A New Optimal Method for Calculating the Null Space of a Robot using NOC Algorithm; Application on Parallel 3PRS Robot

Hami Tourajizadeh*
Department of Mechanical Engineering, Faculty of Engineering,
University of Kharazmi, Tehran, Iran
E-mail: Tourajizadeh@khu.ac.ir
*Corresponding author

Oveas Gholami
Department of Mechanical Engineering, Faculty of Engineering,
University of Kharazmi, Tehran, Iran
E-mail: oveas.gholami@gmail.com

Received: 1 May 2019, Revised: 16 August 2019, Accepted: 16 November 2019

Abstract: In this paper, a new optimal method for modelling of a 3PRS robot is proposed according to NOC algorithm. An optimal method of selecting the generalized coordinate is presented and a new algorithm of extracting the null space of over and under constrained robots is proposed through which a lower amount of mathematical calculations is required. In this method, using the principal of derivatives of implicit functions, the null space of constraint matrix will be extracted. Afterwards the null space matrix is calculated with orthogonal columns. The proposed method is implemented on a 3PRS robot which is an under constrained robot. This robot is a kind of parallel spatial robot with 6 DOFs which can be controlled using 3 active prismatic joints and 3 passive rotary ones. This robot similar to other parallel robots has heavy, complicated and nonlinear model which needs heavy and time consuming mathematical calculations. The proposed strategy of extracting the null space of the robot, extremely and heavily decreases the volume of required mathematical calculations for modelling the robot and consequently decreases the inevitable consumed time of processing and numerical errors and increases the accuracy of simulations.

Keywords: Constrained Robot, Modeling, NOC, Null Space Matrix, 3PRS Parallel Robot

Reference: Hami Tourajizadeh, Oveas Gholami, “A New Optimal Method for Calculating the Null Space of a Robot using NOC Algorithm; Application on Parallel 3PRS Robot”, Int J of Advanced Design and Manufacturing Technology, Vol. 13/No. 1, 2020, pp. 1–15.

Biographical notes: Hami Tourajizadeh received his PhD from IUST in the field of mechanics, branch of control and robotics. More than 35 journal papers, 15 accepted conference papers, 1 published book, 1-chapter book and 2 booked inventions are the results of his researches so far. He has been involved in teaching and research activities for more than 10 years in the field of control and dynamics in different universities and he is now assistant professor of Kharazmi University since 2013. His research interests include robotic, automotive engineering, control and optimization. **Oveas Gholami** received his MSc from Kharazmi University in 2018 in the field of applied mechanical design. His research interests include robotic systems and control and optimization methods.

1 INTRODUCTION

Parallel robots are widely applicable because of their special advantages including their speed, strength, accuracy and load carrying capacity. One of the most popular parallel robots is Stewrat which has 6 actuator supporting 6 spatial DOFs [1]. However, the limited workspace of this robot decreases its applications. A new generation of parallel robots is 3PRS which can be considered as a subset of Stewart robot. This robot can cover a wider workspace compared to its limited actuators. In addition, since the number of connecting links between the base and end-effector is three, the probability of jack interference is lower respect to Stewart case.

The robot has three active prismatic joints and three rotary passive ones and all of the spatial DOFs of the end-effector can be controlled by the aid of the mentioned joints. Thus, the robot is under constrained and needs special consideration for its modeling, control and optimization. Since the robot has an acceptable workspace and accuracy, it has a lot of usages specially in micro applications [2]. However, its extreme nonlinear dynamics and its under constrained entity extremely increases the required mathematical calculations and time consumption for which any optimization method toward decreasing these calculations can be appreciated. Modeling these nonlinear systems is challenging and needs heavy and time-consuming mathematical calculations. Especially since the robot is under constrained, calculating the required jack elongation and force of the actuators through its inverse kinematics and dynamics and also modeling the forward dynamics of the system in order to check the robot performance through its related simulations is challenging considering the fact that its corresponding Jacobian matrix is not square.

In order to perform the mentioned calculations, null-space of the system is required through which the proper independent generalized coordinates of the system should be selected. Thus, the inverse of the non-square Jacobian and inertia matrices can be extracted consequently. The conventional methods of performing such operations need heavy and time consuming mathematical calculations and needs super computers in some cases. The main studies toward modeling of such robots and extracting their corresponding null space is as follow:

Li and Xu have developed the dynamics of the 3PRS using Lagrange and virtual work [3]. The same algorithms are employed in [4] in order to extract the matrix form of dynamic of this robot in order to be used for control systems. However, in these researches no effort was made for eliminating the Lagrange multipliers. Altuzarra et al. extracted the dynamic equation of the system and verified it by the aid of experimental tests [5]. Since the Lagrange equation for this parallel system is too heavy, the employed model of the robot in this research is Boltzmann-Hamel. It should be considered that if the equations can be simplified by eliminating the Lagrange multipliers, the necessity of using the mentioned model can be cancelled.

Nikravesh and Haug have proposed a new practical formulation for modeling the dynamic systems with holonomic and non-holonomic constraints [6]. The constraints and motion equations are written here as a to Descartes formulation through which formulating the constraints and generalize forces can be simplified. In [7] a method is delivered toward separating the singular values of constrained dynamic systems. This method in somewhere is more efficient for minimizing the mathematical calculations rather than Gauss elimination method.

Kim and Vanderploeg have delivered a general algorithm according to separation of velocities for analyzing the dynamical systems. Here the dependent generalized coordinates are extracted from the Jacobian matrix. It is proved that this method is more practical compared to Descartes formulation [8]. In [9] a geometric approach is presented for calculating the dynamic response of constrained systems. Here the dependent coordinates are tangent to constraint surface by converting the generalized velocities to a hyperplane and so the constraints will be satisfied. In 1988, Angeles and Lee have introduced an orthogonal matrix with constraint gain matrix in order to separate the dependent and independent generalize coordinates. However, this method is complicated for nonlinear systems such as parallel robots [10]. Afterwards, Angeles et al. implemented the mentioned algorithm for serial manipulators and showed the efficiency of their method for modeling of this kind of robots [11]. Modeling of constrained systems with non-holonomic constraints is performed in [12].

