http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 13 (5): 920-926, 2014
ISSN 1812-5638 / DOI: 10.3923/itj.2014.920.926

Development of Analysis and Study System for One Type 3-PRS Parallel Robot

[1]Jianli Kang, [2]Guoqiang Chen and [2]Junwei Zhao
[1]School of Architectural and Artistic Design, Henan Polytechnic University, China
[2]School of Mechanical and Power Engineering, Henan Polytechnic University, China

Abstract: The 3-PRS parallel robot has many advantages of high stiffness, high accuracy, little cumulate error, large load carrying capacity, compact structure and so on and has been becoming a research hotspot in the world. The characteristic analysis refers to lots of complicated mathematical formulas, trigonometric function and root operation. Aiming at the complexity, the study proposes to develop an analysis and study system for one type of 3-PRS parallel robot that can print texts on the spherical surface. The key modules and technology including dot matrix generation, spherical surface mapping, inverse kinematics computation, results demonstration and storage are discussed in detail. The running result indicates that the developed system has the characteristics of intuition, convenience and practicality. The analysis system can be used in the design, assessment and study of the 3-PRS parallel robot.

Key words: 3-PRS parallel robot, forward kinematics, inverse kinematics, dot matrix type, parasitic motion

INTRODUCTION

The industrial robot includes the serial robot and the parallel robot on the whole. The parallel robot, also called parallel machine tool, has many advantages of high stiffness, high accuracy, little cumulate error, large load carrying capacity, compact structure and so on over their serial counterpart, so it has gained wide applications in all kinds of fields (Wahl, 2000; Pouliot *et al*., 1998; Yu *et al*., 2004; Liu *et al*., 2003; Li and Xu, 2007). The 3-PRS parallel robot falls within the [PPS] type parallel robotic mechanisms and is a typical imperfect-DOF (degree of freedom) parallel mechanism (Liu *et al*., 2008). The parallel robot has been becoming a research hotspot in the world and is a key subject of technical training and study in the university and institute since it was proposed (Hunt, 1983). The characteristic analysis including kinematics, error, working space and so on, refers to lots of complicated mathematical formulas, trigonometric function and root operation, which brings out much difficulty for the beginner (Li and Xu, 2007). For example, the forward kinematics problem is very difficult to solve because it is a system of nonlinear equations including the trigonometric function. The analytical solution brings out a large number of undesired solutions and the further complex examination is needed to get the feasible one (Tsai *et al*., 2003). Because the 3-PRS parallel robot has only three DOF, it generates three parasitic motions which brings out problems in some cases (Carretero *et al*., 2000; Li *et al*., 2011; Abbasnejad *et al*., 2010; Huang *et al*., 2012; Tsai and Yuan, 2011; Selvakumar *et al*., 2012; Tsai *et al*., 2003). How to efficiently analyze and demonstrate the characteristic of the 3-PRS robot is an important issue many beginners, instructors and engineers have been facing for a long time. An excellent Robotics Toolbox developed by Peter Corke in MATLAB provides many functions for the study and simulation of classical arm-type robotics and can be downloaded freely from Website http://www.petercorke.com. It is very necessary to develop an analysis and study system for the 3-PRS parallel robot. Among hundreds of development environments and high languages, MATLAB is a commercial "Matrix Laboratory" package and is well adapted to numerical computation, symbolic operation and graphical representation. GUIs (also known as graphical user interfaces or UIs) in MATLAB provide point-and-click control of software applications and is used in the study. Aiming at a typical application of the robot, the study discusses the key technology to develop an analysis and study system for one type of 3-PRS parallel robot that can print texts on the spherical surface.

