ISSN: 0975-833X

Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 9, Issue, 05, pp.49806-49814, May, 2017

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

RESEARCH ARTICLE

DESIGN AND KINEMATIC ANALYSIS OF A SIMPLY VERTICAL WINDOW CLEANING ROBOT

[1,*]Erol Türkeş, [2]Serkan Beller, [1]Olcay Ekşi, [1]Sencer Karabeyoğlu and [1]İsmet Tıkız

[1]Kırklareli University, Engineering Faculty, Mechanical Engineering Department, Kırklareli/Turkey
[2]Çukurova University, İmamoğlu Technical Science College, Gas and Fitting Technologies Department, Adana/ Turkey

ARTICLE INFO

Article History:
Received 13th February, 2017
Received in revised form
15th March, 2017
Accepted 07th April, 2017
Published online 19th May, 2017

Key words:

Cleaning, Cartesian Coordinate,
Robot Design.

ABSTRACT

Today, common types of industrial robots are used for many different applications and in many different areas. Designing and producing a prototype robotic system requires knowledge of the system requirements and coordinated work. All of these processes can be realised by working in partnership with mechanical, electronic and computer engineers. In this study, a robotic prototype (including software) request was designed with a Cartesian based coordinate structure at very low cost with an initial goal of cleaning a window or any further tasks to be determined by the end user. This robot is designed to be used in daily life, as well as the existing building design based on the required software. Therefore, the robot's current structure is quite promising.

Copyright©2017, Erol Türkeş et al. *This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.*

Citation: Erol Türkeş, Serkan Beller, Olcay Ekşi, Sencer Karabeyoğlu and İsmet Tıkız, 2017. "Design and kinematic analysis of a simply vertical window cleaning robot", *International Journal of Current Research*, 9, (05), 49806-49814

INTRODUCTION

Robot technology is being developed by researchers and industry every day. The reason is that robots are being asked to work instead of people in more difficult and dangerous jobs, including in construction, manufacturing, and security, because they are able to adapt to different environments and situations. Robots have conquered nearly all the environments in which they have been tested. There is an increasing requisition for the improving of various type service robots to ease humans from risky jobs, such as fire rescue, cleaning the glass-surfaces of skyscrapers, and the inspection of high walls and pipes (Sun *et al*., 2004). Recently, research investigating housework automation has attracted the attention of many researchers. One of the most desired functions is house cleaning. In fact, many cleaning robots have been commercialised. However, previous cleaning robots have not seen widespread use in ordinary families, as there are still a significant number of imperfect elements. The important functions of a cleaning robot include garbage removal, autonomous movement, and determining areas that should be cleaned. Cleaning a residential window and/or glass-surfaces in a house is considerably easier in theory than in reality. The first is that most windows or glass surfaces are too long to reach outside the house, and someone needs to extend the arm out of the house to clean them. Trying to clean the outer surface of a window glass surface from the inside requires a strange move. Because a person will have to access the top or bottom of the window, and will only be able to clean accessible parts of the window. Thus, this approach can also be a chaotic effort, especially for the glass cleaner. Additionally, the handicapped community encounters even greater difficulty in cleaning windows due to limited mobility and motion restrictions. The architecture of today's modern buildings is a complex and exaggerated form. The complexity of the building geometry makes it impossible to reach all parts of the building for cleaning, maintenance and common repairs. It is very important to clean the large glass facades regularly. The task of cleaning glass surfaces of large and complex buildings requires special devices to be developed (Takeshita *et al*., 2015; Koren *et al*., 2015). In this study, we developed a small-size and lightweight three axis (in Cartesian coordinates) window cleaning robot. A prototype of the window cleaning robot was developed. The reach dimensions of the prototype robot are approximately 400 mm x 300 mm x 100 mm, and its weight is approximately 4 kg. Both hardware and software were designed. The hardware deals with the mechanical design and construction, the electronics and the circuitry. The software deals with the programming of the Programmable Logic Controller (PLC) to direct the motion of the processes. The prototyped robot consists of three independently driven step motors and a fixed DC motor. The control system includes one microprocessor card and a short range distance sensor for autonomous

***Corresponding author:* Erol Türkeş,**
Kırklareli University, Engineering Faculty, Mechanical Engineering Department, Kırklareli/Turkey.

operation. The programming language used was C and C++. Additionally, an Arduino IDE compiler with an open-source code was used. This study describes the design of a window cleaning robot, installation of a traveling control system and reports the quantitative results of basic traveling experiments and autonomous wiping motions on a primarily vertical window.

Designing the window cleaning robot

The window cleaning robot system consists of hardware and software. The hardware deals with the mechanical design and construction, electronics and circuitry. The software deals with the programming of the PLC to control the motion of the processes. Figure 1 shows a block diagram of the designed system. The mechanical design and construction, as well as the electronics and circuitry of the window cleaning robot, are presented.

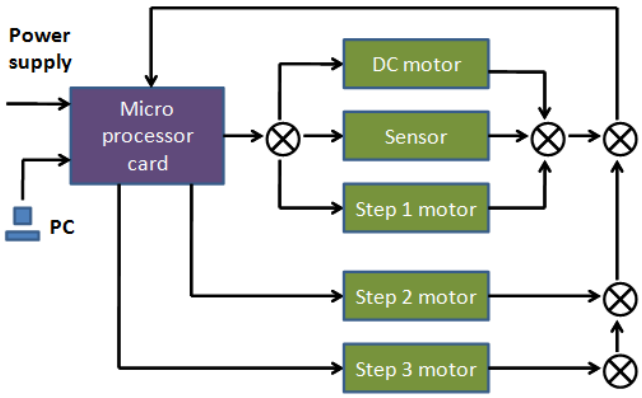

Fig.1. Block diagram of the system

The mechanical system for the robot was designed with a focus on a window cleaning robot for a single windowpane. It is necessary to cross over the window frame or joint line in order for it to be applicable to any window. However, single windowpanes, such as a show window, also are an important potential application. All of the robotic system elements are mounted on a stable resistance platform. Telescopic rails are used for the robot's linear movements. Linear movements in the X, Y and Z axes are performed using ballscrews. Electric motors and ballscrew connections are provided by servo couplings. In order to carry out the cleaning work on the glass surface, the cleaning robot should know when to start or stop the cleaning process and how to control the orientation. For this reason, the distance between the robot's direction, the distance between the robot and the window frame, and the distance between the robot and the point to be cleaned (Sun *et al*., 2004). Some recent work on sensing systems of cleaning robots has been documented (Simoncelli *et al*., 2000; Kurazume and Hirose, 2000). In (Simoncelli *et al*., 2000), ultrasonic sonar was used for automatic localisation. In (Kurazume and Hirose, 2000), a "cooperative positioning system" was proposed to repeat a search until it reached the target position. Nevertheless, for a glass surface climbing robot, many conventional methods, such as laser and ultrasonic sensors, do not apply to measuring the distance between the robot and the window frame. In this prototype, the control system consists of one microprocessor card and one short range distance sensor. The robot was designed for cleaning a single windowpane. Crossing over the window frame or the joint line is required in order to be applicable to any window. However, single windowpanes, such as a show window, are also an important application. Figure 2 shows a potential traveling path that can be sued to sweep over the entire window plane.