Terze et al. have modeled the constrained systems using Null Space Integration Method. In this research, the state space of the system is extracted in its minimal form considering the constraints of the system. The proposed method is verified on a biomechanical system. The constraints and its related Lagrange multipliers are eliminated using the null space of the system. The results are compatible with the ordinary solvers of differential equations [13]. In this research, NOC matrix is extracted using the constraints related to speed generalized coordinates. Also, calculation of Eigen values and singular points in order to extract the NOC matrix is avoided. However, the calculation of null space is decreased through the proposed method of this paper, but again it is not completely applicable for heavy nonlinear equations of parallel robots. Pendar et al. have extracted the dynamics of 3RPS robot employing NOC [14]. The proposed method to extract the NOC of the robot is new and applicable to some extent, however calculating the inverse of a matrix of order 2 or 3 is required to extract the NOC which again limits its applicability for the systems which has matrices with long length elements. A parallel robot with 6 sliding joints is investigated in [15] through which the dynamics of the system is extracted employing a Decoupled NOC matrix (DeNOC).

The method of extracting the null space in this paper is based on using the Jacobian matrix and making a relation between derivatives of the dependent and independent variables of the system. This algorithm is not suggested since calculating the inverse of a high order matrix is required here. Phong and Hoang [16] have studied the null space application for simulating the closed loop constrained systems. The Lagrange multipliers are eliminated using the null space of the related Jacobian matrix. The proposed method is robust against singularity conditions and is verified by numerical results. Marino et al. have proposed a non-concentrated structure to control the mobile robots by the aid of null space technique [17]. Here the speed of controlling procedure of the robot is increased as a result of increasing the speed of calculating the null space of the system. Afterwards Raoofian et al. have analyzed the direct dynamics of the parallel robot using DeNOC method [18].

Virtual Spring Method (VSM) is used here to cover the mentioned theory. It should be noticed that though a new method is presented here to employ the NOC matric, no approach is delivered for increasing the speed of calculating of this matrix. Some researchers are focused on simplifying the calculation of NOC matrix which can be useful to make further progress in this way: Coleman and Sorensen have proposed a new method to evaluate the orthonormal basis of the null space of a matrix [19]. Three methods are proposed here that all of which suffer from a same disadvantage. In these methods it is required to replace the rows of the matrix and defining the sign of each element (such as cofactors) which limits the applicability of the proposed method. Berry et al. have delivered a new method for calculating the basis for a null space of a matrix.

The method is based on recursive algorithm according to Gaussian elimination method. However the mentioned method is numerical which is not suitable for analytic applications [20]. In [21] after careful investigation of existing methods for calculation of the null space of a matrix, two methods are proposed. In the first method, the fundamental basis of null space is extracted using embedded identity matrix. In the second approach, the triangular basis of a null space is extracted using an upper triangular matrix. The simulations have illustrated that using these methods, the calculated null space is completely sparse. Also, the second proposed method is time-consuming. In these methods, solving some linear equations are required which is not suitable for parametric problems with heavy calculations. Then Dai and Jones could find a new simple method for solving the set of linear equations without using Gauss–Seidel_algorithm [22]. Afterwards they proposed a method based on cofactors to calculate the null space of a matrix.

Comparing the proposed algorithm with previous existing ones showed that the proposed method has a better accuracy. Although the proposed method is analytic and can calculate the parametric problem, however, the necessity of calculating a lot of cofactors and their related signs decreases its popularity especially when the order of the matrix is more than two which extremely increases the required time and calculations. In addition, the resultant null space in this research is not orthogonal. Later, in [23] the application of NOC algorithm was explained for MIMO systems and its implementation for these system was defined. In this research, no idea was proposed for calculating the null space of a matrix and just this null space is employed for their suggested approach. Nie has extended a new method for extracting the null space regarding to solve the system of simultaneous nonlinear equations [24]. This method is a numerical approach and based on iteration process. However, in many problems, the exact solution of the system in a parametric way is required. Dimensionality reduction is an important pre-processing step in many applications. As a result, analytic analysis of null space is studied in [25].

In this research, the conventional approaches such as QR and SVD are employed to calculate the null space of a matrix. These methods are not sufficiently efficient and applicable for many on line and real time industrial applications. In [26] a discrete approach is delivered for solving the null space of constrained systems with holonomic constraints. Afterwards the mentioned method is extended in [27] for systems which have interaction with each other’s. The method is again based on iteration process which needs numerical solution. Although this method is proper for vibrating plants, however, this method is not suitable for the robotic plants in which the exact parametric solution is required. Afterwards in [28] null space application is studied for the constrained and flexible systems.

Two series of constraints are considered in this paper including internal constraints which are related to the solid mode of the system and external constraints which are related to the joint space of the system. Null space is again employed here to eliminate the corresponding Lagrange multipliers. Afterward the applicability of the proposed method is verified for a flexible beam of a robotic arm. Finally, Leyendecker et al. explained in [29] the application of the mentioned method of optimization using variation for different dynamic systems.

As can be seen through the mentioned literature, there is not enough researches about optimal calculation of null-space and specially no study is performed for optimal calculation of modeling the 3PRS robot so far. Thus in this paper the complete kinematic and kinetic modeling of the 3PRS robot considering the constraints of the system and elimination of the constraints using Lagrange multipliers is performed for the first time. The main novelty of the paper is modeling the robot considering the constraints of the system and also proposing a novel method for calculation of the robot null space and eliminating the constraints with a faster method respect to existing ones.

