DKP) [8-20].

The goal in IKP is to determine the value of an actuated joint, leg length, or joint angle, based on the given position and orientation of the mobile platform (e.g., the mandible). In contrast, DKP seeks to determine the position and orientation of the mobile platform, based on the given value of the actuated joint, leg length, or joint angle. The former problem is usually considered in control applications, whereas the latter is often utilized for simulation purposes [8, 9].

Newton's method is widely employed to solve the DKP of parallel robots [21-26]. The initial guess highly impacts the number of iterations, needed for solution-finding and even the convergence of the process. Pratik and Lam presented a novel strategy to provide an appropriate initial guess for the standard Newton–Raphson technique by using neural networks [27]. In fact, many researchers have employed the artificial neural network (AAN) approach to solve DKPs, so far [28-32].

The multi-layer perceptron AAN (MLPANN) is a universal approximation for non-linear input-output mapping. The masticatory system can be modeled by a parallel robot, where the skull, mandible, and muscles are represented by the fixed base, the moving platform, and linear actuators, respectively. Although more than 20 muscles are involved in the process of human mastication, only six play a major role [33].

Numerous studies have evaluated the performance of temporomandibular joint (TMJ), using the trajectory of condylar paths [34]. Previous studies have presented 2D and 3D characterizations of TMJ compound movements. Spoor and Gallo et al. used the screw axis model (also known as the finite helical axis model) to present a 3D model of TMJ movements [35, 36]. Also, Grimes et al. employed a tracking system and a program with the screw displacement axis model to perform a mathematical 3D analysis of an object movement [37]. They calculated the rotation, translation, and 2D and 3D charting of the condylar path.

In this study, we first developed the robot kinematics and then proposed an algorithm to solve the forward kinematic problem of the mastication robot. By introducing a novel method, we aimed to accelerate the solution-finding process for FKP of the mastication robot. To achieve this goal, by combining AAN and the third-order numerical algorithm, an improved hybrid strategy was proposed for direct kinematic analysis of parallel manipulators.

In this method, a near-exact solution for DKP was produced by the neural network. This solution was then considered as the initial guess for the third-order numerical technique, which solved the non-linear direct kinematic equations to provide an answer with the desired level of accuracy.

In the present study, the proposed mastication robot is described in details in the system

description section. The mathematical modeling, as well as the inverse and direct kinematic formulations for the robot, is presented in the kinematic model section, which also covers the details of the proposed improved hybrid method. In the section of tracking jaw movements, the actual chewing trajectories of three male subjects and the recording process are discussed. The results and discussion sections present the simulation results of the biomechanical model. Finally, some concluding remarks are made in the conclusion section of the article.

2. Materials and Methods

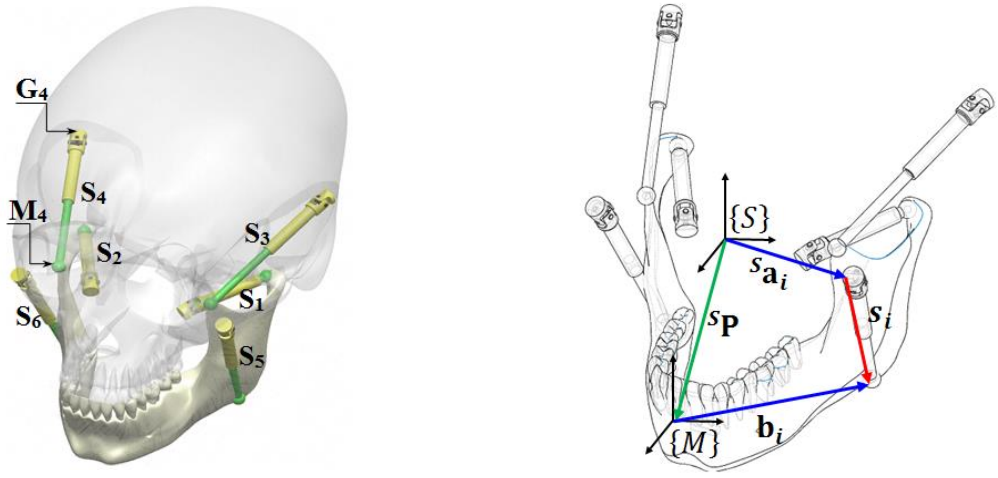
2.1 . *System description*

Although more than 20 muscles are involved in the process of human mastication, only six of these muscles play a major role [6, 38]. To represent the geometry of human mastication, we proposed the use of a general 6-universal-prismatic-spherical (UPS) Stewart–Gough platform. The mobile and stationary platforms represented the human mandible and skull, respectively, and the actuators denoted jaw muscles (schematic presentation in Figure 1).