MATERIALS AND METHODS

Scheme and mobility of 3-PRS parallel robot: The schematic representation of the 3-PRS parallel robot is shown in Fig. 1 (Li *et al*., 2011). The robot is composed of a moving platform, three limbs, three vertical rails and a fixed base. Three vertical rails are vertically linked to the base B_1 B_2 B_3 that form an equilateral triangle that lies on

Corresponding Author: Guoqiang Chen, School of Mechanical and Power Engineering, Henan Polytechnic University, China

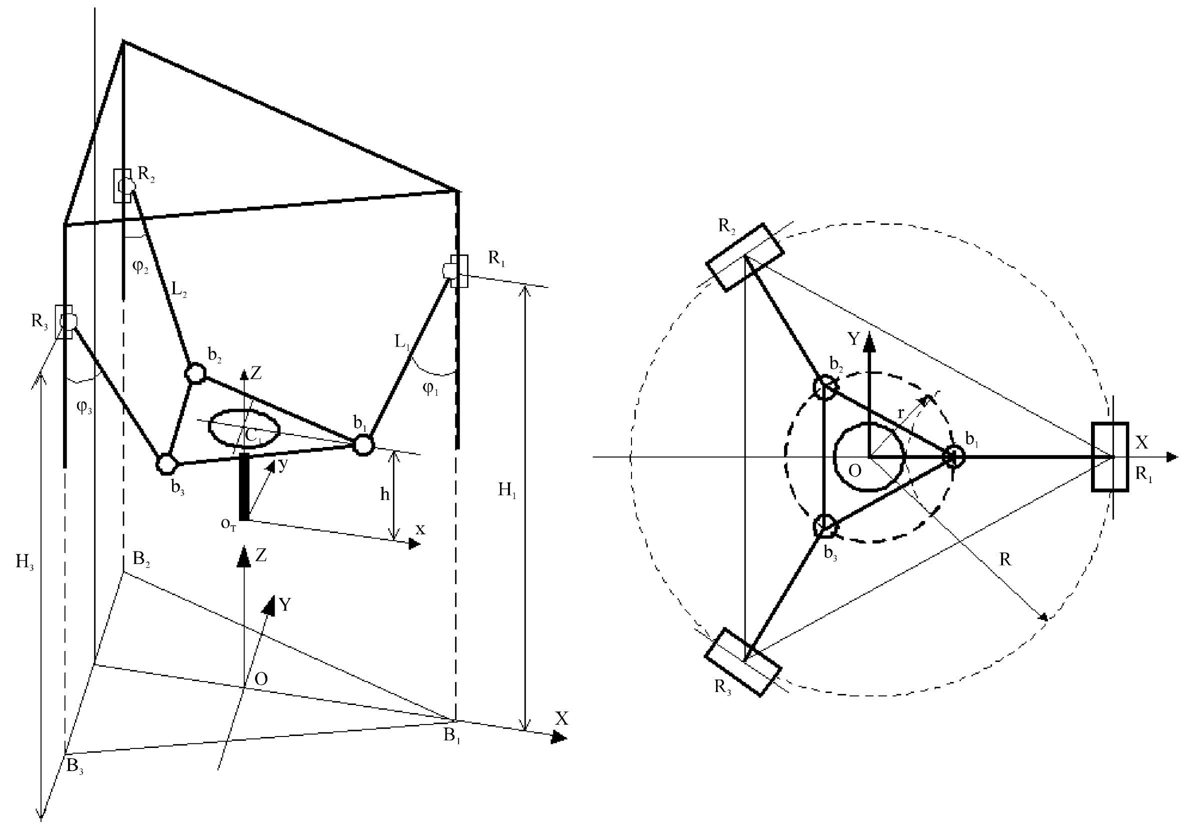

Fig. 1: Schematic representation of the 3-PRS parallel robot

a circle with the radius R.The axis of the revolute pair R_i for i = 1, 2 and 3 is perpendicular to the prismatic pair. Each limb L_i for i = 1, 2 and 3 with the length l_i connects the corresponding rail by a prismatic pair R_i. The moving platform and three limbs are connected by three spherical pairs b_1 b_2 and b_3 that form an equilateral triangle that lies on a circle with the radius r. The cutter is placed at the center of the moving platform with the length of h. The feed of the prismatic pair is given as H_i. Angle φ_i for i = 1, 2 and 3 is defined from the vertical rail to its corresponding limb L_i.