Dynamic formulation of the robot is extracted using Lagrange multipliers and its inverse dynamics is employed as the feedforward signal of the robot control while its forward model is also solved as the plant of the robot. Afterwards by eliminating the Lagrange multipliers and using the state space of the system the formulation of NOC is implemented on the dynamics of the system through which the null space of the robot is calculated using a lower amount of mathematical calculations. The proposed algorithm of null space calculation in this paper is completely analytic and its accuracy is not affected by any numerical method. Moreover, the derived algorithm of NOC in this paper unlike all of the previous algorithms is able to extract the null space matrix with orthogonal bases. Also some simplifications are implemented through the presented algorithm by which the complexity of calculations can be significantly decreased.

This orthogonality results in increasing the accuracy of modeling and decreasing the error of simulation. It is shown that using the proposed method for calculation of the null space, a faster calculation of the null space can be realized which is performable employing any CPU. Also the orthogonal nature of the resultant null space causes the best accuracy that can be achieved. This claim is verified by applying the mentioned methods on new robot of 3PRS. The structure of the paper is as follow: In section two the formulation of NOC and the proposed method of calculation of the null-space are derived. Afterwards in section three the modeling of the robot including of kinematics and kinetics is represented. Moreover, the Lagrange multipliers are eliminated using the proposed method and the dynamic model of the robot is completed by the aid of its corresponding null space and NOC.

The state space of the robot is extracted then and Computed Torque Method (CTM) is implemented to control the robot in its open loop state. Finally, in section four the correctness of all of the stated modeling and control and the efficiency of the proposed optimization method for calculation of null space is verified on a 3PRS robot for both of analytic and numerical approaches by the aid of some comparative simulation scenarios conducted in MATLAB. It is shown that implementing the proposed methods on the 3PRS robot, the required calculations and time for modeling and simulating the robot is significantly decreased while its accuracy is improved.

2 OPTIMUM CALCULATION OF NULL SPACE

The first objective of optimization of null space calculation is minimization of required mathematical operation and required time of calculation. So an orthogonal null space will be achieved. According to mathematical representation of the null space of a matrix, the following set of simultaneous equations should be solved for *X:*

$$A_{m\times n}\vec{X}_{n\times 1}=0 \tag{1}$$

Where *X* is the states of the system and *A* is its relative gains which is generally a non-square matrix. Thus it can be concluded that:

$$AX=0 \;\rightarrow\; \dot{A}X+A\dot{X}=0 \tag{2}$$

If the rank of matrix *A* would be m and we have $(m<n)$ it can be concluded that the number of independent states of *X* is equal to (*n-m*) and the other *m* states are dependent. Each dependent state can be represented as a function of other independent states. Suppose that the independent states are placed at the end rows of the *X* vector and is presented by *v*:

$$\begin{cases}\left(\dfrac{\partial A}{\partial X}\dfrac{\partial X}{\partial v}\dot{v}\right)X+A\left(\dfrac{\partial X}{\partial v}\dot{v}\right)=0\\ \vec{v}=\left[x_{m+1}\;\cdots\;x_{n}\right]^{T}_{1\times(n-m)}\end{cases} \tag{3}$$

If $\frac{\partial A}{\partial X}=0$, then:

$$A\left(\frac{\partial\vec{X}}{\partial\vec{v}}\dot{\vec{v}}\right)=0 \;\rightarrow\; \begin{cases}AN\dot{\vec{v}}=0\\ N=\dfrac{\partial\vec{X}}{\partial\vec{v}}\end{cases} \tag{4}$$

In a homogenous equation like equation (4), considering the fact that the states of vector *v* are independent, it can be concluded that the gain matrix *AN* should be equal to zero:

$$AN=0 \tag{5}$$

Where *N* is the null space of the system. In a holonomic case like the considered robot in this paper, $\frac{\partial A}{\partial X}\neq 0$. Now we have:

$$f(\vec{X}) = 0 \tag{6}$$

Where *f* is the constraint formulation of the considered robot. Thus, it can be concluded that:

$$A\dot{\vec{X}} = 0 \quad , \quad A = \frac{\partial f}{\partial \vec{X}} \tag{7}$$

Then we have:

$$A\left(\frac{\partial \vec{X}}{\partial \vec{v}}\dot{\vec{v}}\right) = 0 \quad \rightarrow \quad \begin{cases} AN = 0 \\ N = \dfrac{\partial \vec{X}}{\partial \vec{v}} \end{cases} \tag{8}$$

Where *N* is the null space of the system. It is obvious that this null space matrix is independent of $\dot{v}$ however it is not independent of *v* necessarily. Thus, the mentioned formulation does not have any contradiction with the stated conclusion. In order to extract matrix *N* using equation (8) we have:

$$\begin{gathered} N_{ij} = \frac{\partial \vec{X}_i}{\partial \vec{v}_j} = \frac{\partial X_{i=1,2,\ldots,m}}{\partial X_{j=m+1,m+2,\ldots,n}} = X_{i,j} \\ N = \left[\begin{array}{ccc} X_{1,(m+1)} & \cdots & X_{1,n} \\ \vdots & \ddots & \vdots \\ X_{m,(m+1)} & \cdots & X_{m,n} \\ \hline & I_{(n-m)\times(n-m)} & \end{array}\right]_{n\times(n-m)} \end{gathered} \tag{9}$$

The elements of matrix *A* can be divided into following sub matrices:

$$\begin{gathered} A = \left[\begin{array}{ccc:ccc} A_{11} & \cdots & A_{1m} & A_{1(m+1)} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ A_{m1} & \cdots & A_{mm} & A_{m(m+1)} & \cdots & A_{mn} \end{array}\right] \\ = \left[\begin{array}{c:c} B_{m\times m} & C_{m\times(n-m)} \end{array}\right] \end{gathered} \tag{10}$$