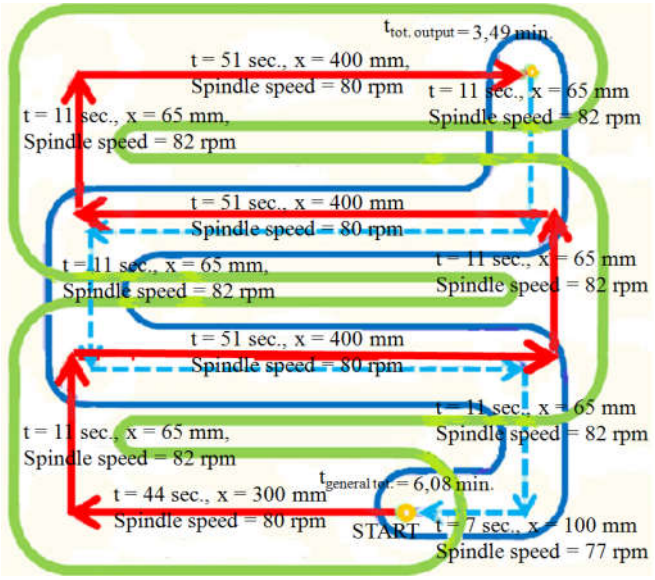

Fig.2. Working path for the proposed window cleaning robot

The desired traveling trajectory consists of horizontal and vertical lines, as shown in Fig. 2. There are a total of four DC motors used in this prototype. These step motors are 57-mm Series YAKO 2-phase hybrid stepping motors. First DC motor is used to rotate the wiper. The first step motor is used to approach and withdraw from the windowpane. The other two step motors are used to move the wiping arm up and down and move the wiper left and right. A short range distance sensor is used to determine the position with respect to the windowpane. This sensor is an infrared proximity sensor and has an analog output that varies from 3.1 V at 3 cm to 0.3 V at 30 cm and a supply voltage between 4.5 and 5.5 VDC.

YAKO 2D3 the DC motor driver system. This model is a constant torque driver with microstep capabilities and a voltage range from 12-48 VDC with a single power input. This model can match a rated current less than 3.0 A and has a flange size that ranges from 42 to 57 mm. Due to the bipolar constant current chopping circuit, the motor can run smoothly with hardly any noise. Increasing the voltage can greatly improve the high speed performance and output torque of the motor. Once the pulse stops for 100 ms, the phase current is automatically cut in half, which reduces the chances of the motor overheating. The driver can be operated using microstepping for low speed operations. The microprocessor card of the control system is an Arduino Duemilanove. This card is a microcontroller board based on an ATmega328 (datasheet) or ATmega168 (datasheet). The card has six analog inputs, 14 digital input/output pins (of which 6 can be used as PWM outputs), a 16 MHz crystal oscillator, a power jack, a USB connection, a reset button, and an ICSP header. This card contains everything needed to support a microcontroller used in the system. To start this microcontroller, it is sufficient to connect a computer with a USB cable or power it with an AC-DC adapter or battery. Arduino is an open source electronic prototyping platform based on flexible, easy to use hardware and software. The Arduino can feel the environment through the input field of various sensors and can influence the

environment by motors, controlling lights and other actuators. The microcontroller on the board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can communicate alone or with software running on a computer. The Arduino Duemilanove can be programmed using the Arduino software. The ATmega328 or ATmega168 on the Arduino Duemilanove comes preprogrammed with a bootloader that allows the user to upload new code without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files) (http://arduino.cc/en/Main/arduinoBoardDuemilanove).

