J. Biomed. Eng. Res.
Vol. 20, No. 3, 479-484, 1999

6 자유도의 병렬기구를 사용한 저작(詛嚼)모델의 기구학적 분석

강 곤·堤定美*

건국대학교 의과대학 의공학과, Research Institute of Biomedical Engineering, Kyoto University, Kyoto. Japan*
(1999년 4월 15일 접수, 1999년 6월 2일 채택)

Kinematic Analysis of a Mastication Model Employing the 6-DOF Parallel Mechanism

G. Khang, Sadami Tsutsumi*

Dept. of Biomedical Engineering, Collage of Medicine, Konkuk University Choongju-city, Korea
(Received August 15, 1999, Accepted June 2, 1999)

요 약 : 본 연구에서는 사람의 턱 운동과 위아래 턱 사이에 작용하는 힘(혹은 압력)을 그대로 나타낼 수 있는 저작(詛嚼)로봇을 개발하는 것을 궁극적인 목표로 한다. 이러한 저작로봇이 개발되면, 치과의사가 환자의 턱운동에 나타나는 병변을 진단하고 치료하는데 큰 도움이 될 것으로 사료된다. 또한, 본 연구에서 채택한 병렬기구(parallel mechanism)에 대한 순기구학적(forward kinematics) 분석은 일반적인 병렬기구의 설계에도 응용될 것으로 기대된다.

본 연구진이 1차적으로 설계한 모델은 베이스와 플랫폼(platform), 그리고 이 둘을 연결하는 3개의 다리로 구성되어 있다. 다리와 플랫폼은 3 자유도의 관절로, 다리와 베이스는 1 자유도의 경첩 관절로 연결되어 있으며, 이 3개의 경첩 관절은 베이스 위의 수평면에서 직선을 따라 움직인다. 경첩 관절의 수평 변위와 세 다리의 길이가 주어졌을 때 플랫폼의 위치와 오리엔테이션을 구하는 순기구학의 해(解)를 계산해내는 알고리즘을 개발하였다. 이 알고리즘의 특징은 매 순간 오차를 계산하여 이 오차가 줄어드는 방향으로 나아가도록 시간간격(time step)을 조절하는 것이다.

본 알고리즘은 현재 가장 보편적으로 사용되고 있는 뉴튼-랩슨 방법에 비하여 3가지 장점을 나타내고 있다. 우선, 초기치(initial guess)에 관계없이 수렴한다는 것이다. 또한, 본 알고리즘은 뉴튼-랩슨 방법에 비하여 수렴 속도가 훨씬 빠르며, 연산 시간이 매우 짧아서 실제적인 실시간 적용에 적합하다. 마지막으로, 뉴튼-랩슨 방법에서는 여러 개의 해 가운데 어느 곳으로 수렴할지 예측할 수 없으나 본 알고리즘에서는 초기치에 가장 가까운 해로 수렴한다. 이러한 순기구학의 다중성(multiplicity) 문제를 해결하기 위하여 두 개의 조건을 제시하였으며, 이를 적용한 시뮬레이션 결과에 의하면 항상 원하는 해(true solution)에 수렴할 수 있었다.

Abstract : The ultimate goal of this research is to develop a mastication robot that can copy the human mandibular dynamics; not only kinematics but also pressure acting between jaws. This research was motivated by the fact that a mastication robot can be of great help to dentists in diagnosing and treating masticatory dysfunction, and the corresponding kinematic and dynamic analysis can be applied to other parallel mechanisms.

Our first-stage design consists of a base, 3 legs, 3 slides and a platform. One can change the position and orientation of the platform by varying the 3 horizontal displacements and 3 leg lengths, which is called the forward kinematics. We derived 3 nonlinear equations representing the forward kinematics of this design and developed a new algorithm to solve the forward kinematics. One character of our algorithm is that the step size is properly adjusted at each time instant in order to force the algorithm to converge.