- We selected a general 6-UPS parallel mechanism rather than the previously recommended spherical-prismatic-spherical (SPS) joints [6] for several reasons. Firstly, UPS joints model the muscle motion more realistically than SPS joints. Also, 6-SPS provides six additional passive degrees of freedom (DOF), which allow in-place rotation of the leg around its own axis. However, this leg motion does not represent a realistic motion of the jaw muscle which does not rotate along its axial axis.

Secondly, according to Chebychev–Grübler–Kutzbach criterion [6], 6-UPS and 6-SPS both offer six DOFs. Thirdly, the six additional passive DOFs in the legs in 6-SPS do not affect the kinematic input-output relationships of the manipulator, although they increase the size of dynamic equations [39]. Although 6-UPS has six differential equations, 6-SPS must have 12 equations to account for the six additional passive DOFs.

Actuators S_1 to S_6 in Figure 1(a) represent the lateral pterygoid, temporalis, and masseter muscles, respectively, while G_i and b_i (i=1, 2 ..., 6) represent the connecting locations of these muscles relative to the skull and mandible. The muscle actuators are connected to the spherical and universal joints at the mandible and skull. It should be mentioned that the dimensions were selected, based on the available literature [6, 7].



(a) (b)

Figure 1. (a) The mandible, actuators, and connecting points, (b) the schematic diagram of the 6-UPS mastication robot

2.2. *The kinematic model*

In this section, we applied vector algebra to obtain the closed-form equations for the kinematics of the human-like mastication robot. First, the inverse kinematic formulations were derived, and the closed-form solutions were obtained. Next, the direct kinematic formulations and their solutions, obtained via the improved hybrid method, were presented. The kinematic models and the provided solutions could enable us to investigate the relationship between the length and orientation of muscles during different mandibular movements.

2.2.1 Inverse kinematics

Inverse kinematics present the motion properties of the actuator, such as displacement, velocity, and acceleration, based on the IP trajectory during mastication. The configuration of the mastication robot was specified by the position and orientation of the system {**M**} with respect to {**S**} (Figure 1b). For each leg, a vector loop equation could be written as follows:

$$ {}^{S}\mathbf{a}_i + {}^{S}\mathbf{s}_i = {}^{S}\mathbf{P} + {}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_i \quad (i = 1, 2, .., 6) \tag{1}$$

where ${}^{S}\mathbf{P}$ and ${}^{S}\mathbf{a}_i$ are the position vectors of IP and mandible joint with respect to the skull coordinate system, respectively. Also, ${}^{M}\mathbf{b}_i$ represents the position vector of the ball joint with respect to the coordinate system of the mandible. The magnitude of s_i represents the actuated muscle length.

The orientation of the mandible system {**M**} with respect to skull coordinates {**S**}, referred to as the mandible's rotation matrix ${}^{S}_{M}\mathbf{R}$, was defined, using roll, pitch, and yaw angles (γ, β, and α), as shown in Equation (2):

$$ {}^{S}_{M}\mathbf{R} = \begin{bmatrix} c\gamma\ c\beta & c\gamma\ s\beta\ s\alpha - s\gamma\ c\alpha & c\gamma\ s\beta\ c\alpha + s\gamma\ s\alpha \\ s\gamma\ c\beta & s\gamma\ s\beta\ s\alpha + c\gamma\ c\alpha & s\gamma\ s\beta\ c\alpha - c\gamma\ s\alpha \\ -s\beta & c\beta\ s\alpha & c\beta\ c\alpha \end{bmatrix} \tag{2}$$

Where $c\beta = \cos(\beta)$, $s\beta = \sin(\beta)$, and so on. According to Equation (1), we could obtain Equation (3):

$$ \begin{Bmatrix} s_x \\ s_y \\ s_z \end{Bmatrix} = \begin{Bmatrix} P_x \\ P_y \\ P_z \end{Bmatrix} + {}^{S}_{M}\mathbf{R} \begin{Bmatrix} b_x \\ b_y \\ b_z \end{Bmatrix} - \begin{Bmatrix} a_x \\ a_y \\ a_z \end{Bmatrix} \tag{3}$$

Therefore, the lengths of the actuators could be obtained as follows:

$$ S_i = \|\mathbf{s}_i\| = \sqrt{s_{ix}^2 + s_{iy}^2 + s_{iz}^2} = \sqrt{{}^{S}s_i^T\ {}^{S}s_i} \quad (i = 1,2, \dots ,6) \tag{4}$$

Equation (4) could provide a solution to IKP. Position vectors ${}^{S}\mathbf{a}_i$ and ${}^{M}\mathbf{b}_i$ were fixed and established in the IKP. Therefore, by specifying the trajectory of the mandible, and therefore, having ${}^{S}\mathbf{P}$ and ${}^{S}_{M}\mathbf{R}$, Equation (4) could provide the required lengths of the six muscles.

2.2.2. The improved hybrid method for direct kinematics

The aim of direct kinematic approach was to determine the position and orientation of the mobile platform on the basis of the given leg lengths. Overall, the DKP of parallel robots is quite complicated, and the few available analytical approaches are not suitable for real-time control.