For the sake of analysis, two coordinate frames are used, as shown in Fig. 1. A fixed Cartesian reference coordinate system OXYZ is located at the center O of the base B_1 B_2 B_3. X-axis and Y-axis are in the base plane B_1 B_2 B_3, X-axis points in the direction of OB_1 and Z-axis is normal to the base plane and points upward. A moving coordinate frame $o_T$xyz is located at the cutter point o_T. The xy plane is parallel to the moving platform b_1 b_2 b_3, x-axis points in the direction of C_1b_1 and z-axis is normal to the moving platform. The position and orientation of the cutter can be described using the coordinates (x_T, y_T, z_T) of the cutter point and three Euler angles α, β and γ rotating about the Z, Y and X axes of the fixed reference frame. The rotation matrix from the moving frame $o_T$xyz to the fixed frame OXYZ can be expressed as follows:

$$T=\begin{bmatrix} C\beta C\gamma & -C\beta S\gamma & S\beta & x_T \\ S\alpha S\beta C\gamma + C\alpha S\gamma & -S\alpha S\beta S\gamma + C\alpha C\gamma & -S\alpha C\beta & y_T \\ -C\alpha S\beta C\gamma + S\alpha S\gamma & C\alpha S\beta S\gamma + S\alpha C\gamma & C\alpha C\beta & z_T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

where S_α, S_β, S_γ, C_α, C_β and C_γ stand for sinα, sinβ, sinγ, cosα, cosβ and cosγ, respectively.

The 3-PRS parallel robot possesses 3- DOF that are rotation α about the Z-axis and β about the Y-axis and a translational motion z_T along the Z-axis. Three parasitic motions are one rotation γ about the X-axis, one translational motion x_T about the X-axis and one translational motion y_T about the Y-axis. The three parasitic motions γ, x_T and y_T can be expressed using the other three independent motions α, β and z_T. The rotation γ can be computed as follows:

$$\gamma = -\arctan\left(\frac{\sin\alpha\cdot\sin\beta}{\cos\alpha+\cos\beta}\right) \tag{2}$$

Two parasitic translational motion x_T and y_T can be expressed as follows:

$$\begin{cases} x_T = \frac{r}{2}(\cos\beta\cos\gamma + \sin\alpha\sin\beta\sin\gamma - \cos\alpha\cos\gamma) - h\sin\beta \\ y_T = h\sin\alpha\cos\beta - r\sin\alpha\sin\beta\cos\gamma - r\cos\alpha\sin\gamma \end{cases} \quad (3)$$

SYSTEM DEVELOPMENT AND KEY TECHNOLOGY

System architecture and function: The whole function of the developed software system is to analyze and demonstrate the key technology, control methods, structural characteristics of 3-PRS parallel robot based on one kind of typical application on how to print texts/characters on the spherical surface. As shown in Fig. 2, the architecture of the system can be divided into five modules:

- **Character and parameter setting:** This module receives inputs about the 3-PRS robot, sphere and so on from the user
- **Dot matrix generation:** In order to print characters, the characters should be expressed in dot matrix format. This module genates the dot matrix type for the set characters with different font
- **Spherical surface mapping:** This module maps the dot matrix of the character from plane to spherical surface patch
- **Inverse kinematics computation:** The three inputs H_1, H_2, H_3 and other dynamic data for the 3-PRS robot are computed using inverse kinematics based on the dot matrix on the spherical surface
- **Results demonstration and storage:** This module demonstrates the analysis results using appropriate figures or stores them in files

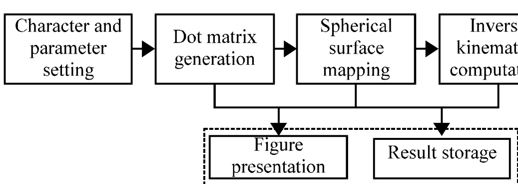

Fig. 2: System architecture and function

Spherical surface mapping: As shown in Fig. 3, the parametric equation of the sphere surface with the radius R_s can be expressed using three parameters as follows:

$$\begin{cases} x = R_s\sin\theta\cos\lambda \\ y = R_s\sin\theta\sin\lambda \\ z = R_s\cos\theta \end{cases} \qquad (0 \le \theta \le \pi,\ 0 \le \lambda < 2\pi) \quad (4)$$

In the study, a spherical surface patch can be defined by five parameters: R_s, θ_1, θ_2, λ_1 and λ_2, as shown in Fig. 3. That is to say, every point (x, y, z) in patch can be computed and satisfys the constraints: $\theta_1 \le \theta \le \theta_2$ and $\lambda_1 \le \lambda \le \lambda_2$.