The new algorithm has three major advantages over the conventional Newton-Raphson(N-R) method which is being widely used. First, our algorithm shows convergence for a wider range of the initial guess, whereas the N-R method is extremely sensitive to the initial guess. Secondly, our algorithm results in much faster convergence than the N-R method. In addition, the computation time is short

이 연구는 1996년 한국학술진흥재단 교수 해외파견지원으로 수행되었으며 이에 감사드립니다.
통신저자 : 강 곤, (380-701) 충북 충주시 단월동 322,
건국대학교 의과대학 의공학과
Tel. (0441)840-3761, Fax. (0441)851-0620
E-mail. gon.khang@kku.edu

enough for a real-time control of the mastication robot. The third advantage is that our algorithm enables us to find the solution closest to the initial guess among the potential solutions, while we cannot predict the position of the resulting solution in the N-R method. We also proposed two conditions to avoid reaching an undesirable solution, and those two conditions guaranteed uniqueness of the forward kinematics problem.

Key words : Mastication model, Parallel mechanism, Forward kinematics

INTRODUCTION

This research was motivated by an assumption that a mastication robot can be of great help to dentists in diagnosing and treating masticatory dysfunctions. The ultimate goal is, therefore, to develop a robot which can represent human mandibular motion and force(or pressure) acting between two mandibles as well. If we can reproduce mandibular motion based on the dynamic data taken from patients [1,2], dental diagnosis and treatment including oral surgery and orthodontics would be much more reliable than using the conventional articulator. The articulator can provide nothing but basic and rough information about patients' mandibular motion and involved force since the dentist uses his/her hands to move the articulator.

We employed the parallel mechanism to design a first-stage mastication model. The parallel mechanism has been attracting much attention since the Stewart platform was proposed[3]. The Stewart platform has six parallel legs between the platform and the base, and the position and orientation of the platform can be controlled by varying the length of each leg. The parallel mechanism is currently utilized in many industrial areas such as machining and welding[4].

Although a few problems should be solved before the parallel mechanism can be used for various practical applications, it is advantageous over the serial mechanism in many aspects. First of all, the position error is averaged and thus small while the position error of the serial mechanism is accumulated and becomes large as the number of links increases. Another advantage is that the parallel mechanism can generate large platform(or end-effector) force for a given set of actuators since the platform force is the very sum of the actuator forces. Noting that the workspace of the parallel mechanism is significantly smaller than that of the serial mechanism, one can say that the parallel mechanism is appropriate to use for accurate and strong manipulation inside a small workspace.

One of the major obstacles related to the parallel mechanism is lack of a reliable and fast algorithm to solve the forward kinematics. The forward kinematics is to obtain or find the position and orientation of the platform given the inputs, usually the leg lengths. Considering that almost every algorithm starts from an initial guess of the solution, i.e., the position and the orientation of the platform, we should guarantee convergence; the algorithm has to reach a solution. Much work has been done to develop a reliable algorithm with little success. Some algorithms are reported to work under specified conditions[5,6]. Another problem is that the currently available algorithms are not fast enough for practical real-time application, if their convergence is guaranteed. For example, genetic algorithms show high convergence despite of slow converging speed[7], and therefore they have been used together with the Newton-Raphson(N-R) method known to have fast converging speed[8]. Emphasis should be placed on multiplicity of the forward kinematics[9,10]; the position and orientation of the platform is not unique for a set of inputs (leg lengths in the Stewart platform). So, numerical computation starting from an initial guess may lead us to many different solutions of the forward kinematics. More research is needed to find the "true" solution starting from an arbitrary initial guess. Finally, it is not easy to determine the workspace compared to the serial mechanism, and this workspace analysis[11,12] is directly related to the above multiplicity problem. In this paper, we describe the first-stage design and the forward kinematic analysis of our mastication model. The convergence and converging speed of our new algorithm is compared with that of the conventional N-R method which is the most popular method these days. Also, we propose two conditions that can be imposed to avoid the multiplicity problem of the forward kinematics, and show numerical computation results.