Each column of matrices *B* and *C* can be written as a vector of size$(m \times 1)$. In order to calculate the elements of matrix *N*, using equation (1) and considering the principal of derivatives of implicit functions, we can write [30]:

$$\begin{gathered} \begin{cases} B = [B_1 \quad B_2 \quad \cdots \quad B_m] \\ C = [C_1 \quad C_2 \quad \cdots \quad C_{n-m}] \end{cases} \\ X_{i,j} = -\frac{|B_1 \quad \cdots \quad B_{i-1} \quad C_j \quad B_{i+1} \quad \cdots \quad B_m|}{|B|} \\ = -\frac{\delta_{ij}}{|B|}. \end{gathered} \tag{11}$$

Where $|\ \ |$ shows determinant of the parameters. Considering the equation (11), it can be seen that the denominator of all of the fractions are equal to $|B|$ and thus the equation (9) can be rewritten as:

$$N = \frac{-1}{|B|}\left[\begin{array}{ccc} \delta_{1(m+1)} & \cdots & \delta_{1(n)} \\ \vdots & \ddots & \vdots \\ \delta_{m(m+1)} & \cdots & \delta_{m(n)} \\ \hline & -|B|\, I_{(n-m)\times(n-m)} & \end{array}\right]_{n\times(n-m)} \tag{12}$$

Where the gain $\frac{-1}{|B|}$ can be ignored without losing the generality of equation (5). Thus, the matrix *N* can be written as below:

$$\begin{cases} \hat{N} = \left[\begin{array}{ccc} \delta_{1(m+1)} & \cdots & \delta_{1(n)} \\ \vdots & \ddots & \vdots \\ \delta_{m(m+1)} & \cdots & \delta_{m(n)} \\ \hline & -|B|\, I_{(n-m)\times(n-m)} & \end{array}\right]_{n\times(n-m)} \\ N = \dfrac{-1}{|B|}\hat{N} \end{cases} \tag{13}$$

It should be noticed that in the case in which the rank of the matrix *A* is not equal to its rows, first, the echelon form (A_e) of the matrix *A* should be calculated and the mentioned process needs to be performed for the echelon form of the matrix *A*.

2.1. Orthogonal Basis of Null Space

For the second stage of null space calculation, the achieved null space matrix will be orthogonal zed in order to achieve the maximum possible accuracy of modeling and simulation process. Each column of N can be written as a vector of size $n \times 1$:

$$N = [N_1 \quad \cdots \quad N_{n-m}] \tag{14}$$

Considering the N_1 as a basis for the orthogonal null space of N, the other bases are orthogonal respect to this base. Generally speaking, it can be said that each two vectors of this space are orthogonal with respect to each other and thus it can be written:

$$N_1^T \vec{X} = 0 \tag{15}$$

And it means that we can add a row to the matrix A without losing the validity of equation (1):

$$\hat{A} = \begin{bmatrix} A \\ N_1^T \end{bmatrix} \tag{16}$$

Now it is possible to rewrite the equation 13 for the $\hat{A}$ and find out a new base for the orthogonal matrix. Repeating this procedure for (*n*-*m*) independent variables, the new matrix $\hat{A}$ will be:

$$\hat{A}_{n\times n} = \begin{bmatrix} A_{m\times n} \\ N^{T} \end{bmatrix} \tag{17}$$

Where N^T is the transpose of the orthogonal null space. This matrix can be written as equation (14) as mentioned. Considering the orthogonality of the calculated null space it can be written:

$$D = N^{T} N = \begin{bmatrix} N_1^T N_1 & 0 & L & 0 \\ 0 & N_2^T N_2 & & M \\ M & & O & 0 \\ 0 & L & 0 & N_{n-m}^T N_{n-m} \end{bmatrix}_{(n-m)\times(n-m)} \tag{18}$$

Please refer to the appendix for more explanations.

3 MODELING OF THE 3PRS ROBOT

Consider a 3PRS parallel robot like "Fig.1" through which 6DOFs of circular end-effector of $B_1B_2B_3$ is controlled using three active prismatic S_1, S_2 and S_3 and three passive angles of $\alpha_1, \alpha_2, \alpha_3$. Point O is the origin of the global coordinate while points A_i are the initial origins of the slides.

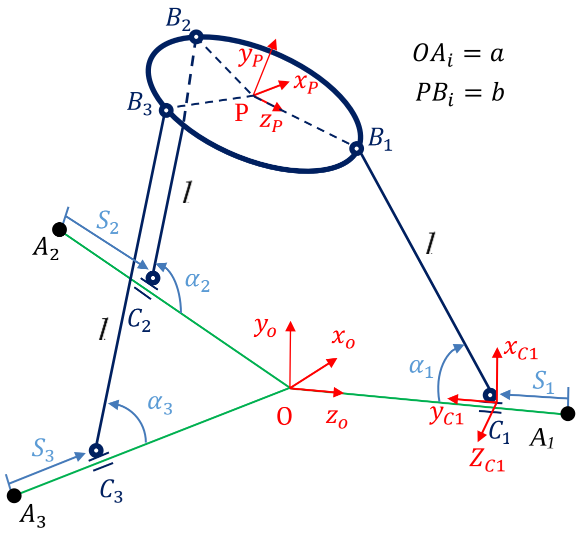

Fig. 1 Schematic view of the parallel mechanism of 3PRS.

3.1. Kinematic Modeling

The coordinates of the key points of the system are:

$$O = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, A_1 = \begin{bmatrix} 0 \\ 0 \\ a \end{bmatrix}, A_2 = \begin{bmatrix} \frac{a\sqrt{3}}{2} \\ 0 \\ \frac{-a}{2} \end{bmatrix}, A_2 = \begin{bmatrix} \frac{-a\sqrt{3}}{2} \\ 0 \\ \frac{-a}{2} \end{bmatrix}. \tag{19}$$

Where a is the length of the prismatic jacks. Using Denavit-Hartenberg (DH) parametrizations, we have the following "Table 1" [31].