The dot matrix type of the character as shown in the left part of Fig. 4 is stored in a matrix variable Z with m rows and n columns. The whole rectangular zone that the character resides can be divided into m×n squares. Every square is corresponding to an element of matrix Z. The value of the corresponding element is set to 1 if any stroke crosses it, while values of the other elements are set to 0. The method to mapping dot matrix of the character from type to the spherical surface patch is shown in Fig. 4. The spherical surface patch is divided

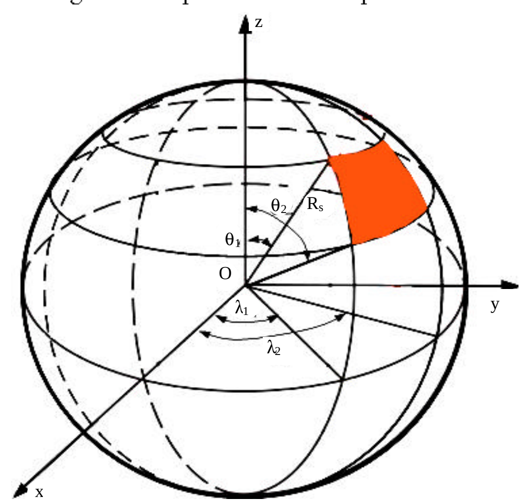

Fig. 3: Meanings of three parameters in the sphere equation and definition of a spherical surface patch

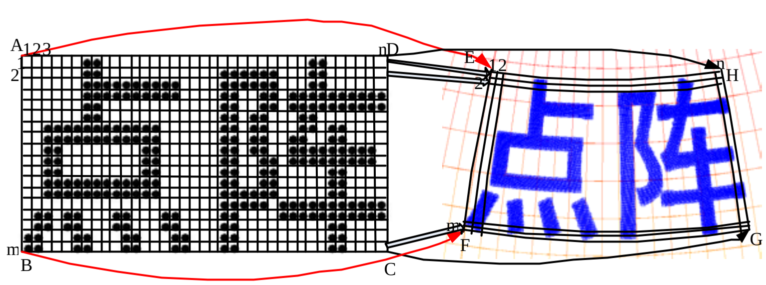

Fig. 4: Method to mapping dot matrix of the character from type to spherical surface patch

into m×n sub-patches. Every sub-patch is corresponding to a square of the tpye. Every sub-patch is defined by four angles, that is to say, the sub-patch with the i-th row and j-th column satisfys the following constraints:

$$\begin{cases}\theta_1+(i-1)\delta_\theta \le \theta \le \theta_1 + i\delta_\theta \\ \lambda_1+(j-1)\delta_\lambda \le \lambda \le \lambda_1 + j\delta_\lambda\end{cases} \quad (i=1,2,\cdots,m \text{ and } j=1,2,\cdots,n) \tag{5}$$

where, $\delta_\theta = (\theta_2-\theta_1)/(m-1)$ and $\delta_\lambda = (\lambda_2-\lambda_1)/(n-1)$

Dot matrix type generation: The intuitive method to abstract the font type is to reading it from the font dot matrix database files. Many database files are needed because there are many different fonts, several styles and so on. The study adopts a new method to generate the dot matrix type. The procedure can be expressed as follows:

Step 1: Setting the font property. The Font selection dialog box function UISETFONT is used to set the font and style. S = UISETFONT (FIN, 'dialogTitle') displays a dialog box for the user to fill in and returns the values to S

Step 2: Displaying the character in a figure. The text annotation function TEXT is used. TEXT(X, Y,'string') adds the text 'string' in the quotes to location (X, Y) on the current axes. In order to locate the text at the center of the figure, (X, Y) is set as (0, 0.5). The number of single-byte characters is assumed as n_s. In order to make the text fill the whole figure as much as possible, the FontSize property should change with n_s and is set as $3/2n_s$ in the study

Step 3: Saving the figure in an image file. The function SAVEAS can save Figure or Simulink block diagram in desired output format. In the study, the figure that displays the character is written to windows bitmap file

Step 4: Abstracting the dot matrix type from the figure. The function IMREAD (FILENAME, FMT) reads a grayscale or color image with the format of the file by its standard specified by the text string FMT from the file specified by the string FILENAME. The return value is an array containing the image data. The file in Step 3 is a grayscale image, so the return value is an m-by-n array. The value of the corresponding element in the arry is regulated to 1 or 0 according to a specified threshold value which is set to 128 in the study. The margins in the dot matrix type are also removed