METHOD

1. Design

The first-stage design consists of 3 legs and 3 slides between the platform and the base as shown in Fig. 1, whereas the conventional Stewart platform has 6 legs[1]. Each

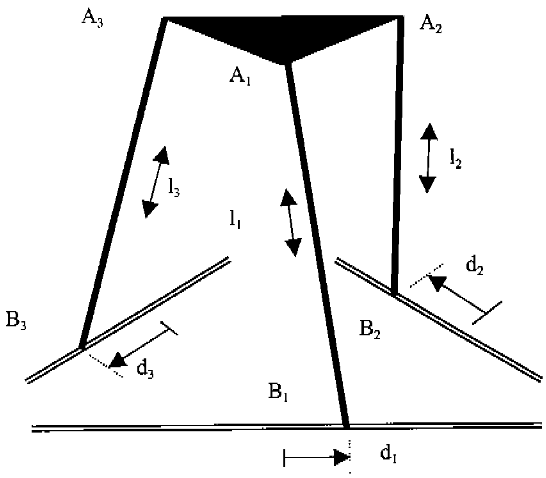

Fig. 1. Schematic of our Mastication Model

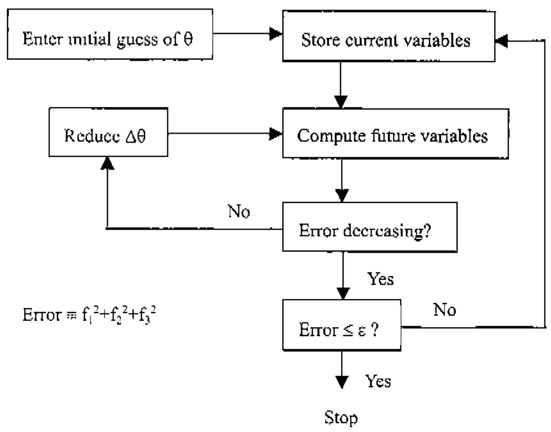

Fig. 2. Flowchart of our Algorthm for the Forward Kinematics

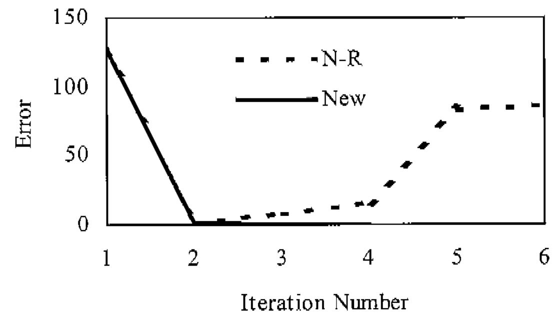

Fig. 3. Convergence
"N-R" and "New" denotes "Newton-Raphson method" and "our algorithm", respectively

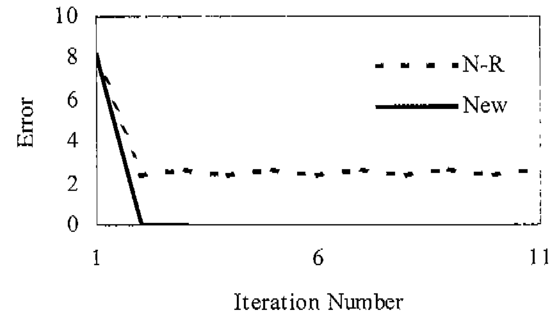

Fig. 4. Convergence vs Oscillation
The new algorithm shows satisfactory convergence while one can see oscillation in the N-R methold