In this study, we described a strategy for solving DKP, which consisted of MLPANN and the third-order Newton-Raphson method [27, 40, 41] (Figure 2). With this strategy, referred to as the "improved hybrid method", MLP was first trained and then used to provide the initial guess, employed in the Newton-Raphson method. By using the improved hybrid method, the direct kinematic solution of the mastication robot could be obtained with low computational time and high accuracy. Each component of the hybrid method will be discussed in the proceeding paragraphs.

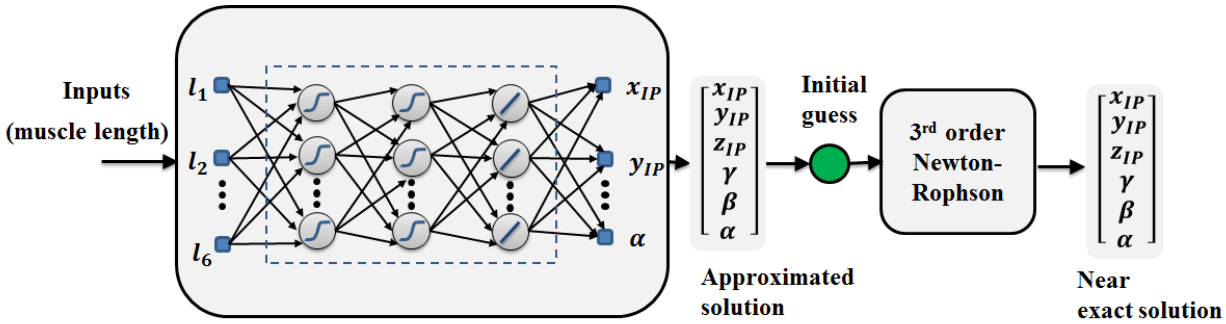


Figure 2. The improved hybrid method

2.2.2.1. *MLPANN*

To reduce the simulation time, an MLPANN structure was adopted for producing the initial guess and implementing the Newton-Raphson method [27]. Detailed descriptions of MLPANN can be found in the literature [28-32]. The leg lengths (muscle lengths) of the mastication robot, $l_i (i = 1,2,\dots,6)$, and the position and orientation of the mandible $[x_{IP}, y_{IP}, z_{IP}, \gamma, \beta, \alpha\,]$, were considered as the inputs and outputs of the network, respectively.

The inverse kinematic approach, describing the system, was used to produce data for training the network. First, an approximate workspace of the human mastication movements was specified. Next, a random sample of points (positions and orientations) in the workspace was selected, and the corresponding leg lengths were obtained. Finally, the obtained leg lengths and the Cartesian space parameters of the mandible were used as the inputs and outputs for training the MLPANN, respectively.

In this study, MLPANN consisted of two layers, with 15 and 20 neurons in the first and second layers, respectively. All neurons had a sigmoid activation function, except for those in the output layer with a linear activation function. The Levenberg Marquardt (LM) algorithm was used for back-propagation training of the network.

2.2.2.2. *Newton–Raphson method*

The Newton–Raphson method yields accurate results, given the sufficient number of algorithm iterations [40, 41]. For a system of multiple equations and variables, the Newton-Raphson method presents the following equation:

$$\mathbf{X}_{m+1} = \mathbf{X}_m - \left(\frac{\partial \mathbf{F}(\mathbf{X}_m)}{\partial \mathbf{X}_m}\right)^{-1} \mathbf{F}(\mathbf{X}_m) \qquad (5)$$

where $\mathbf{X}$ is a vector of the variables to be estimated, $\mathbf{F}$ is a vector function, which approaches zero during the iterative process, and m is denotes the iteration number. For the mastication robot, we defined:

$$\mathbf{X}^T = [x_{IP}, y_{IP}, z_{IP}, \alpha, \beta, \gamma\,] \qquad (6)$$

and

$$\mathbf{F}(\mathbf{X}) = \begin{bmatrix} \left\| {}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_{1i} + {}^{S}\mathbf{P} - {}^{S}\mathbf{a}_1 \right\| - l_1 \\ \vdots \\ \left\| {}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_{6i} + {}^{S}\mathbf{P} - {}^{S}\mathbf{a}_6 \right\| - l_6 \end{bmatrix}_{6\times 1} = \begin{bmatrix} \left\| {}^{S}\mathbf{s}_1 \right\| - l_1 \\ \vdots \\ \left\| {}^{S}\mathbf{s}_6 \right\| - l_6 \end{bmatrix}_{6\times 1} \qquad (7)$$

where $\left\| {}^{S}\mathbf{s}_i \right\|$ and l_i are the estimated and actual lengths of the actuated muscle i, respectively. By differentiating (7), we obtained Equation (8):

$$\frac{\partial \mathbf{F}(\mathbf{X}_m)}{\partial \mathbf{X}_m} = \mathbf{J}_{6\times 6}\,(x_{IP}, y_{IP}, z_{IP}, \gamma, \beta, \alpha) \qquad (8)$$

where $\mathbf{J}_{6\times 6}$ is the Jacobian matrix, obtained by differentiating both sides of Equation (4):

$$\dot{S}_i = \frac{{}^{S}s_i^T\ {}^{S}\dot{s}_i}{\sqrt{{}^{S}s_i^T\ {}^{S}s_i}} \qquad (9)$$