Table 1 DH parameters

Link	e	β	d	Θ
OC1	0	-90	$a - s_1$	90

Where e is the distance between the two z axe along x, β is the angle between two z axes around the axis x, d is the perpendicular distance between two origins along the previous z axis and Θ is the angle between two x axes about the previous axis z. The related homogeneous transformation matrix considering the "Table 1" is as follow:

$$H_{c_1}^{o} = R_{z,\Theta} T_{z,d} T_{x,e} R_{x,\beta} \tag{20}$$

Where $R_{z,\Theta}$ is the rotation of Θ around the axis z and $T_{z,d}$ is the translation of d along the axis z. Two other links are the same as the previous one except that they have also rotation around the axis y. Thus, we have:

$$H_{c_1}^{o} = R_{z,\Theta} T_{z,d} T_{x,e} R_{x,\beta} = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & a - s_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_{c_2}^{o} = R_{y,\frac{2\pi}{3}} R_{z,\Theta} T_{z,d} T_{x,e} R_{x,\beta} = \ldots$$

$$\begin{bmatrix} 0 & -\frac{\sqrt{3}}{2} & 0.5 & \frac{\sqrt{3}}{2}(a - s_2) \\ 1 & 0 & 0 & 0 \\ 0 & 0.5 & \frac{\sqrt{3}}{2} & -\frac{1}{2}(a - s_2) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_{c_3}^{o} = R_{y,-\frac{2\pi}{3}} R_{z,\Theta} T_{z,d} T_{x,e} R_{x,\beta} =$$

$$\begin{bmatrix} 0 & \frac{\sqrt{3}}{2} & 0.5 & -\frac{\sqrt{3}}{2}(a - s_3) \\ 1 & 0 & 0 & 0 \\ 0 & 0.5 & -\frac{\sqrt{3}}{2} & -\frac{1}{2}(a - s_3) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}$$

Here C and S are abbreviation of cos and sin respectively. S_i is the displacement of the prismatic joint C_i. The coordinate of B_i with respect to local coordinate attached to the joint C_i is:

$$B_i^{c_i} = \begin{bmatrix} l \sin\alpha_i \\ l \cos\alpha_i \\ 0 \end{bmatrix} \tag{22}$$

Where l is the length of each link and α_i is the angle of link attached to the joint C_i. The coordinate of B_i with respect to global coordinate in O is:

$$\begin{bmatrix} B_i^o \\ 1 \end{bmatrix}_{(S_1,S_2,S_3,\alpha_1,\alpha_2,\alpha_3)} = H_{c_i}^o \begin{bmatrix} B_i^{c_i} \\ 1 \end{bmatrix}. \quad i = 1,2,3 \tag{23}$$

Rotation of the moving plate is considered here respect to the global coordinate attached to the point O and is represented based on Roll, Pitch and Yaw:

$$R_{\varphi,\theta,\psi} = R_{z,\varphi} R_{y,\theta} R_{x,\psi} = \begin{bmatrix} c\varphi c\theta & -s\varphi c\psi + c\varphi s\theta s\psi & s\varphi s\psi + c\varphi s\theta c\psi \\ s\varphi c\theta & c\varphi c\psi + s\varphi s\theta s\psi & -c\varphi s\psi + s\varphi s\theta c\psi \\ -s\theta & c\theta s\psi & c\theta c\psi \end{bmatrix} \tag{24}$$

Where $(x, y, z, \psi, \theta, \phi)$ are the workspace DOFs of the platform. Since the point p is known as $p = [x\ y\ z]^T$ we have:

$$\begin{bmatrix} B_1^o & B_2^o & B_3^o \end{bmatrix}_{(x,y,z,\psi,\theta,\varphi)} = p\begin{bmatrix} 1 & 1 & 1 \end{bmatrix} + R_{\varphi,\theta,\psi} \begin{bmatrix} 0 & \frac{b\sqrt{3}}{2} & -\frac{b\sqrt{3}}{2} \\ 0 & 0 & 0 \\ b & -\frac{b}{2} & -\frac{b}{2} \end{bmatrix} \tag{25}$$

Kinematic constraints of the mechanism considering equations (23, 25) are:

$$\begin{bmatrix} f_1 \\ \vdots \\ f_9 \end{bmatrix} = \begin{bmatrix} B_1^o \\ B_2^o \\ B_3^o \end{bmatrix}_{(x,y,z,\psi,\theta,\varphi)} - \begin{bmatrix} B_1^o \\ B_2^o \\ B_3^o \end{bmatrix}_{(S_1,S_2,S_3,\alpha_1,\alpha_2,\alpha_3)} = 0 \tag{26}$$

Here 9 kinematic constraints are existing which are completely independent.

3.2. Dynamics Modeling

Both of inverse and forward dynamics are extracted here. In inverse dynamics, it is desired to calculate the required prismatic force of the jacks in a way to provide a desired path for the end-effector platform. In order to meet this goal, Lagrange method is employed. Since the robot is under constrained, firstly all of twelve parameters of $[\mathrm{x}, \mathrm{y}, \mathrm{z}, \psi, \theta, \phi, \alpha_1, \alpha_2, \alpha_3, S_1, S_2, S_3]^{\mathrm{T}}$ are considered as the generalized coordinates of the robot. The corresponding kinetic energy of the robot can be calculated using the following equation:

$$\begin{cases} T_p = \frac{1}{2}\omega_p{}^T I_p \omega_p + \frac{1}{2} M \mathrm{v}_p{}^T \mathrm{v}_p \\ T_l = \sum_{i=1}^{3} \left(\frac{1}{2}\omega_i{}^T I_{li} \omega_i + \frac{1}{2} m \mathrm{v}_i{}^T \mathrm{v}_i \right) \end{cases}, \quad T = T_p + T_l \tag{27}$$

Where T_p, M are energy and mass of the moving platform while T_l, m are the same parameters of the links. I_p is the moment of inertia of the moving platform and I_{li} is the same parameter for the ith link with respect to global coordinate. Considering "Fig. 2", the moment of inertia of each link is first calculated in local coordinate $(xyz)_{li}$ and is then transferred to global one by the aid of rotation matrix. Similarly, the moment of inertia of the moving platform is first calculated in coordinate $(xyz)_p$ and is then transferred to global coordinate.