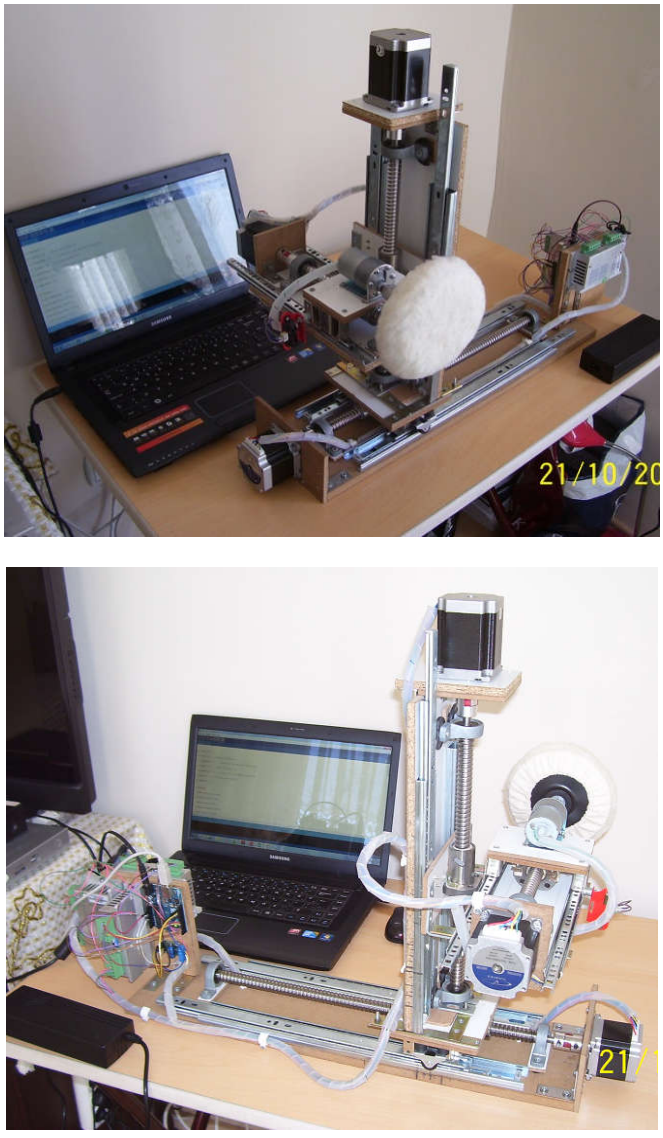

Fig.3. Prototype window cleaning robot

Kinematic analysis of the window cleaning robot mechanism

The kinematic structure term covers all the properties of a mechanism determined by the interconnection pattern between the constituent links of a mechanism alone, and is therefore independent of the properties of the mechanisms. The work of the kinematic constructions attracted the interest of over a hundred researchers and continued to do so. The characteristics of mechanisms are completely determined by the design of interconnection among the links. Degree of freedom is important in the design of robot mechanisms. For this reason the kinematic chains of the mechanism should be designed as perfectly as possible. After the appropriate design of the kinematic chain of the mechanism, a suitable robot driving mechanism for the system is designed. Finally, motion analysis of this driving mechanism can be done easily. While the connections to the rotational pairs have traditionally been of relatively greater interest, several similar studies have been conducted, including types of non-rotating pairs and other mechanisms. The study of these characteristics in the literature has been given such terms as Gruebler synthesis, type synthesis, structural synthesis, number synthesis, structural analysis, systematics, classification and enumeration, census of linkages, and conceptual design (Mruthyunjaya, 2003). Firstly develop an objective function for mechanism and later reach a result using numerical methods with an algorithm is most applied method for the optimal kinematic design of parallel mechanisms (Stock and Miller, 2004; Ryu and Cha, 2003; Chablat and Wenger, 2003). There are general disadvantages for these methodologies. These disadvantages occur as the objective function being highly non-linear and the process being iterative and time consuming. Nevertheless, these methodologies applied to the solution may provide an optimal result, but for such a solution it is not known how good the result is. The most acceptable designs for the design of any machine are generally achieved from comparison. Use of performance charts is applied for this method and they are widely used in classical design and most design manuals. Through the use of performance charts, the interactions of various criteria can be identified widely known. There are various design parameters of any mechanism to be designed. These design parameters are usually link lengths, having a value between zero and infinity. To determine a performance chart in a finite space, it is necessary to normalise the design parameters involved. The most commonly used normalisation approach is dividing all parameters by a chosen parameter (Cervantes-Sa´nchez *et al*., 2001; Mermertas, 2004; Liu and Wang, 2007).

However, this type of normalisation cannot ensure that the normalised parameters are finite. Because of this, the advanced design workspace cannot be used to draw performance charts. Apparently, the most major accuracy measure for a parallel robot would be the maximum position and orientation errors over a given portion of the workspace or for a given nominal configuration, given actuator inaccuracies. Recently, a general design method based on interval analysis has been proposed to calculate near values of the maximum output error in a certain part of the work area, which is the most important information of a designer. Nevertheless, implementing this method is relatively difficult. Because, this method is not provides information on the evolution of the accuracy of the manipulator within its workspace, and is not provides kinematic insight into the optimal design (Merlet, 2006; Briot and Bonev, 2008; Briot and Bonev, 2010). Kinematic model of a manipulator defined the relationship between end-effector motion and the joint displacements of manipulator. Kinematic models are expressed of position, velocity and acceleration formulations all joints and end-effector of manipulator. The position kinematic analysis model relates the joint and end-effector positions. A direct kinematics analysis calculates the end-effector posture from the joint positions given, whereas an inverse kinematics analysis obtains the joint positions necessary to establish a desired end-effector posture. Both analyses are significant for motion analysis studies. The direct analysis is used in design

and simulating manipulator and robotic kinematic chains, whereas the indirect analysis is essential for motion planning resolution algorithms. There are two methods commonly used for kinematic modelling. The first of both methods based on the Denavit–Hartenberg convention and another based on screw theory. The Denavit-Hartenberg is generally used in the literature and the screw theory approach is less known (Rocha *et al*., 2011). Screw theory is a method used in static and kinematic analysis of rigid bodies and mechanisms. Its origins date back to Mozzi and Chasles studies. In 1900, it was systematised by Ball. Later, Hunt, Phillips, Roth e Tsai, among others, and employed it to study mechanisms (Hunt, 2000; Dai, 2006; Ceccarelli, 2000). This theory in kinematic analysis of mechanisms is largely used to velocity analysis. However, Tsai developed a method to represent the poses of a kinematic chain based on Chasles' Theorem (Tsai, 1999). Manipulator of a mobile robot almost always runs beyond the edge of the chassis and should reach the height of the robot's body from ground level. This requirement means the manipulator arm works from the inside or from one side of the work envelope. Certain industrial gantry manipulators work from outside their work envelope, making it difficult to use their layouts on a mobile robot. As shown in Fig. 4, gantry manipulators are Cartesian or rectangular manipulators. This geometry resembles a three dimensional XYZ coordinate system. In fact, that geometry is how the manipulator is controlled and how the working end moves within the work envelope.

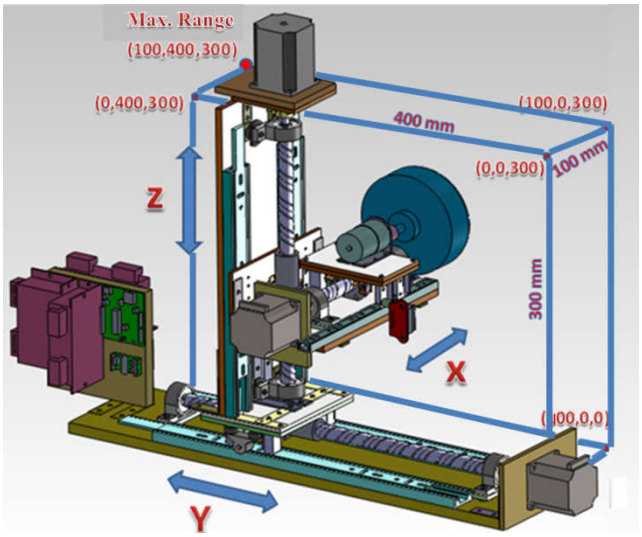

Fig.4. The cantilevered 3-DOF manipulator geometry

A cantilevered layout of the window cleaning robot with three DOFs is shown in Fig. 4. Although it is not the best solution to the problem for working off the front of a robot, the setup will suffice for this work and has the benefit of requiring a very simple control algorithm. The spatial counterpart of this equation is attributed to Kutzbach and several versions of the generalised equations have been suggested in discussions regarding degrees of freedom (F) in the literature (Mruthyunjaya, 2003):

$$F = b(N-1-J)+\sum_{i=1}^{J} f_i , \qquad (3.1)$$

where J and N represent the simple joints and number of links in the kinematic chain, the b represents the degrees of freedom of space or the order for the equivalent screw system (e.g., $b=3$ for planar motion and $b=6$ for general spatial motion), and f_i is the degree of freedom of the ith joint. In the window cleaning robot mechanism shown in Fig. 4, fixed platform 1, ballscrew system 2, moving platform 3 in the Y axis, ballscrew system 4, moving platform 5 in the Z axis, ballscrew system 6, moving platform 7 in the X axis are defined with numbers. The total numbers of links and joints in the chain in the mechanism are N=7 and J=9. The mechanism performs planar motions in each plane, meaning that $b=3$ and the total f_i, which is the degrees of freedom of the ith joint is $f_{tot}=12$. Therefore, the obtained number of degrees of freedom (F) for the window cleaning robot mechanism, as determined from Eq.(3.1) is 3. The maximum displacements and range points for the window cleaning robot mechanism are given by the 3D-Cartesian coordinate system shown in Fig. 5.