For ${}^{S}_{M}\dot{\mathbf{R}} = {}^{S}\boldsymbol{\omega}_{IP} \times {}^{S}_{M}\mathbf{R}$ and $\mathbf{A}\cdot(\mathbf{B}\times\mathbf{C}) = (\mathbf{C}\times\mathbf{A})\cdot\mathbf{B}$, we obtained Equation (10):

$$\dot{S}_i = \frac{1}{S_i}\left[\left({}^{S}\mathbf{P}_i + {}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_i - {}^{S}\mathbf{a}_i\right)^T \left({}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_i \times \left({}^{S}\mathbf{P}_i - {}^{S}\mathbf{a}_i\right)\right)^T\right]\begin{bmatrix} {}^{S}\mathbf{v}_{IP} \\ {}^{S}\boldsymbol{\omega}_{IP} \end{bmatrix} \quad (10)$$

where ${}^{S}\mathbf{v}_{IP}$ and ${}^{S}\boldsymbol{\omega}_{IP}$ are the transitional and angular velocities of the mandible, respectively. Based on equations (8) and (10), Equation (11) could be obtained:

$$\frac{\partial \mathbf{F}(\mathbf{X}_m)}{\partial \mathbf{X}_m} = \mathbf{J} = \begin{bmatrix} \frac{1}{S_1}\left[\left({}^{S}\mathbf{P} + {}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_1 - {}^{S}\mathbf{a}_1\right)^T\right] & \frac{1}{S_1}\left[\left({}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_1 \times \left({}^{S}\mathbf{P} - {}^{S}\mathbf{a}_1\right)\right)^T\right] \\ \vdots & \vdots \\ \frac{1}{S_6}\left[\left({}^{S}\mathbf{P} + {}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_6 - {}^{S}\mathbf{a}_6\right)^T\right] & \frac{1}{S_6}\left[\left({}^{S}_{M}\mathbf{R}\ {}^{M}\mathbf{b}_6 \times \left({}^{S}\mathbf{P} - {}^{S}\mathbf{a}_6\right)\right)^T\right] \end{bmatrix}_{6\times 6} \quad (11)$$

The convergence rate of the Newton–Raphson technique is highly dependent on the selected order, with higher orders providing more accurate solutions. However, higher orders require the calculation of multiple-order derivatives (higher than one). A practical approach has been recently proposed for solving systems of non-linear equations, which allows the use of higher orders of Newton–Raphson technique with no need to calculate higher-order derivatives [40, 41]. Consider the non-linear Equation (12):

$$\mathbf{F}(\mathbf{X}) = 0 \quad (12)$$

where $\mathbf{F}^{n\times 1}$ is the non-linear system and $\mathbf{X}$ is a vector of the variables we aim to estimate. By using the following iterative scheme, Darvishi and Barati solved the non-linear system in Equation (12) and showed that this method has a convergence order of three [40]:

$$\mathbf{X}_{m+1} = \mathbf{X}_m - \mathbf{F}'(\mathbf{X}_m)^{-1}\left(\mathbf{F}(\mathbf{X}_m) + \mathbf{F}(\mathbf{X}^*{}_m)\right) \quad (13)$$

$$\mathbf{X}^*{}_{m+1} = \mathbf{X}_m - \mathbf{F}'(\mathbf{X}_m)^{-1}\mathbf{F}(\mathbf{X}_m) \quad (14)$$

where $\mathbf{F}'$ is the Jacobian matrix and $\mathbf{J}_{6\times 6}$, is defined by Equation (15):

$$\mathbf{F}'(\mathbf{X}_m) = \frac{\partial \mathbf{F}(\mathbf{X}_m)}{\partial \mathbf{X}_m} \quad (15)$$

where the stopping criterion is defined as follows:

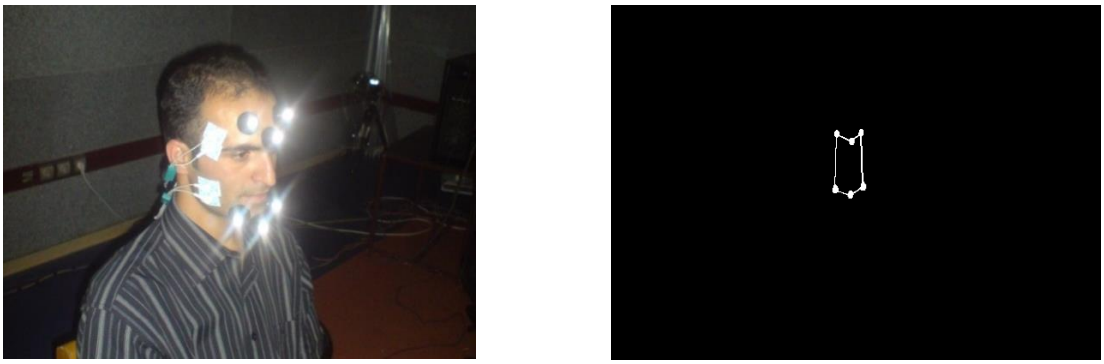
$$\|\mathbf{X}_{m+1} - \mathbf{X}_m\|_\infty < E_{\max} \quad (16)$$