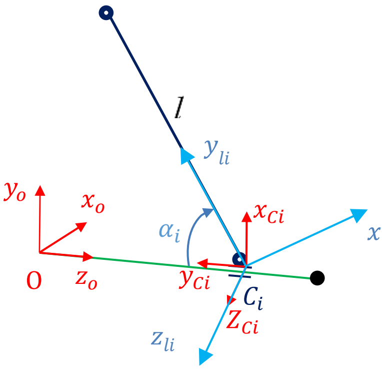

Fig. 2 Local and global coordinates for calculating the moments of inertia.

Considering the above figure, it can be written:

$$R_{li}^o = R_{c_i}^o R_{z,-\alpha_i}, \begin{bmatrix} R_{c_i}^o & d_{c_i}^o \\ 0 & 1 \end{bmatrix} = H_{c_i}^o \tag{28}$$

Where $I_{li},\ I_p$ can be calculated as:

$$I_p = R_{\varphi,\theta,\psi} \begin{bmatrix} \frac{1}{4}Mb^2 & 0 & 0 \\ 0 & \frac{1}{2}Mb^2 & 0 \\ 0 & 0 & \frac{1}{4}Mb^2 \end{bmatrix} R_{\varphi,\theta,\psi}^T, \tag{29}$$

$$I_{li} = R_{li}^o \begin{bmatrix} \frac{1}{3}ml^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3}ml^2 \end{bmatrix} R_{li}^{oT}$$

All of the velocities are considered with respect to the reference coordinates:

$$\begin{cases} \omega_p = \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix}, v_p = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}, \omega_i = R^o_{c_i} \begin{bmatrix} 0 \\ 0 \\ -\dot{\alpha}_i \end{bmatrix}, v_i = \frac{d}{dt} G_i , \\ G_i = H^o_{c_i} \begin{bmatrix} 0.5l \sin \alpha_i & 0.5l \cos \alpha_i & 0 & 1 \end{bmatrix}^T \end{cases} \tag{30}$$

Also the potential energy of the robot can be calculated as:

$$U = Mgy + mg \frac{l}{2} \sum_{i=1}^{3} \sin \alpha_i \tag{31}$$

Where g is the gravitational acceleration of the earth. Using Lagrange multiplier method, we have:

$$\begin{gathered} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} + \sum_{k=1}^{9} \lambda_k \frac{\partial f_k}{\partial q_i} = Q_i , i = 1, \ldots, 12 \\ \begin{cases} L = T - U, \quad Q = \left[[0]_{1\times 9}, F_1, F_2, F_3 \right]^T \\ q = \left[x, y, z, \psi, \theta, \varphi, \alpha_1, \alpha_2, \alpha_3, s_1, s_2, s_3 \right]^T \end{cases} \end{gathered} \tag{32}$$

Where Q is the generalized force of the system and f_k is the constraint relations:

$$\begin{gathered} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) = \frac{\partial}{\partial \dot{q}_j}\left(\frac{\partial L}{\partial \dot{q}_i}\right) \ddot{q}_j + \frac{\partial}{\partial q_j}\left(\frac{\partial L}{\partial \dot{q}_i}\right) \dot{q}_j , \\ \frac{\partial}{\partial \dot{q}_j}\left(\frac{\partial L}{\partial \dot{q}_i}\right) = m_{ji} , \quad \frac{\partial}{\partial q_j}\left(\frac{\partial L}{\partial \dot{q}_i}\right) = c_{ji} , \\ \frac{\partial L}{\partial q_i} = g_i , \quad \frac{\partial f_k}{\partial q_i} = a_{ki} \end{gathered} \tag{33}$$

Using the Lagrange multiplier and conducting the related calculations, the dynamic equation of the robot movement can be extracted as follow:

$$M\ddot{q} + C\dot{q} - G + A^T \lambda = Q \tag{34}$$

Where M is the inertia matrix, C is Coriolis matrix, G is gravity vector, A is the gain matrix of Lagrange multipliers, Q is the generalized force of the system and these parameters are calculated for this robot as equation (35).

The rank of the extracted matrix A here is equal to the number of its rows and this is contributed to the fact that the mentioned kinematic constraints of equation 26 are independent. Thus, it can be concluded that for calculating the null space matrix of A, the echelon of the matrix is not required.

$$\begin{gathered} M = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_{44} & m_{45} & m_{46} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_{54} & m_{55} & m_{56} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_{64} & m_{65} & m_{66} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & m_{77} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{88} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{99} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1 \end{bmatrix}_{12\times 12} , \\ C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{44} & c_{45} & c_{46} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{54} & c_{55} & c_{56} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{64} & c_{65} & c_{66} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c_{77} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{88} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{99} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{12\times 12} , G = \begin{bmatrix} 0 \\ -9.8 \\ 0 \\ g_4 \\ g_5 \\ g_6 \\ g_7 \\ g_8 \\ g_9 \\ 0 \\ 0 \\ 0 \end{bmatrix} , \\ A = \begin{bmatrix} -1 & 0 & 0 & a_{1,4} & a_{1,5} & a_{1,6} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & a_{2,4} & a_{2,5} & a_{2,6} & a_{2,7} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & a_{3,4} & a_{3,5} & 0 & a_{3,7} & 0 & 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & a_{4,4} & a_{4,5} & a_{4,6} & 0 & a_{4,8} & 0 & 0 & -0.86 & 0 \\ 0 & -1 & 0 & a_{5,4} & a_{5,5} & a_{5,6} & 0 & a_{5,8} & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & a_{6,4} & a_{6,5} & 0 & 0 & a_{6,8} & 0 & 0 & 0.5 & 0 \\ -1 & 0 & 0 & a_{7,4} & a_{7,5} & a_{7,6} & 0 & 0 & a_{7,9} & 0 & 0 & -0.86 \\ 0 & -1 & 0 & a_{8,4} & a_{8,5} & a_{8,6} & 0 & 0 & a_{8,9} & 0 & 0 & 0 \\ 0 & 0 & -1 & a_{9,4} & a_{9,5} & 0 & 0 & 0 & a_{9,9} & 0 & 0 & 0.5 \end{bmatrix}_{9\times 12} , \\ \lambda = \begin{bmatrix} \lambda_1 & \lambda_2 & \lambda_3 & \lambda_4 & \lambda_5 & \lambda_6 & \lambda_7 & \lambda_8 & \lambda_9 \end{bmatrix}^T \end{gathered} \tag{35}$$