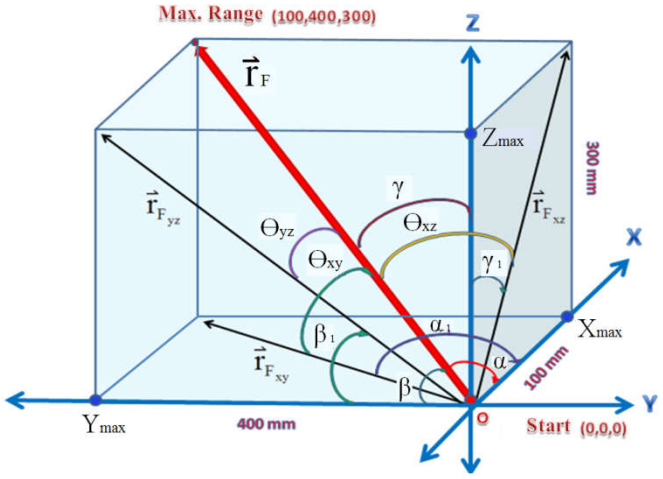

Fig.5. 3D-Cartesian coordinate system for the mechanism

Kinematic analysis of robots can be quite difficult and complex. There are two types of kinematic analysis methods. These are direct kinematics and inverse kinematics. Both direct and inverse kinematics problems can be solved by various methods, including geometric vector analysis, matrix algebra, screw algebra, and numerical integration. In this study, two kinematic analysis approaches investigated. These are screw algebra and geometric vector analysis. According to Fig. 5, the kinematic equations of the robot mechanism in terms of screw theory can using the following.

Assuming three arbitrary unit screws in space,

$$E_i=\left(\tilde{L}_i,\tilde{M}_i,\tilde{N}_i\right), E_j=\left(\tilde{L}_j,\tilde{M}_j,\tilde{N}_j\right), E_k=\left(\tilde{L}_k,\tilde{M}_k,\tilde{N}_k\right), \qquad (3.2)$$

where $E_{i,j,k}$ are unit screws the in X, Y, and Z axes. In general: $E=e+we^0$, where e is the unit vector of the current screw axis, e^0 is the moment of e with respect to the origin of the coordinate system, and w is the Clifford operator (w^2=0). Where, screw algebra is the vector algebra of this dual vector. A screw can be described using three dual coordinates, such as: $E=\left(\tilde{L},\tilde{M},\tilde{N}\right)$. Each dual coordinate of a screw

can be divided into two parts: $\tilde{L} = L + wP$, $\tilde{M} = M + wQ$, and $\tilde{N} = N + wR$. The six real coordinates of a screw are known as the Plücker coordinates of the unit screw **E** (*L*, *M*, *N*, *P*, *Q*, *R*). The variables *L*, *M*, and *N* are the components of the unit vector e and *P*, *Q*, and *R* are the components of the moment vector e^0. The moment of the unit vector component e^0 is defined as the vector product of the radius vector and the unit vector as $e^0 = \rho \times e$, where $\rho = \rho(x, y, z)$ is the radius vector to an arbitrary point on the screw axis E. Dual angles between those three unit screws are defined as $A_{ij} = \alpha_{ij} + w a_{ij}$, $A_{jk} = \alpha_{jk} + w a_{jk}$, and $A_{ki} = \alpha_{ki} + w a_{ki}$, where α_{ij}, α_{jk}, and α_{ki} are the angles between the corresponding screw axis and a_{ij}, a_{jk}, and a_{ki} are shortest distances between the corresponding screw axis (Bayram, 2003). These are defined as $\alpha_{ij} = \alpha_{jk} = \alpha_{ki} = \pi/2$ and $a_{ij} = a_{jk} = a_{ki} = 0$, as shown in Figs.4 and 5. Additionally, each ballscrew acts independently in the window cleaning robot mechanism. Thus, each dual coordinate of a screw can be written into the X, Y, and Z axes for each ballscrew as

$$\tilde{L} = L + wP,$$
$$\tilde{M} = M + wQ,$$
$$\tilde{N} = N + wR. \quad (3.3)$$

The velocity and acceleration of a unit screw in space can be obtained from the first and second derivatives of Eq. (3.3) taken with respect to time:

$$\dot{\tilde{L}} = \dot{L} + w\dot{P},\ \dot{\tilde{M}} = \dot{M} + w\dot{Q},\ \dot{\tilde{N}} = \dot{N} + w\dot{R}, \quad (3.4)$$

$$\ddot{\tilde{L}} = \ddot{L} + w\ddot{P},\ \ddot{\tilde{M}} = \ddot{M} + w\ddot{Q},\ \ddot{\tilde{N}} = \ddot{N} + w\ddot{R}. \quad (3.5)$$

From equations (3.4) and (3.5), we can write the time derivatives of the Plücker coordinates of a unit screw as

$$\dot{E} = \left(\dot{L}, \dot{M}, \dot{N}, \dot{P}, \dot{Q}, \dot{R}\right), \ddot{E} = \left(\ddot{L}, \ddot{M}, \ddot{N}, \ddot{P}, \ddot{Q}, \ddot{R}\right). \quad (3.6)$$

According to Fig. 5, the kinematic equations for the geometric vector analysis of the robot mechanism can be written using the following (Beller, 2012)

For the displacement analysis of the end effector, the written vector equation is

$$\vec{r}_F = \vec{r}_{F_{XY}} + \vec{r}_{F_{YZ}} + \vec{r}_{F_{XZ}}, \quad (3.7)$$

$$\vec{r}_F = r_F e^{j\theta_{XY}} + r_F e^{j\theta_{YZ}} + r_F e^{j\theta_{XZ}},$$

$$\vec{r}_F = r_F \cos\theta_{XY} + j\, r_F \sin\theta_{XY} + r_F \cos\theta_{YZ} + j\, r_F \sin\theta_{YZ} + r_F \cos\theta_{XZ} + j\, r_F \sin\theta_{XZ}$$