The improved hybrid method presented in this study extends previous research [27] by increasing the order of Newton–Raphson technique to the fifth degree. Specifically, equation (5) was used to construct the hybrid method, and equations (13) and (14) were used to obtain the improved hybrid method. All kinematic codes were implemented in MATLAB R2010b.

2.3 *Tracking jaw movements*

To obtain the chewing trajectory, three male subjects (age range: 24–28 years) were selected. All subjects were informed about the procedure, and written informed consents were obtained prior to the experiment, which consisted of three sessions of six mandibular opening and closing movements. The Ethics Committee of Ferdowsi University of Mashhad approved this study.

To track the jaw motion, small reflective markers, nearly 10 mm in diameter, were adhered to specific facial locations (Figure 3). Forehead markers were used as reference points. Tracking was performed, using a Simi Reality Motion System GmbH (Germany). The recorded data were pre-processed prior to modeling, using the same system [35]. The camera output was digitized to 250 frames per second, and frequencies above 7 Hz were discarded.



(a) (b)

Figure 3. (a) Marker position on the subject's face, (b) the two-dimensional reconstruction of the marker set

3. Results

3.1. Inverse kinematics

Figure 4 shows the time-dependent 3D trajectory of the mandible during mastication in subject No. 2.Figure 5 demonstrates the time-varying behaviors of muscle lengths in two subjects; this information was used to reproduce the chewing pattern. The temporalis muscles experienced larger length changes, compared to masseter and pterygoid muscles in all three subjects. The behavior of the muscles could be approximated by harmonic series functions.

3.2. Direct kinematics: Results obtained by the improved hybrid method

In this section, we solved DKP, using the improved hybrid technique and compared its performance with the Newton–Raphson method. The solution for DKP allowed us to have muscle lengths as the input and to obtain the position and orientation of the mandible. This information would enable us to characterize the behavior of muscles from a different perspective than the IKP solution. Additionally, this solution could be used for simulation and control purposes.

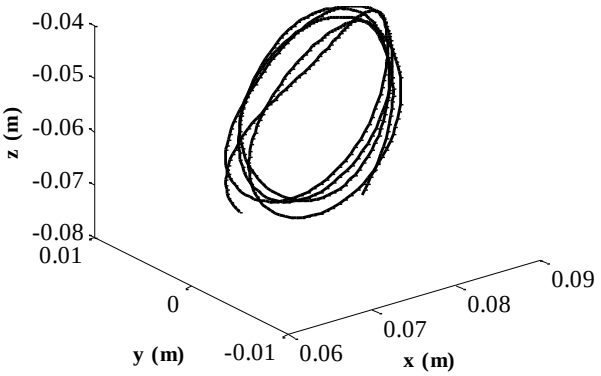
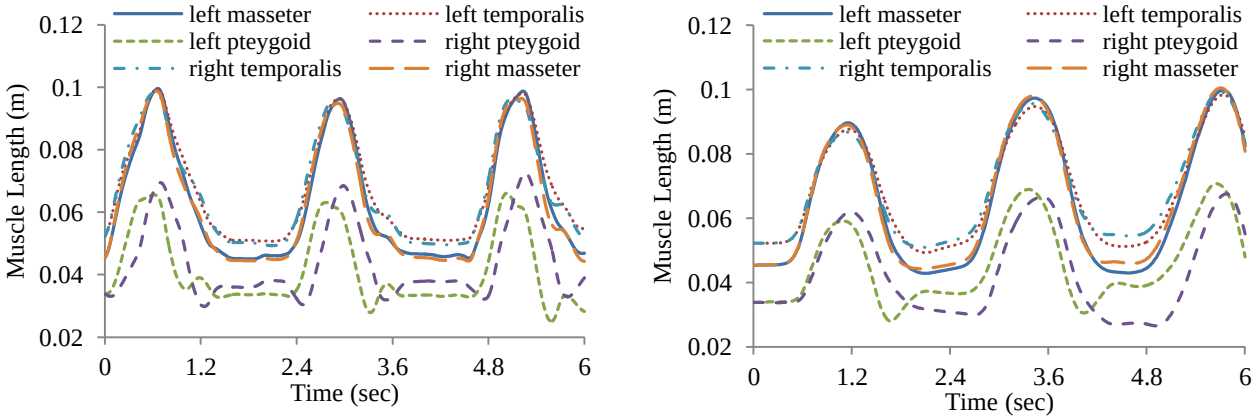