4 DYNAMIC MODELING CONSIDERING LAGRANGE MULTIPLIERS ELIMINATION

According to the mentioned optimal null space calculation, it is now possible to extract the dynamics of the 3PRS robot considering Lagrange multipliers elimination with a lower amount of calculation. The proposed method in this paper is based on decreasing the engaged dynamic parameters of the system by selecting the proper generalized coordinates. Consider a robot with p moving parts. First, it is required to define the DOFs of each part. If n_i is the nth DOF of part i, the summation of DOF will be:

$$n=\sum_{i=1}^{p} n_i \tag{36}$$

Now if *m* kinematic constraints are involved in, *n-m* degrees of freedom are independent and the rest *m* ones are dependent. Now considering the constraints of equation (26) one can conclude:

$$f_i(x_1,x_2,x_3,\ldots,x_n)=0,\quad i=1,2,\ldots,m. \tag{37}$$

In order to define the dynamic equation of a robot using Lagrange method, it is first required to expand its kinetic and potential energy. These energies should be calculated for each part using its own DOFs respect to the reference coordinate. In this step the DOFs are not summarized as independent generalized coordinate. This is contributed to the fact that calculating the dependent parameters as a function of independent parameters needs heavy mathematical calculations. Especially for the nonlinear robots and specifically the parallel ones with complicated dynamic equations, this importance requires significant consumption of time and effort. Another reason for employing the dependent parameters is related to simplifying the inertia matrix to semi-diagonal one which is vital in direct dynamic simulation through which calculating the inverse of the matrix is required. Thus using the dependent parameters, results in appearing the Lagrange multipliers in the dynamic equation of the system which can be shown as:

$$M\ddot{q}+C\dot{q}-G+A^T\lambda=Q \tag{38}$$

Where *M* is the inertia matrix, *C* is Coriolis matrix, *G* is gravity vector, *Q* is the vector of generalized forces, and λ are the Lagrange multipliers. In order to eliminate the Lagrange multipliers, it is required to extract the null space of matrix *A* using equation (13). Considering the fact that the kinematic constraints are independent, the rank of matrix *A* is equal to its rows. By multiplying the equation (38) by N^T we have:

$$N^TM\ddot{q}+N^TC\dot{q}-N^TG=N^TQ \tag{39}$$

Where $N^TQ=\vec{F}=[F_1\quad F_2\quad F_3]^T$. Now it is possible to simulate the direct dynamics and rewrite the above equation in the form of state space using the following equation:

$$\begin{cases} AN=0 \\ \vec{v}=\begin{bmatrix}\dot{S}_1 & \dot{S}_2 & \dot{S}_3\end{bmatrix}^T \end{cases},\quad \begin{cases} \dot{\vec{q}}=N\vec{v} \\ \ddot{\vec{q}}=\dot{N}\vec{v}+N\dot{\vec{v}} \end{cases} \tag{40}$$

In order to calculate the derivative of the null space, two approaches are possible. The first one is calculating the derivative of the null space using the parametric presentation of the matrix, and the second approach is by employing equation (5) as below:

$$\begin{cases} AN=0 \;\rightarrow\; \dot{A}N+A\dot{N}=0 \;\rightarrow\; \dot{N}=-A^{-1}\dot{A}N \\ A^{-1}=A^T(AA^T)^{-1} \end{cases} \tag{41}$$

Thus for extracting the inverse dynamics which can be used as the feedforward controlling term of the robot, it is sufficient to calculate the desired s_i from the desired path of the moving platform using the explained kinematics and by the aid of constraint relations of equation (26). Also the generalized coordinates and their related derivatives can be calculated and substituted in equations (40, 41). By substituting the equation (40) in equation (39) we have:

$$N^TM(\dot{N}\vec{v}+N\dot{\vec{v}})+N^TCN\vec{v}-N^TG=\vec{F} \tag{42}$$

While equation (42) can be rewritten as follow:

$$\dot{\vec{v}}=(N^TMN)^{-1}N^T(Q+G-CN\vec{v}-M\dot{N}\vec{v}) \tag{43}$$

And therefore by selecting $\vec{X}_{6\times1}=[S_1,S_2,S_3,v^T]^{\mathrm{T}}$ as the states of the system, its corresponding state space can be developed as:

$$\begin{cases} \dot{X}=\begin{bmatrix}\vec{v}\\ \dot{\vec{v}}\end{bmatrix}=\begin{bmatrix}\vec{v}\\ h\end{bmatrix}+\begin{bmatrix}0\\ (N^TMN)^{-1}\end{bmatrix}\vec{F} \\ h=(N^TMN)^{-1}N^T(G-CN\vec{v}-M\dot{N}\vec{v}) \end{cases} \tag{44}$$

Considering equation (13) we have:

$$N=\frac{-1}{|B|}\hat{N}\quad\rightarrow \\ (N^TMN)^{-1}=|B|^2(\hat{N}^TM\hat{N})^{-1} \tag{45}$$