,

where;

$$\vec{r}_{F_{XY}} = r_{F_{XY}} e^{j\alpha_1}, \vec{r}_{F_{XZ}} = r_{F_{XZ}} e^{j\gamma_1}, \vec{r}_{F_{YZ}} = r_{F_{YZ}} e^{j\beta_1}.$$

The linear and/or angular velocities and accelerations of the end effector with respect to the starting point (O) can be obtained from the first and second derivatives of Eq. (3.7) taken with respect to time:

$$\dot{\vec{r}}_F = \dot{\vec{r}}_{F_{XY}} + \dot{\vec{r}}_{F_{YZ}} + \dot{\vec{r}}_{F_{XZ}}, \quad (3.8)$$

$$\dot{\vec{r}}_F = \dot{r}_F e^{j\theta_{XY}} + r_F j\omega_{XY} e^{j\theta_{XY}} + \dot{r}_F e^{j\theta_{YZ}} + r_F j\omega_{YZ} e^{j\theta_{YZ}} + \dot{r}_F e^{j\theta_{XZ}} + r_F j\omega_{XZ} e^{j\theta_{XZ}}$$

,

where:

$$\dot{\vec{r}}_{F_{XY}} = \dot{r}_{F_{XY}} e^{j\alpha_1} + r_{F_{XY}} j\omega_{\alpha_1} e^{j\alpha_1}, \dot{\vec{r}}_{F_{XZ}} = \dot{r}_{F_{XZ}} e^{j\gamma_1} + r_{F_{XZ}} j\omega_{\gamma_1} e^{j\gamma_1},$$
$$\dot{\vec{r}}_{F_{YZ}} = \dot{r}_{F_{YZ}} e^{j\beta_1} + r_{F_{YZ}} j\omega_{\beta_1} e^{j\beta_1},$$

$$\ddot{\vec{r}}_F = \ddot{\vec{r}}_{F_{XY}} + \ddot{\vec{r}}_{F_{YZ}} + \ddot{\vec{r}}_{F_{XZ}}, \quad (3.9)$$

$$\begin{aligned}\ddot{\vec{r}}_F = {} & \ddot{r}_F e^{j\theta_{XY}} + 2\dot{r}_F j\omega_{XY} e^{j\theta_{XY}} + r_F j\xi_{XY} e^{j\theta_{XY}} - r_F \omega_{XY}^2 e^{j\theta_{XY}} \\ & + \ddot{r}_F e^{j\theta_{YZ}} + 2\dot{r}_F j\omega_{YZ} e^{j\theta_{YZ}} + r_F j\xi_{YZ} e^{j\theta_{YZ}} - r_F \omega_{YZ}^2 e^{j\theta_{YZ}} \\ & + \ddot{r}_F e^{j\theta_{XZ}} + 2\dot{r}_F j\omega_{XZ} e^{j\theta_{XZ}} + r_F j\xi_{XZ} e^{j\theta_{XZ}} - r_F \omega_{XZ}^2 e^{j\theta_{XZ}}\end{aligned}$$

,

where:

$$\ddot{\vec{r}}_{F_{XY}} = \ddot{r}_{F_{XY}} e^{j\alpha_1} + 2\dot{r}_{F_{XY}} j\omega_{\alpha_1} e^{j\alpha_1} + r_{F_{XY}} j\xi_{\alpha_1} e^{j\alpha_1} - r_{F_{XY}} \omega_{\alpha_1}^2 e^{j\alpha_1}$$

,

$$\ddot{\vec{r}}_{F_{XZ}} = \ddot{r}_{F_{XZ}} e^{j\gamma_1} + 2\dot{r}_{F_{XZ}} j\omega_{\gamma_1} e^{j\gamma_1} + r_{F_{XZ}} j\xi_{\gamma_1} e^{j\gamma_1} - r_{F_{XZ}} \omega_{\gamma_1}^2 e^{j\gamma_1},$$
$$\ddot{\vec{r}}_{F_{YZ}} = \ddot{r}_{F_{YZ}} e^{j\beta_1} + 2\dot{r}_{F_{YZ}} j\omega_{\beta_1} e^{j\beta_1} + r_{F_{YZ}} j\xi_{\beta_1} e^{j\beta_1} - r_{F_{YZ}} \omega_{\beta_1}^2 e^{j\beta_1}.$$

In Eq. (3.8) and Eq. (3.9), (ω_n) and (ξ_n) are the angular velocity and acceleration, respectively.

Dynamic analysis of the robot mechanism

Generally, dynamics considers with forces and/or torques required to cause the motion of a system of bodies. Also, it includes inertial forces as a principal concern. The dynamics of robot manipulators are a complex issue that is being studied by many researchers. For example, it is quite complex that an end effector is steered along a certain path with certain predetermined motion properties. Most favorable torque and/or force effect functions must be applied at the actuated joints to produce that most suitable motion for end effector. This driving force and/or torque functions depend not only on the spatial and temporal characteristics of the given path, but also on the mass properties of, for instance, the links, the payload, and the externally applied forces. Because the cartesian parallel mechanism is fully isotropic within the entire workspace, the cartesian parallel mechanism kinematically behaves as a traditional serial X–Y–Z cartesian mechanism. Although the cartesian parallel mechanism has simple kinematics and a regular workspace, the dynamics are more complex than for a serial X–Y–Z cartesian mechanism. Numerical solutions of dynamics of manipulator or robot may be suitable for analysing the characteristics of dynamics. However, these aren't directly applicable for controller design. For this reason, some dynamic modeling approaches presented in the literature have been developed for control design purposes of robots. Therefore, inverse dynamics approach has been solved for general Gough–Steward parallel manipulators. The inverse dynamics method

has also been used for the control of translational parallel manipulators. The Newton–Euler approach to formulate dynamic equations is computationally efficient despite its complexity and has been proposed for a general Gough–Steward parallel manipulator and a three degree of freedom translational parallel manipulators. Although the Lagrange method has been applied to solve a specific five-bar closed chain mechanism, the result is difficult to extend to other parallel manipulators. To the best of the authors' knowledge, an efficient dynamics model for an isotropic translational parallel manipulators to obtain the decoupling dynamic characteristics has not been proposed (Yen and Lai, 2009). First, for the robotic mechanism, the fundamental properties are determined for the cleaning surface. The maximum press forces and friction forces on the surface are obtained from these properties. The ultimate tensile strength (compression) (σ_u) and Young's Modulus (E) are determined for the glass. Compression (F_c) and friction forces (F_f) acting on the rotating wiper of the end effector are shown in Fig. 6.