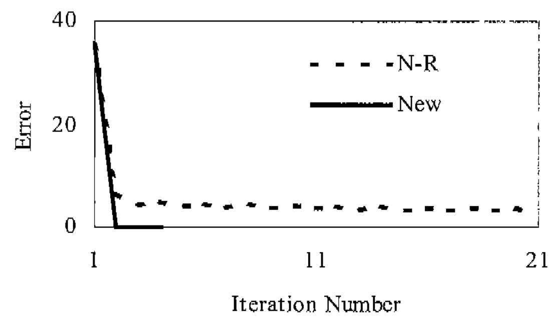

Fig. 5. Converging Speed
The new algorithm leads to significantly faster convergence than the N-R method

of the three legs is connected to the base by a 1-degree-of-freedom(DOF) hinge joint $B_i(i=1,2,3)$ and these joints move along the slides. Note that each leg is always perpendicular to the corresponding slide. The platform $A_1A_2A_3$ representing the lower mandible is connected to the legs by three 3-DOF ball-and-socket joints A_i's (i=1,2,3). Assuming the upper mandible to be fixed in the space, we can reproduce the masticatory motion by moving the lower mandible, i.e., the platform.

Our workspace analysis indicated that the current dimension of the model summarized below in centimeters can cover the range of the human mandibular movement.

$$-3 \le d_i \le 3,\ 24 \le l_i \le 30,\ A_1A_2 = A_2A_3 = A_3A_1 = 5\ (i=1,2,3)$$

$$B_1B_2 = B_2B_3 = B_3B_1 = 10 \text{ at the neutral position}$$

(see below for definition)

The position and orientation of the platform can be controlled by adjusting the inputs; d_i's, displacements of B_i's, and l_i's (i=1,2,3), lengths of the legs. Let's assume that $d_i=0$ and $l_i=l_0$ (i=1,2,3) at the neutral position where B_i (i=1,2,3) coincides with the midpoint of each slide, and call the angles between the legs and the base θ_i's (i=1,2,3). In order to simplify the mathematical analysis we assumed

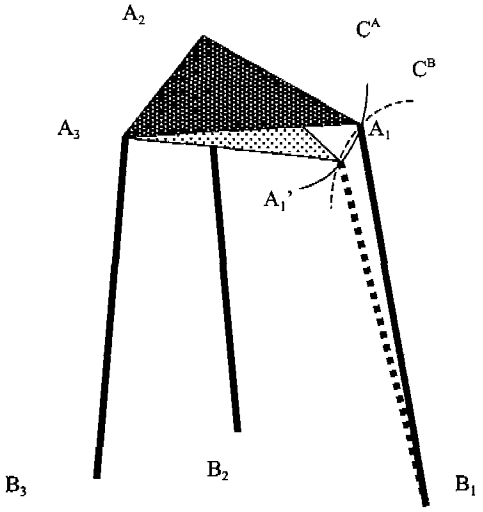

Fig. 6. Multiplicity of the Forward Kinematics

that every element of the model is a rigid body so that the platform and the legs do not change their shape regardless of forces and moments acting on the model.

2. Forward Kinematics

Given the input, d_i's and l_i's (i=1,2,3), we derived 3 nonlinear equations $f_i(d_n, l_k, \theta_n)$'s (i,j,k,n=1,2,3) to solve the forward kinematics of our model, i.e., to obtain the corresponding position and orientation of the platform. The forward kinematics can be characterized by 3 nonlinear equations of the leg angles θ_1, θ_2 and θ_3, whereas 6 unknowns, 3 coordinates of an arbitrary point of the platform and 3 orientations of the platform, should be solved in case of the Stewart platform. The Cartesian coordinates of A_i's can be expressed as functions of d_i's, l_i's and θ_i's (i=1,2,3). Considering the constraints that A_i's should comprise a triangle, assumed to be equilateral up to date, we obtained the following 3 nonlinear equations to be solved for θ_i's;

$$
\begin{aligned}
f_1(d_i, l_j, \theta_k) &= A_1A_2 - L_{12} = 0 \\
f_2(d_i, l_j, \theta_k) &= A_2A_3 - L_{23} = 0 \qquad (1) \\
f_3(d_i, l_j, \theta_k) &= A_3A_1 - L_{31} = 0 \quad (i,j,k=1,2,3)
\end{aligned}
$$