Figure 4. The chewing trajectory in subject No. 2



(a) Subject 1 (b) Subject 2

Figure 5. Time-varying lengths of muscles during the chewing pattern

To train MLPANN, 500 randomly configured data points (positions and orientations) were selected within the workspace of mastication motion [6]. These data were considered as the input to the IKP, and the corresponding muscle lengths (output) were obtained. Then, the obtained leg lengths and the Cartesian space parameters of the mandible were used as the inputs and outputs for training MLPANN, respectively. Upon learning, MLPANN was utilized to obtain the initial guess for the improved hybrid method. Performance of the improved hybrid method, evaluated using the recorded trajectory for subject No. 2, is shown in Figure 6. As illustrated, the improved hybrid method could provide a near-exact solution.

To permit a comprehensive discussion, the performance of different iterative methods (i.e., the improved hybrid method, conventional Newton-Raphson method, and hybrid method) were evaluated and compared with four levels of precision. The number of iterations, the maximum allowed error, and computational time are presented in Table 1, where *N* and *t* are the average number of iterations and running time of the corresponding method, respectively.

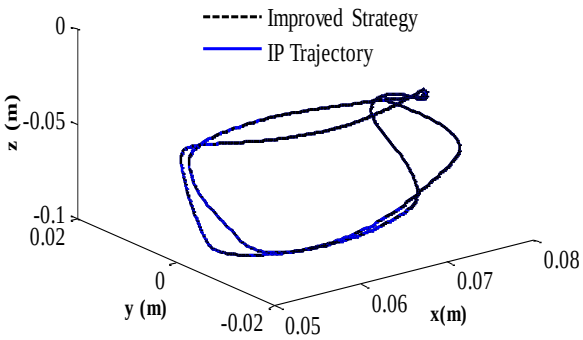


Figure 6. Comparison of the hybrid strategy output with the real IP trajectory. The chewing pattern was obtained from subject No. 2.

As shown in Table 1, faster computational performance was achieved with the second-order hybrid strategy (SOH) for relatively low levels of accuracy (e.g., 10^{-3}m) and the third-order hybrid strategy (TOH) for relatively high levels of accuracy. These two hybrid strategies (i.e., SOH and TOH) showed a better performance than the Newton-Raphson method. Moreover, with TOH strategy, the number of iterations, as well as the running time, decreased, regardless of the level of accuracy.

Table 1. Comparison of various iterative methods

Precision level	Stop criteria	Newton-Rophson method	SOH* strategy	TOH** strategy
1	10^{-3}	N=4.7391 t=1.874 (sec)	N=1.4052 t=0.51 (sec)	N=1.3339 t=0.5919(sec)
2	10^{-6}	N=6.24 t=2.679 (sec)	N=3.9565 t=1.404 (sec)	N=2.8748 t=1.2495 (sec)
3	10^{-10}	N=8.1391 t=2.883 (sec)	N=5.4904 t=1.937(sec)	N=3.6626 t=1.6114 (sec)
4	10^{-15}	N=10.2487 t=3.65 (sec)	N=7.6365 t=2.692 (sec)	N=5.3687 t=2.3293 (sec)

* SOH: Second-order hybrid strategy, ** TOH: Third-order hybrid strategy. SOH strategy is the hybrid strategy, while TOH strategy is the improved hybrid method.

Table 2 shows the percentage of improvements in computational time and number of iterations. For all precision levels, the hybrid strategies showed significant performance improvements, compared to the conventional Newton-Raphson method. For higher levels of accuracy, TOH strategy showed better time performance and fewer iterations, compared to SOH technique.

Table 2. The percentage of performance improvement in hybrid strategies versus the Newton-Raphson method

Precision level	Stop criteria		SOH*	TOH**
1	10^{-3}m	Improvement in *N*	+237%	+255%
		Improvement in *t*	+267%	+216%
2	10^{-6}m	Improvement in *N*	+58%	+118%
		Improvement in *t*	+91%	+114%
3	10^{-10}m	Improvement in *N*	+48%	+122%
		Improvement in *t*	+49%	+78%
4	10^{-15}m	Improvement in *N*	+34%	+90%
		Improvement in *t*	+36%	+56%

* SOH: Second-order hybrid strategy, ** TOH: Third-order hybrid strategy

4. Discussion

In recent years, many efforts have been made to decrease the required time for solving the DKP of parallel robots. To address this issue, two types of approaches, namely analytical [42-43] and numerical approaches [21-26], have been introduced. In general, for real-time control and simulation, analytical approaches are unacceptable. Therefore, it seems necessary to develop methods, based on numerical approaches.