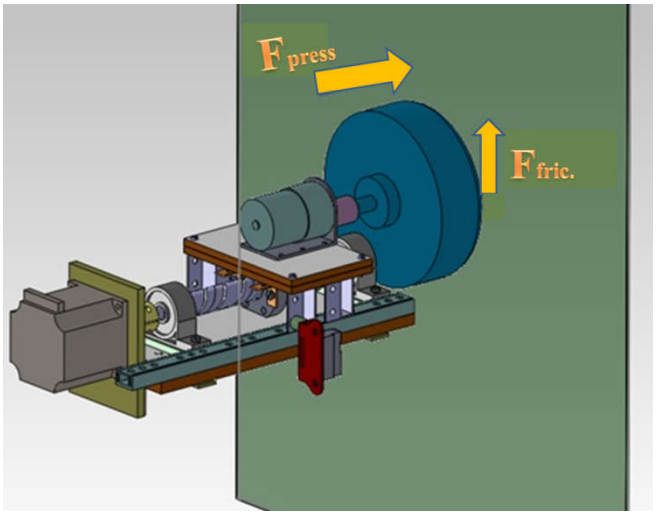

Fig.6. Compression and friction forces on the rotating wiper

In order to determine the DC motor torque on the rotating wiper of the end effector, we use

$$\sigma_u = \frac{F_c}{A} \rightarrow A = \pi r_w^2, \qquad (4.1)$$

$$F_f = F_c\, \mu_{1y},$$

$$T_f = F_f\, r_w,$$

where μ_{1y} is the friction coefficient of a wet glass surface, T_f is the DC motor torque on the rotating wiper and r_w is the radius of the rotating wiper. The DC motor torque is determined by Eq. (4.3).

Henceforward, ballscrew selection should be performed for each axis and should be checked for compliance against the load on the system. If the selected ballscrew is appropriate for the design, the selection of the stepper motor is performed afterwards. The suitability of the stepper motor is controlled by the torque. In these calculations, the diameter (D_b), length (L_b), material (q), step (P_b) and efficiency (η) of the ballscrew should be known. The total sled mass (m), maximum stroke $(S_{\max})$, instant traverse speed $(V_{\max})$, service life of the ballscrew system (L_t), friction coefficient for the sliding surfaces (μ), and maximum motor spindle speed $(N_{\max})$ should also be known. The following calculation procedure is applied for determining the suitability of the spindle diameter and the service life of the ballscrew system.

First, pitch selection $l \geq \frac{V_{\max}}{N_{\max}}$ and the axial load acceleration $\alpha_1 = \frac{V_{\max}}{t_1}$ are determined for the time t_1 for the system to start moving. Hence, for the horizontal and vertical positions of the ballscrews:

the axial load (F_1) at the time (t_1) that the system starts to move is

$$F_{1h} = m_{tot.}\left(\mu\, g + \alpha_1\right),\quad F_{1v} = m_{tot.}\left(g + \alpha_1\right), \qquad (4.4)$$

the axial load (F_2) at a constant speed (t_2) is:

$$F_{2h} = m_{tot.}\mu\, g\ ,\quad F_{2v} = m_{tot.}\, g\ , \qquad (4.5)$$

and the axial load (F_3) during deceleration (t_3) is:

$$F_{3h} = m_{tot.}\left(-\mu\, g + \alpha_1\right),\qquad F_{3v} = m_{tot.}\left(g - \alpha_1\right).$$

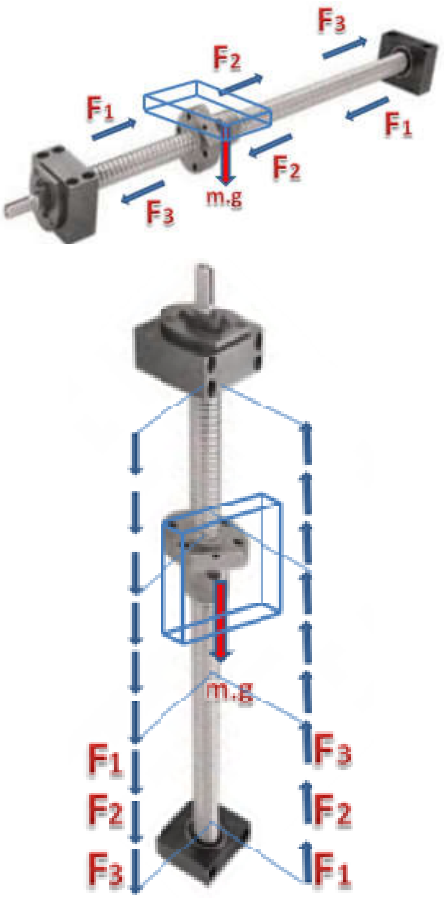

Fig.7. Horizontal and vertical positions of the ballscrews

In this way, the greater axial load is selected as the critical buckling load (P) and determines the diameter of the ballscrew from

$$d_r \geq \left(\frac{P L_b^2}{m_{tot.}} \right)^{1/4} . \quad (4.7)$$

The desired life (L_{tw}) is determined to control the service life using

$$L_{tw} = W_h \, D_y \, Y \, P_{act} \ , \quad (4.8)$$

where W_h is the number of daily working hours, D_y is the number of working days per year, Y is the number of operating years, and P_{act} is the activity percentage. In the above formula, the required working time can be easily calculated for each axis. The axis speed and time graphs in Fig. 8 permit the determination of the operating time and if it meets the requirements for the selected system.