Inverse kinematics computation: The unit direction vectors of the X, Y and Z coordinate axes are $\vec{E}_1$ = [1, 0, 0], $\vec{E}_2$ = [0, 1, 0] and $\vec{E}_3$ = [0, 0, 1]. The unit direction vectors of the x, y and z coordinate axes in the coordinate system OXYZ are assumed as $\vec{e}_1$, $\vec{e}_2$ and $\vec{e}_3$. The cutter vector corresponding to a point with the coordinates (x, y, Z_T) can be expressed as:

$$\vec{e}_3 = -\begin{bmatrix}x & y & z_T\end{bmatrix}^T \Big/ \sqrt{x^2+y^2+z_T^2} \tag{6}$$

The direction cosine matrix composed by the 9 elements in the upper left of T can be expressed as:

$$D = \begin{bmatrix}\vec{E}_1\bullet\vec{e}_1 & \vec{E}_1\bullet\vec{e}_2 & \vec{E}_1\bullet\vec{e}_3 \\ \vec{E}_2\bullet\vec{e}_1 & \vec{E}_2\bullet\vec{e}_2 & \vec{E}_2\bullet\vec{e}_3 \\ \vec{E}_3\bullet\vec{e}_1 & \vec{E}_3\bullet\vec{e}_2 & \vec{E}_3\bullet\vec{e}_3\end{bmatrix} \tag{7}$$

So the orientation angles α, β and γ can be computed using Eq. 1, 6 and 7. Three parasitic motions x_T and y_T can be computed using Eq. 3. The rotation matrix T can be gotten.

In the coordinate system $o_T$xyz, the coordinates of the three spherical pairs b_1, b_2 and b_3 can be expressed in vectors as follows:

$$\begin{cases} b_{1o_T} = \begin{bmatrix}r & 0 & h\end{bmatrix}^T \\ b_{2o_T} = \begin{bmatrix}-r/2 & \sqrt{3}r/2 & h\end{bmatrix}^T \\ b_{3o_T} = \begin{bmatrix}-r/2 & -\sqrt{3}r/2 & h\end{bmatrix}^T \end{cases} \quad (i=1,2,3) \tag{8}$$

So the coordinate vectors in the coordinate system OXYZ can be computed:

$$\begin{bmatrix}b_i \\ 1\end{bmatrix}_o = T\cdot\begin{bmatrix}b_{io_T} \\ 1\end{bmatrix}_{o_T} \tag{9}$$

The feeds of the prismatic pair H_i for i = 1, 2, 3 can be computed:

$$H_i = Z_{bi} + \sqrt{l_i^2 - (X_{Ri}-X_{bi})^2 - (Y_{Ri}-Y_{bi})^2} \tag{10}$$

where, X_{Ri}, X_{bi} are the coordinates of R_i and b_i along X-axis and Y_{Ri}, Y_{bi} along Y-axis.

An X-Y table is always used to correct the parasitic motions x_T and y_T under the real-time control (Li *et al*., 2011). Along X-axis and Y-axis in the coordinate system OXYZ, the corresponding compensation motions x_{Tab} and y_{Tab} of the table are:

(a)

(b)

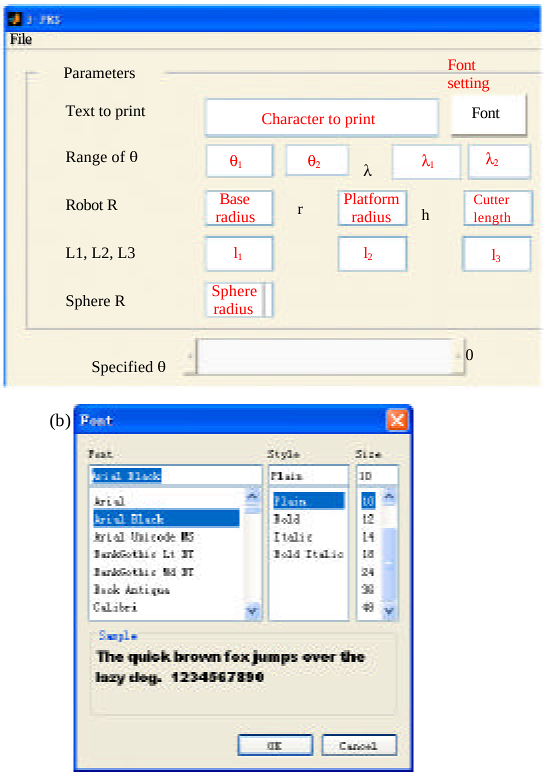

Fig. 5(a-b): Software interface for (a) Parameter meanings and setting and (b) Font setting