A_iA_j(i,j=1,2,3) denotes the distance between A_i and A_j, and L_{ij} means the predetermined side length of the triangle $A_1A_2A_3$. As far as the numerical computation is concerned, the forward kinematics is to find θ_i's (given d_i's and l_i's) that minimize the error defined as

$$\mathrm{Error}(cm^2) \triangleq f_1^2 + f_2^2 + f_3^2 \qquad (2)$$

We modified an existing algorithm[13] to solve equation (1), which is schematically shown in Fig. 2. The iteration starts from an initial guess of the solution θ_i's (i=1,2,3). Storing the current values of the variables, the algorithm computes the variables at the next time instant. And then the error defined as (2) is computed and compared with the current value to check if the error is decreasing. In other words, the algorithm checks if the iteration is advancing in the right direction. If the error is not decreasing, the increment $\Delta\theta$ is reduced and the algorithm again computes the variables at the next time instant. If the error is decreasing, we can say that we are getting close to the solution. If the error is smaller than the predetermined small constant ε, the iteration stops and the current θ_i's are regarded as the solution. If not, the iteration advances storing the current values of θ_i's. The resulting θ_i's can be immediately converted into the position and orientation of the platform through the inverse kinematics which is much simpler than the forward kinematics.

RESULTS AND DISCUSSION

Our design is advantageous over the Stewart platform in that its forward kinematics can be characterized by 3 nonlinear equations, while 6 nonlinear equations should be solved in case of the Stewart platform. It can be, therefore, easily expected that our design needs less computation time than the Stewart platform. Shortening the computation time for the forward kinematics can make the real-time control possible.

Reducing the number of equations usually enables us to shorten the computation time, if nonlinearity of the equations are not much different. This is obviously one reason why our design is advantageous over the Stewart platform. However, it mostly depends on "dynamics", not kinematics, whether the total numerical computation takes short time enough for real-time control[12]. Therefore, we have to go through dynamic analysis of our model before the real-time controllability is discussed. On the other hand, reducing the number of legs from 6 to 3 not only enhances appearance of the model but also gives more room to actuators and control devices. This is another benefit of our model in terms of practical usage. In our model, however, friction acting at the three joints of the base (B_1, B_2 and B_3 in Fig. 1) should be taken into deep consideration since the three horizontal actuators have to overcome this friction caused by the total weight of the platform and three legs in order to move these joints.

Our new algorithm has three major advantages over the conventional N-R method, which is being widely used[6]. First, our algorithm shows convergence for a wider range of the initial guess, whereas the N-R method is so sensitive to the initial guess that it easily diverges. Fig. 3 represents a typical iteration for a random initial guess. We can see that both the algorithms seem to be converging fast but the N-R method diverges after two or three iterations. Fig. 4 shows another case where our algorithm also converges fast when the N-R method oscillates never to reach the solution. Although our algorithm does not guarantee convergence for any initial guess, we can always force it to converge by choosing an initial guess close to the solution.

Secondly, our algorithm results in much faster convergence than the N-R method. Fig. 5 shows that the N-R method requires much more iterations in order to converge than our algorithm. The computation time was not more than 10^{-10} second (when using a PC equipped with a pentium processor) which is short enough for a real-time control of a mastication robot unless its dynamics takes too much computation time. It should be noted that most of the genetic algorithms are not adequate for practical application because they require too much computation time in spite of good convergence[7].

The third advantage is that our algorithm enables us to reach the solution closest to the initial guess among the potential solutions(see below for uniqueness of the solution). This advantage plays a very important role in numerical computation in that the "true" solution can be always reached as long as one provides a nearby initial guess. This is not true in any other algorithm, i.e., the final destination cannot be predicted even if the algorithm converges.

Like most serial and parallel robots as well, our kinematic analysis indicated that equation (1) has multiple solutions; given a set of inputs, we obtain multiple sets of θ_i's $(i=1,2,3)$ satisfying equation (1). As an example, assuming $d_1=d_2=d_3=0$ and $l_2=l_3\neq l_1$, we can think of two different positions of A_1 as shown in Fig. 6, where C^A is the locus of A_1 when the triangle $A_1A_2A_3$ is rotated about A_2A_3 and C^B is part of the circle formed by rotating the leg B_1A_1 about B_1. Since the two loci meet at two different positions A_1 and A_1', the forward kinematics of our model does not guarantee uniqueness. We, therefore, applied the following two conditions to select the "true" position of the platform for a given set of inputs, and found that the solution satisfying the above conditions is unique, if exist;

(1) B_i's$(i=1,3)$ should be always below the plane formed by extending the triangle $A_1A_2A_3$.