By applying the proposed method, it was demonstrated that replacing the ordinary Newton-Raphson algorithm by its third-order counterpart could lead to a decline in the number of iterations (required to reach the desired accuracy level) and thus, reduce the DKP analysis time. By reducing the processing time allocated to solving DKP, more time could be devoted to control calculations. Therefore, more complicated control algorithms with better performances could be implemented. Moreover, this algorithm could be applied to any DKPs of parallel robots for obtaining near-exact solutions. It should be mentioned that direct kinematic approaches are commonly applied in simulation studies.

The modified hybrid strategy provided a near-exact solution for DKP and was computationally efficient. Specifically, the proposed method improved the accuracy of prediction, compared to the second-order hybrid strategy and the conventional Newton-Raphson method. For relatively high levels of accuracy, the number of iterations and computational time decreased by 90% and 56%, respectively, compared to the conventional Newton-Raphson method. Also, these parameters reduced by 56% and 20%, respectively in the proposed model, compared to the second-order hybrid strategy.

5. Conclusion

In this paper, we provided a framework for the development, analysis, and evaluation of a human-like mastication robot. We proposed the use of a general 6-UPS robot to simulate the human mastication process and presented solutions to IKP and DKP of the model. The DKP of parallel manipulators is usually complicated. In general, a closed-form solution cannot be proposed and often multiple solutions are offered. Therefore, we proposed an improved hybrid method, which was generated by combining the third-order Newton–Raphson method and MLPANN. In this method, MLPANN was used to determine the initial guess for the Newton-Raphson method. The results showed that the muscular actuations show periodic behaviors and the maximum length variation of the temporalis muscle is larger than that of masseter and pterygoid muscles.

This paper contributes to the literature by proposing the use of a general 6-UPS model as the masticatory system and applying vector algebra to obtain its direct and inverse kinematic formulations. Consequently, the closed-form equations of IKP were obtained, and an improved hybrid method was proposed, which combined the third-order Newton-Raphson method with MLPANN to obtain near-exact solutions with high accuracies and low computational time.

Acknowledgments

The work described in this paper was supported by a grant from the research office of Ferdowsi University of Mashhad, Mashhad, Iran. We thank all the subjects who helped us by voluntarily participating in our study.

References

1. Posselt U. Movement areas of the mandible. The Journal of Prosthetic Dentistry. 1957;7(3):375-85.
2. Pap J, Xu WL, Bronlund J. A robotic human masticatory system: kinematics simulations. International Journal of Intelligent Systems Technologies and Applications. 2005;1(1-2):3-17.
3. Röhrle O, Anderson I, Pullan A, editors. From jaw tracking towards dynamic computer models of human mastication. IFBME Proceedings of 12th International Conference on Biomedical Engineering, Singapore; 2005.
4. Koolstra JH, van Eijden TM. A method to predict muscle control in the kinematically and mechanically indeterminate human masticatory system. J Biomech. 2001 Sep;34(9):1179-88.
5. Torrance JD. kinematics, Motion control and force estimation of a chewing robot of 6RSS parallel mechanism. Ph.D dissertation, Massy University, Palmerston North, New Zealand; 2011.
6. Xu W, Bronlund JE. Mastication Robots: Biological Inspiration to Implementation: Springer Berlin Heidelberg; 2010.
7. Van Eijden TM, Korfage JA, Brugman P. Architecture of the human jaw-closing and jaw-opening muscles. Anat Rec. 1997 Jul;248(3):464-74.
8. Patel AJ, Ehmann KF. Calibration of a hexapod machine tool using a redundant leg. International Journal of Machine Tools and Manufacture. 2000;40(4):489-512.
9. Enferadi J, Tootoonchi AA. Accuracy and stiffness analysis of a 3-RRP spherical parallel manipulator. Robotica. 2011;29(02):193-209.
10. Enferadi J, Tootoonchi AA. A novel approach for forward position analysis of a double-triangle spherical parallel manipulator. European Journal of Mechanics-A/Solids. 2010;29(3):348-55.
11. Gosselin C . Kinematics Analysis, Optimization and programming of parallel robotic manipulators. Ph.D dissertation, McGill University, Montreal, Canada; 1988.
12. Nanua P, Waldron KJ, Murthy V. Direct kinematic solution of a Stewart platform. Robotics and Automation, IEEE Transactions on. 1990;6(4):438-44.
13. Innocenti C, Parenti V. Direct position analysis of the Stewart platform mechanism. Mech Mach Theory. 1990;25(6):611-21.
14. Innocenti C. Forward kinematics in polynomial form of the general Stewart platform. J Mech Des. 2001;123(2):254-60.
15. Kamali K, Akbarzadeh A. Fast estimation of direct kinematics of 3-RRR parallel manipulator using takagi–sugeno model. CFIS; Tehran, Iran. October 2008.
16. Mahmoodinia P, Kamali K, Akbarzadeh A. A novel approach for direct kinematics solution of 3-RRR parallel manipulator following a trajectory. ISME; Tehran, Iran. May 2008.
17. Kamali K, Akbarzadeh A. A novel method for direct kinematics solution of fully parallel manipulators using basic regions theory. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering. 2011;225(5):683-701.
18. Gregorio R, Parenti V. Position analysis in analytical form of the 3-PSP mechanism. J Mech Des. 2001;123(1):51-5.
19. Rezaei A, Akbarzadeh A, Nia PM, Akbarzadeh-T M-R. Position, Jacobian and workspace analysis of a 3-PSP spatial parallel manipulator. Robotics and Computer-Integrated Manufacturing. 2013;29(4):158-73.
20. Zhang Y, Liu H, Wu X. Kinematics analysis of a novel parallel manipulator. Mech Mach Theory. 2009;44(9):1648-57.
21. Merlet JP. Direct kinematics of parallel manipulators. Robotics and Automation, IEEE Transactions on. 1993;9(6):842-6.