$$\begin{cases} x_{Tab} = x - x_T \\ y_{Tab} = y - y_T \end{cases} \tag{11}$$

Parameters setting and results presentation: The inputs parameters include characters to print and font, parameters of the 3-PRS robot, the sphere radius, the print area and so on. The parameter setting interface and the corresponding paramerers are shown in Fig. 5.

The input parameters of the angles and the robot are converted from string matrix to numeric array using the function str2num. The function MESH is used to plot the sphere and the view point is specified by the function VIEW. In order to conveniently observe the print text, the viewpoint to the Cartesian coordinates X, Y and Z is set as:

$$\left(\sin\frac{\theta_1+\theta_2}{2}\cos\frac{\lambda_1+\lambda_2}{2},\ \sin\frac{\theta_1+\theta_2}{2}\sin\frac{\lambda_1+\lambda_2}{2},\ \cos\frac{\theta_1+\theta_2}{2}\right) \tag{12}$$

RESULTS AND DISCUSSION

The parameters are set as: $\theta_1 = 10°$, $\theta_2 = 20°$, $\lambda_1 = 0°$, $\lambda_2 = 60°$, R = 350, r = 200, h = 280, $l_1 = l_2 = l_3 = 1107$ and $R_s = 100$. The texts to print are "Journal", the font is Arial and the style is Bold Italic. The character type and the spherical surface mapping results are shown in Fig. 6. In order to print texts on the spherical surface patch, the airbrush of the inkjet printer must be controlled to move arbitrarily above the spherical surface patch. There are many strategies on moving the airbrush of the inkjet printer. Two intuitive strategies are to move the airbrush line by line along the longitude or the latitude. The airbrush prints when the airbrush is crossing the sub-patch with value 1 in the dot matrix type. In the study, the airbrush moved line by line along the latitude. The kinematics data of the 3-PRS robot are shown in Fig. 7 and 8, respectively for the latitude with $\theta_1 = 14°$ and $\theta_1 = 20°$.

(a)

(b)

(c)

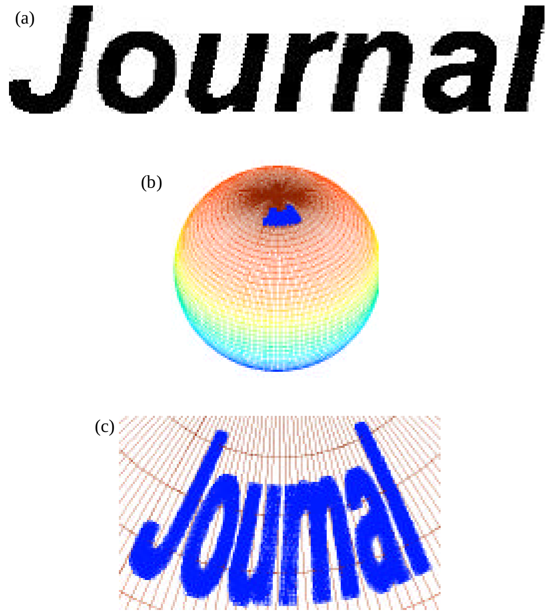

Fig. 6(a-c): Results of (a) Dot matrix type and print texts, (b and c) Spherical surface with texts print where texts for print "Journal", the font Arial and the style bold italic

Many characteristics of the 3-PRS robot can be demonstrated by these figures. Three parasitic motions are usually nonzero and is determined by the geometrical and parameters of 3-PRS robot as discussed by Li *et al*. (2011). For the architecture of a 3-PRS robot in Fig. 1, the parasitic rotation motion γ is 0 if and only if α or β is 0. From Fig. 7a and 8a, the parasitic rotation motion γ almost zero because the absolute values of the other Euler angles α and β are very small, as can be explained by Eq. 2. An X-Y table is needed to compensate the parasitic motions x_T and y_T and the compensations are shown in Fig. 7d and 8d. As pointed by Tsai *et al*. (2003), the X-Y table can