(2) When we assume an axis parallel to d_1, the coordinate of A_2 should be always greater than that of A_3.

We claimed that our algorithm is sure to converge to a solution near the initial guess. The question would be then "Is that solution what we want?". First, the two conditions can be the answer to the above question. According to our simulation results, the algorithm always reach the "true" solution without exception. We, however, have to say that this statement should be followed by a theoretical proof, which is one of our continuing research topics. Second, the answer is YES only if we can provide the first initial guess close to "true" solution. Suppose we simulate a platform movement caused by adjusting the inputs, d_i's and l_i's $(i=1, 2,3)$ in Fig. 1, and the movement starts from a known position, or the initial position can be always measured. The platform position at the next time instant can be reached if we take advantage of the initial (known) position as the intial guess, as long as the time step and the increment of the movement is small. We can successively compute the platform position at the next time instant using the current position as the initial guess.

REFERENCES

1. T. Fujimura and E. Bando, *"The development of a digital type jaw movement analyzer"*, J Jpn Prosthodont Soc, vol. 35, pp.830-842, 1991
2. E. Bando and H. Takeuchi, *"Analyzing systems of jaw movements"*, J Jpn Dental Engrg, no. 102, pp. 27-32, 1992
3. D. Stewart, *"A platform with six degrees of freedom"*, Proc of the Institute for Mech Engrg, London, vol. 180, pp.371-386, 1965
4. "特輯 パラレルメカニズムが機械を変える", Nikkei Mechanical, vol. 3, pp.26-49, 1995
5. K. E-Zamganeh and J. Angeles, *"Real-time direct kinematics of general six-degree-of-freedom parallel manipulators with minimum-sensor data"*, J Robotic Systems, vol. 12, pp.833-844, 1995
6. C.M. Gosselin, L. Perreault and C. Vaillancourt, *"Simulation and computer-aided kinematic design of three-degree-of-freedom spherical parallel manipulators"*, J Robotic Systems, vol. 12, pp.857-869, 1995

7. R. Boudreau and N. Turkkan, "*Solving the forward kinematics of parallel manipulators with a genetic algorithm*", J Robotic Systems, vol. 13, pp.111-125, 1996
8. P.R. McAree and R.W. Daniel, "*A fast, robust solution to the Stewart platform forward kinematics*", J Robotic Systems, vol. 13, pp.407-427, 1996
9. X. Shi and R.G. Fenton, "*A complete and general solution to the forward kinematics problem of platform-type robotic manipulators*", Proc of IEEE Int Conf on Robotics and Automation, San Diego, USA, pp.3055-3062, 1994
10. L. Notash and R.P. Podhorodeski, "*Forward displacement analysis and uncertainty configurations of parallel manipulators with a redundant branch*", J Robotic Systems, vol. 13, pp.587-601, 1996
11. J. Merlet, "*Trajectory verification of parallel manipulators in the workspace*", Proc of IEEE Int Conf on Robotics and Automation, San Diego, USA, pp.2166-2171, 1994
12. J. Bessala, P. Bidaud and F. Ouezdou, "*Analytical study of Stewart platforms workspace*". Proc of IEEE Int Conf on Robotics and Automation, Minneapolis, USA, pp.3179-3184, 1996
13. "*Numerical recipes in Fortran 77: the art of scientific computing*", Cambridge University Press, chapter 9, 1998
14. G. Lebret, K. Liu and F.L. Lewis, "*Dynamic analysis and control of a Stewart platform manipulator*", J Robotic Systems, vol. 10, pp.629-655, 1993