22. Der-Ming K. Direct displacement analysis of a Stewart platform mechanism. Mech Mach Theory. 1999;34(3):453-65.s
23. Sadjadian H, Taghirad H, editors. Numerical methods for computing the forward kinematics of a redundant parallel manipulator. Proceedings of the IEEE Conference on Mechatronics and Robotics, Aachen, Germany; 2004.
24. Kai L, Fitzgerald JM, Lewis FL. Kinematic analysis of a Stewart platform manipulator. Industrial Electronics, IEEE Transactions on. 1993;40(2):282-93.
25. Li Y, Xu Q. Kinematic analysis of a 3-PRS parallel manipulator. Robot Cim-Int Manuf. 2007;23(4):395-408.
26. Cheng HH, Lee JJ, Penkar R. Kinematic analysis of a hybrid serial-and-parallel-driven redundant industrial manipulator. Int J Robot Autom. 1995;10(4):159-66.
27. Parikh PJ, Lam SSY. A hybrid strategy to solve the forward kinematics problem in parallel manipulators. Robotics, IEEE Transactions on. 2005;21(1):18-25.
28. Laosiritaworn W, Chotchaithanakorn N. Artificial neural networks parameters optimization with design of experiments: An application in ferromagnetic materials modeling. Chiang Mai J Sci. 2009;36(1):83-91.
29. Sekar BD, Ming Chui D, Jun S, Xiang Yang H. Fused Hierarchical Neural Networks for Cardiovascular Disease Diagnosis. Sensors Journal, IEEE. 2012;12(3):644-50.
30. Boudreau R, Levesque G, Darenfed S. Parallel manipulator kinematics learning using holographic neural network models. Robot Cim-Int Manuf. 1998;14(1):37-44.
31. Sadjadian H, Taghirad H, Fatehi A. Neural networks approaches for computing the forward kinematics of a redundant parallel manipulator. Int J Artif Intell . 2005;2(1):40-7.
32. Kang R, Chanal H, Bonnemains T, Pateloup S, Branson DT, Ray P. Learning the forward kinematics behavior of a hybrid robot employing artificial neural networks. Robotica. 2012;30(05):847-55.
33. Kalani H, Moghimi S, Akbarzadeh A. SEMG-based prediction of masticatory kinematics in rhythmic clenching movements. Biomedical Signal Processing and Control. 2015;20:24-34.
34. Clayton J. A pantographic reproducibility index for use in diagnosing TMJ function: a report on research. J Pros Dent. 1985;54(6):827-31.
35. Spoor CW. Explanation, verification and application of the helical axis error propagation formulas. Hum Mov Sci. 1984;3(1):95-117.
36. Airoldi RL, Gallo LM, Palla S. Precision of the jaw tracking system JAWS-3D. J Orofac Pain. 1994 Spring;8(2):155-64.
37. Grimes S-. A Review of Temporomandibular Disorder and An Analysis of Mandibular Motion [dissertation]. Tennessee Univ; 2005.
38. Xu WL, Fang FC, Bronlund JE, PotgieterJ. Generation of rhythmic and voluntary patterns of mastication usingMatsuoka oscillator for a humanoid chewing robot. Mechatronics. 2009;19(2):205-17.
39. Simi reality motion systems. Available at: http://www.simi.com. Accessed Feb 3, 2016.
40. Darvishi M, Barati A. A third-order Newton-type method to solve systems of nonlinear equations. Appl Math Comput. 2007;187(2):630-5.
41. Li Z, Peng C, Zhou T, Gao J. A new Newton-type method for solving nonlinear equations with any integer order of convergence. J Comput Inform Syst. 2011;7(7):2371-8.
42. Huang X, Liao Q Wei S. Closed-form forward kinematics for a symmetrical 6-6 Stewart platform using algebraic elimination. Mechanism and Machine Theory. 2010;45(2):327-34.
43. Lee TY , Shim JK. Forward kinematics of the general 6–6 Stewart platform using algebraic elimination. Mechanism and Machine Theory. 2001;36(9):1073-85.