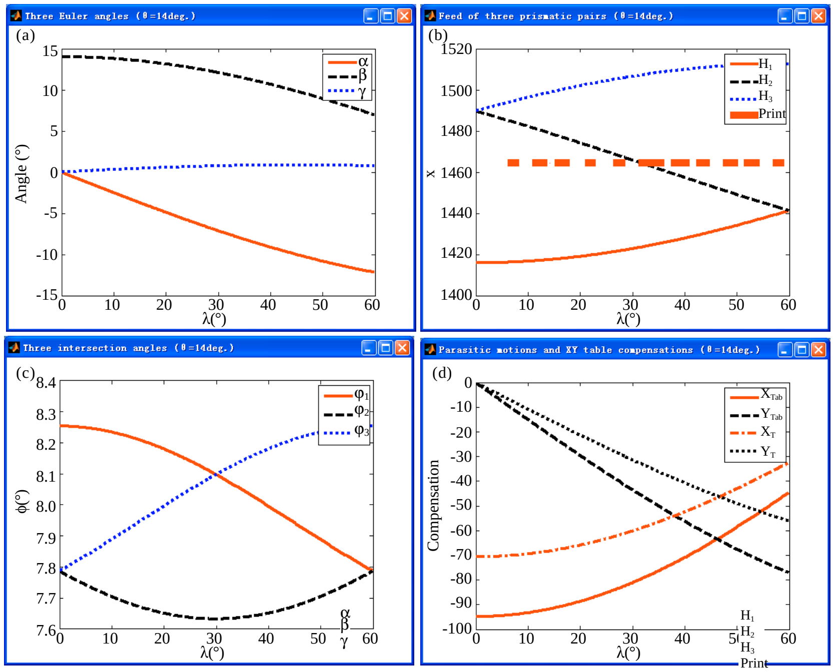

Fig. 7(a-d): Results for specified θ_1 = 14°, texts to print "Journal", the font Arial and the style bold italic, (a) Three Euler angles, (b) Feed of three prismatic pairs, (c) Three intersection angles and (d) Parasitic motions and XY table compensations

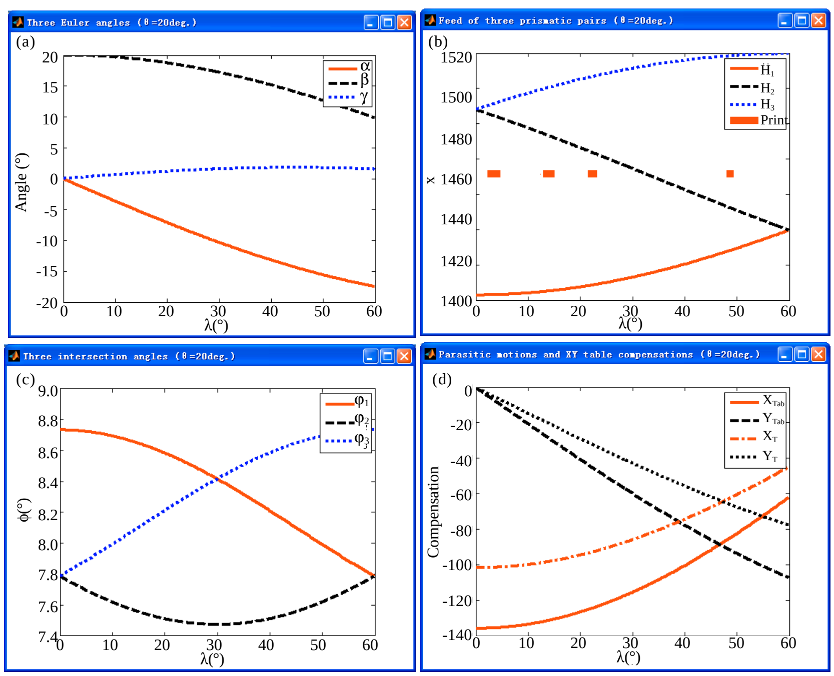

Fig. 8(a-d): Results for specified θ_1 = 20°, texts to print "Journal", the font Arial and the style bold italic, (a) Three Euler angles, (b) Feed of three prismatic pairs, (c) Three intersection angles and (d) Parasitic motions and XY table compensations enlarge the working space and provide high-speed performance. Three intersection angles φ_i shown in Fig. 7c and 8c for i = 1, 2 and 3 are formed naturally according to the feeds of three prismatic pairs. The corresponding relation between three feeds and "print on" signal is shown in Fig. 7b and 8b.