If the result is greater than the necessary L_t time, the current system is suitable for this job. The following variables are defined in the previous equations: (l) is the pitch (mm), α_1 is the axial load acceleration (m/s^2), F_1, F_2, and F_3 are the axial loads (N), L is the shaft length (mm), m is the buckling load factor (see related table), t_a, t_b, and t_c are different times (s), and C_a and f_w are the dynamic and load (see related table) coefficients. These are used to show that the selected ballscrew is suitable to resist the forces in the system. Then, the total moment of inertia is converted to the stepper motor axis to determine the size of the motor that is required to drive this portion of the system. After the stepper motor is selected for each axis, its parameters must be replaced in the calculation for the driving torque. The effective torque must be controlled.
The moment of inertia of the ballscrew is defined as

$$J_B = \frac{\pi \gamma D^4 L}{32} . \quad (4.12)$$

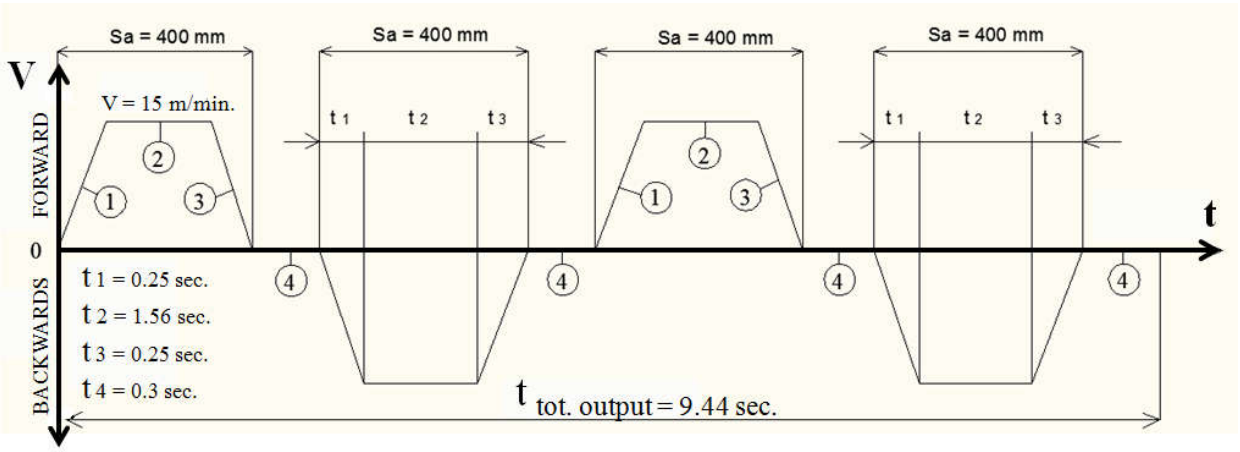

Fig.8. Speed and time graphs for a single axis

The chart shows the axial loads, rotational speed and elapsed time during the acceleration period, constant velocity period and deceleration period, which are defined as
Average load:

$$F_m = \left(\frac{F_1^3 \, N_1 t_a + F_2^3 \, N_2 t_b + F_3^3 \, N_3 t_c}{N_1 t_a + N_2 t_b + N_3 t_c} \right)^{1/3} ,$$

Average operation speed:

$$N_m = \frac{N_1 t_a + N_2 t_b + N_3 t_c}{t_1 + t_2 + t_3 + \ldots + t_n} , \quad (4.10)$$

Life time calculation:

$$L_t = \left(\frac{C_a}{F_m \, f_w} \right)^3 \frac{10^6}{60 \, N} . \quad (4.11)$$

The moment of inertia of the moving parts is defined as

$$J_W = m \left(\frac{1}{2\pi} \right)^2 . \quad (4.13)$$

The total moment of inertia on the ballscrew axis is defined as

$$J_L = J_B + J_W + J_C \ , \quad (4.14)$$

(4.9)

where J_C is the coupling inertia.

Choosing the electrical motors

The moment of inertia (J_M) and the holding torque (T_M) must be known for choosing the industrial stepping motor. This provided torque is required to continuously rotate the ballscrew at the same speed against any external loads:

$$T_1 = T_a + T_{p\max} + T_u \ , \quad (5.1)$$

where T_a is the driving torque at constant speed (Nm), $T_{p\max}$ is the upper limit of the dynamic friction torque for the ballscrew (Nm from the related tables), and T_u is the friction torque for the support bearings (Nm). According to the number of rolling bearings, T_u is scaled up to determine the total frictional torque (from the related tables). As shown in Fig. 9, F_s acts in the direction parallel to the current axis and originates outside the system from the frictional force on the glass acting against the shaft by the rotating action of the wiper $F_{cutting} = F_s$.

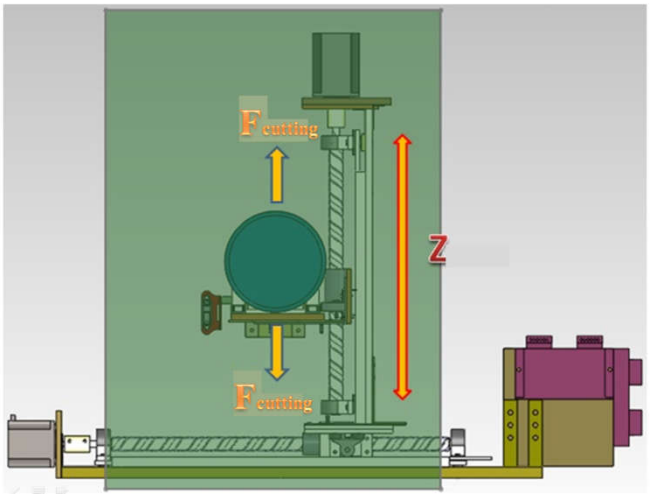

Fig. 9. Forces acting parallel to the current axis

The driving torque calculation is defined as

$$T_a = \frac{F_a\, l}{2\pi\, \eta_1}. \qquad (5.2)$$

The force acting in the direction of the horizontal axis is defined as

$$F_a = F_s + \mu\, m\, g. \qquad (5.3)$$

The force acting in the direction of the vertical axis is defined as

$$F_a = F_s + \mu\, m\, g + m\, g. \qquad (5.4)$$

The driving torque required for continuous speed operations is defined as

$$T_1 = \frac{F_a\, l}{2\pi\, \eta_1} + T_{p\max} + T_u. \qquad (5.5)$$

The driving torque during the acceleration period is defined as

$$T_2 = T_1 + \frac{J\, 2\pi\, n}{60\, t_1} \quad \rightarrow J = J_L + J_M. \qquad (5.6)$$

The driving torque during the deceleration period is defined as

$$T_3 = T_1 - \frac{J\, 2\pi\, n}{60\, t_3} \quad \rightarrow \quad J = J_L + J_M. \qquad (5.7)$$

The effective torque control is defined as

$$T_{rms} = \left(\frac{T_1^2\, t_a + T_2^2\, t_b + T_3^2\, t_c}{t_{tot.}} \right)^{1/2}. \qquad (5.8)$$