CONCLUSION

In this study a computer aided analysis and study system is developed in MATLAB for one type of 3-PRS parallel robot that can print texts on the sphere spherical surface. The system includes several key modules: dot matrix generation, spherical surface mapping, inverse kinematics computation, results demonstration and storage and so on. The key modules and technology are discussed in detail. The development procedure makes full use of the image processing and presentation functions and complicated mathematical operations. The running result indicates that the developed system has the characteristics of intuition, convenience and practicality. The analysis system can be used in the design, assessment and study of the 3-PRS parallel robot.

ACKNOWLEDGMENTS

This study is supported by Scientific and Technological Project in Henan Province of China with grant No. 132102210430. The authors would like to thank the anonymous reviewers for their valuable comments.

REFERENCES

Abbasnejad, G., S. Zarkandi and M. Imani, 2010. Forward kinematics analysis of a 3-PRS parallel manipulator. World Acad. Sci. Eng. Technol., 37: 329-335.

Carretero, J.A., C.M. Gosselin, R.P. Podhorodeski and M.A. Nahon, 2000. Kinematic analysis and optimization of a new three degree-of-freedom spatial parallel manipulator. J. Mech. Des., 122: 17-24.

Huang, P., J. Wang, L. Wang and J. Qian, 2012. Dimensional synthesis for 3-PRS mechanism based on identifiability performance. Chinese J. Mech. Eng., 25: 234-240.

Hunt, K.H., 1983. Structural kinematics of in-parallel-actuated robot-arms. J. Mech. Trans. Autom., 105: 705-712.

Li, Y. and Q.S. Xu, 2007. Kinematic analysis of a 3-PRS parallel manipulator. Rob. Comput. Integr. Manuf., 23: 395-408.

Li, Q.C., Z. Chen, Q.H. Chen, C.Y. Wu and X.D. Hu, 2011. Parasitic motion comparison of 3-PRS parallel mechanism with different limb arrangements. Rob. Comput. Integr. Manuf., 27: 389-396.

Liu, D., R.S. Che, Z. Li and X.C. Luo, 2003. Research on the theory and the virtual prototype of 3-DOF parallel-link coordinate-measuring machine. IEEE Trans. Instrum. Meas., 52: 119-125.

Liu, X., C. Wu, J. Wang and I. Bonev, 2008. Attitude description method of [PP]S type parallel robotic mechanisms. Chinese J. Mech. Eng., 44: 19-23.

Pouliot, N.A., C.M. Gosselin and M.A.Nahon, 1998. Motion simulation capabilities of three-degree-of-freedom flight simulators. J. Aircraft, 35: 9-17.

Selvakumar, A.A., R.P. Babu and R. Sivaramakrishnan, 2012. Simulation and singularity analysis of 3-PRS parallel manipulator. Proceedings of the 9th IEEE International Conference on Mechatronics and Automation, August 5-8, 2012, Chengdu, China, pp: 2203-2207.

Tsai, M.S., T.N. Shiau, Y.J. Tsai and T.H. Chang, 2003. Direct kinematic analysis of a 3-PRS parallel mechanism. Mech. Mach. Theory, 38: 71-83.

Tsai, M.S. and W.H. Yuan, 2011. Dynamic modeling and decentralized control of a 3 PRS parallel mechanism based on constrained robotic analysis. J. Intell. Rob. Syst., 63: 525-545.

Wahl, W., 2000. Articulated tool head. WIPO Patent No. WO2000025976 A3.

Yu, J., Y. Hu, S. Bi, G. Zong and W. Zhao, 2004. Kinematics feature analysis of a 3-DOF compliant mechanism for micro manipulation. Chinese J. Mech. Eng., 17: 127-131.