Motor selection criteria includes: maximum operational speed $N_M \geq 3000$ (min^{-1}) and the motor holding torque: $T_M \geq T_{rms}$ (Nm). The motor rotor inertia J_M must be greater than $J_L / 3$; $J_M > J_L / 3$ ($kg.m^2$). Apart from these parameters, the amount of time for the selected engine to reach the maximum speed will determine if it should be used in this system:

$$t_a = \frac{\left(J_L + J_M\right) 2\pi\, n}{\left(T_M^{'} - T_1\right)} 1,4 \quad \rightarrow \; T_M^{'} = 2\, T_M. \qquad (5.9)$$

Conclusion

In this study, a three-axis Cartesian type robot prototype for window cleaning was designed and manufactured. The work on this window cleaning robot consisted of hardware and software design. The hardware portion dealt with the mechanical design and construction, electronics and circuitry. The software portion dealt with the programming of the Programmable Logic Controller (PLC) to control the motion of the cleaning process. First, robot design elements were determined and confirmed later using dynamic and kinematic analyses. Subsequently, motor selection for the different robotic axes was performed and the manufacturing process was begun. Although the present application is temporary, it is highly suitable for daily use of the system with the hardware and software arrangements. The feed direction of the rotating wiper is shown in Fig. 2. An operating speed of 80 rpm was used instead of the maximum operating speed of 3000 rpm. Red arrows inside the green line in Fig. 2 show an upward output direction for the rotating wiper during window cleaning and the light blue arrows inside the blue line show the downward landing direction of the rotate wiper during window cleaning. A full cleaning operation using the rotating wiper can be completed in approximately 6 minutes. This robot can be used in many other aspects of daily life, as well as for existing building designs if software changes are made. Therefore, the robot's current structure is quite promising.

REFERENCES

Bayram, Ç. 2003. *Kinematic and Dynamic Analysis of Spatial Six Degree of Freedom Parallel Structure Manipulator*, A Thesis Submitted to the Graduate School in Partial Fulfillment of the Requirements for the Degree of Master of Science, İzmir Institute of Technology, İzmir, Turkey.

Beller, S. 2012. *A Cleaning Window Robot and Model Application*, A Thesis Submitted for the Degree of Master of Science, Dumlupinar University, Institute of Science, Kütahya, Turkey.

Briot, S. and Bonev, I.A. 2010. Accuracy analysis of 3T1R fully-parallel robots, *Mechanism and Machine Theory*, vol.45, p.695–706.

Briot, S. and Bonev, I.A., 2008. Accuracy analysis of 3-DOF planar parallel robots, *Mechanism and Machine Theory*, vol.43, p.445–458.

Ceccarelli M. 2000. Screw axis defined by giulio mozzi in 1763 and early studies on helicoidal motion. *Mechanism and Machine Theory*, vol.35, p.761–70.

Cervantes-Sa´nchez, J.J., Herna´ndez-Rodrı´guez, J.C., Angeles, J. 2001. On the kinematic design of the 5R planar, symmetric manipulator, *Mechanism and Machine Theory*, vol.36, p.1301–1313.

Chablat, D. and Wenger, Ph., 2003. Architecture optimization of a 3-DoF parallel mechanism for machining applications: the Orthoglide, *IEEE Transactions on Robotics and Automation*, vol.19, no.3, p.403–410.

Dai, J. 2006. An historical review of the theoretical development to frigid body displacements from rodrigues parameters to the finite twist. *Mechanism and Machine Theory*, vol.41, p.41–52.

http://arduino.cc/en/Main/arduinoBoardDuemilanove accessed on October 2015.

Hunt, K. 2000. Don't cross-thread the screw. *A symposium commemorating the legacy, Works and life of sir Robert stawell ball upon the 100th anniversary of a treatise on the theory of screws*, University of Cambridge—Trinity College. Cambridge, United Kingdom: Cambridge University Press; p.1–37.

Koren,Y., Alexander, E., Chow, Y.L., Cohen, J., Jeske,S., Winrobo Window-Washing Robot *http://deepblue.lib.umich.edu/bitstream/handle/2027.42/57935/me450f07project6_report.pdf* accessed on October 2015.

Kurazume, R. and Hirose, S. 2000. Development of a cleaning robot system with cooperative positioning system. *Auton Robots*; vol.9, no.3, p.237–46.

Liu, X.J. and Wang, J. 2007. A New Methodology for Optimal Kinematic Design of Parallel Mechanisms. *Mechanism and Machine Theory*, vol.42, p.1210–1224.

Merlet, J.-P. 2006. Computing the worst case accuracy of a PKM over a workspace or a trajectory, *The 5th Chemnitz Parallel Kinematics Seminar*, Chemnitz, Germany, p.83–96.

Mermertas, V. 2004. Optimal design of manipulator with four-bar mechanism, *Mechanism and Machine Theory,* vol.39, p.545–554.

Mruthyunjaya, T.S. 2003. Kinematic structure of mechanisms revisited, *Mechanism and Machine Theory*, vol.38, p.279–320.

Rocha, C.R, Tonetto, C.P. and Dias, A. 2011. A comparison between the Denavit–Hartenberg and the screw-based methods used in kinematic modeling of robot manipulators, *Robotics and Computer-Integrated Manufacturing*, vol.27, p.723–728.

Ryu, J. and Cha, J. 2003. Volumetric error analysis and architecture optimization for accuracy of HexaSlide type parallel manipulators, *Mechanism and Machine Theory*, vol.38, p.227–240.

Schraft, R.D., Brauning, U., Orlowski, T., Hornemann, M. 2000. Automated cleaning of windows on standard facades, *Automation in Construction*, vol.9, p.489–501.

Simoncelli, M., Zunino, G., Christensen HI, Lange K. 2000. Autonomous pool cleaning: self-localization and autonomous navigation for cleaning. *Auton Robots*, vol.9, no.3, p.261–70.

Stock, M. and Miller, K. 2004. Optimal kinematic design of spatial parallel manipulators: application to linear Delta robot, *Journal of Mechanical Design*, vol.125, p.292–301.

Sun, D., Zhu, J., Lai, C., Tso, S.K. 2004. A visual sensing application to a climbing cleaning robot on the glass surface, *Mechatronics*, vol.14, p.1089–1104.

Takeshita, T., Tomizawa, T., Ohya, A. A House Cleaning Robot System: Path indication and Position estimation using ceiling camera, *www.tomy3.com/Research/ICCAS 2006-BEN.pdf* accessed on October 2015.

Tsai, L-W. 1999. *Robot analysis: the mechanics of serial and parallel manipulators*. New York: Wiley-Interscience.

Yen, P-L. and Lai, C-C. 2009. Dynamic modeling and control of a 3-DOF Cartesian parallel manipulator, *Mechatronics*, vol.19, p.390–398.

****