By substituting the equation (45) in (44) and using the $\dot{N}$ of equation (41):

$$\begin{cases} \dot{X}=\begin{bmatrix}\vec{v}\\ \dot{\vec{v}}\end{bmatrix}=\begin{bmatrix}\vec{v}\\ h\end{bmatrix}+\begin{bmatrix}0\\ |B|^2(\hat{N}^TM\hat{N})^{-1}\end{bmatrix}\vec{F} \\ h=(\hat{N}^TM\hat{N})^{-1}\hat{N}^T(-|B|G-(C-MA^T(AA^T)^{-1}\dot{A})\hat{N}\vec{v}) \end{cases} \tag{46}$$

It can be seen that the term h in equation (46) is significantly summarized respect to the same term in equation (44). In fact, splitting the |*B*| from null space matrix has many advantages toward simplifying the dynamic equation and its related mathematical simulation. In other researches, the existence of this parameter |*B*| is ignored while the speed of calculations is also low. Decreasing the usage of the term$|B|$ in *h* can heavily decrease the complexity of the parametric calculations which finally increases the efficiency of the required consumption of calculation time.

5 SIMULATION STUDY

In order to check the developed kinematics and kinetics and also verify the efficiency of the proposed mathematical simplification of modeling using the explained NOC method, some simulation scenarios are provided and the results are compared and analyzed. Following parameters are employed for the simulating the mentioned robot. ("Table 2")

Table 2 Engaged parameters for simulating the 3PRS robot

Parameter	symbol	Unit	value
Jacks' course	a	(m)	0.8
Radius of the moving platform	b	(m)	0.2
Length of the links	l	(m)	0.5
Mass of the moving platform	M	(kg)	1
Mass of the links	m	(kg)	0.1
Gravitational acceleration	g	$(m/s2)$	9.8

5.1. Model Verification

In order to check the extracted model of the system, the robot is simulated in SimMechanics and its results are compared with the results of MATLAB. The scheme of the modeled robot and its simulation process in SimMechanics is shown in "Figs. 3, 4".

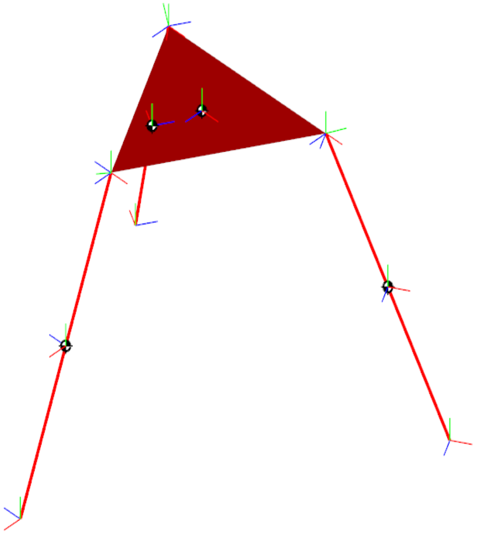

Fig. 3 The scheme of the modeled robot in SimMechanics.

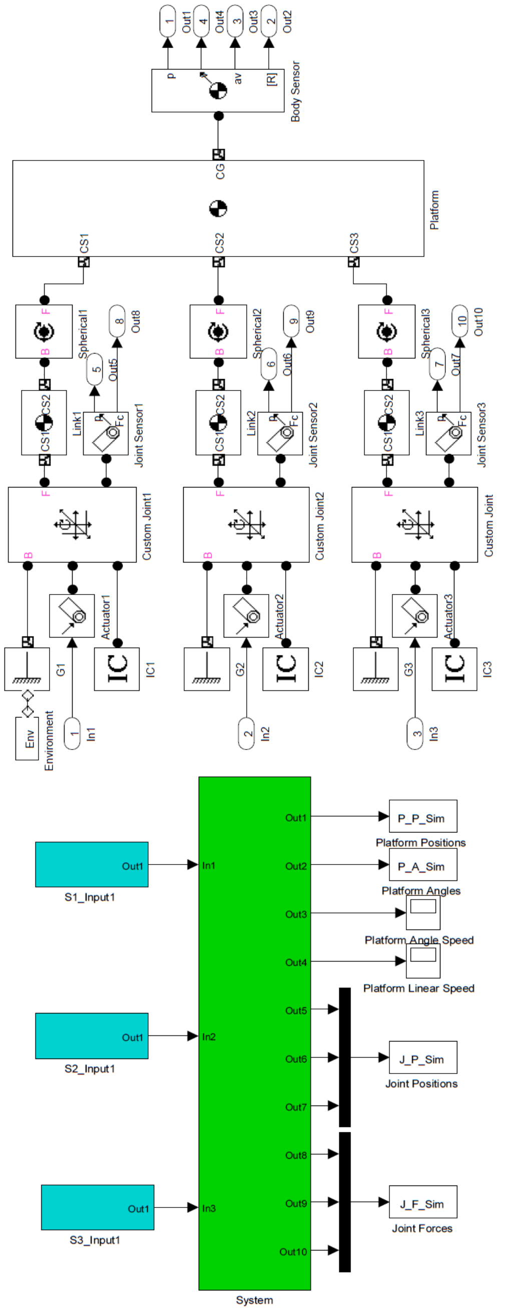

Fig. 4 The scheme of the simulation in SimMechanics.

Consider the following path as the desired trajectory of the sliders S_i which results in the desired platform trajectory of X:

$$\begin{cases} S_1 = 0.6+0.2\sin(t), \\ S_2 = 0.6-0.3\sin(t), \\ S_3 = 0.6-0.2\sin(t). \end{cases} \tag{47}$$

The above desired joint space trajectory is employed for verification process and the actual movement of the workspace platform and also the required force of the jacks are compared between SimMechanics and MATLAB. Comparison of the generalized coordinates and generalized force between MATLAB (C) and SimMechanics (S) is shown in “Fig. 5”.

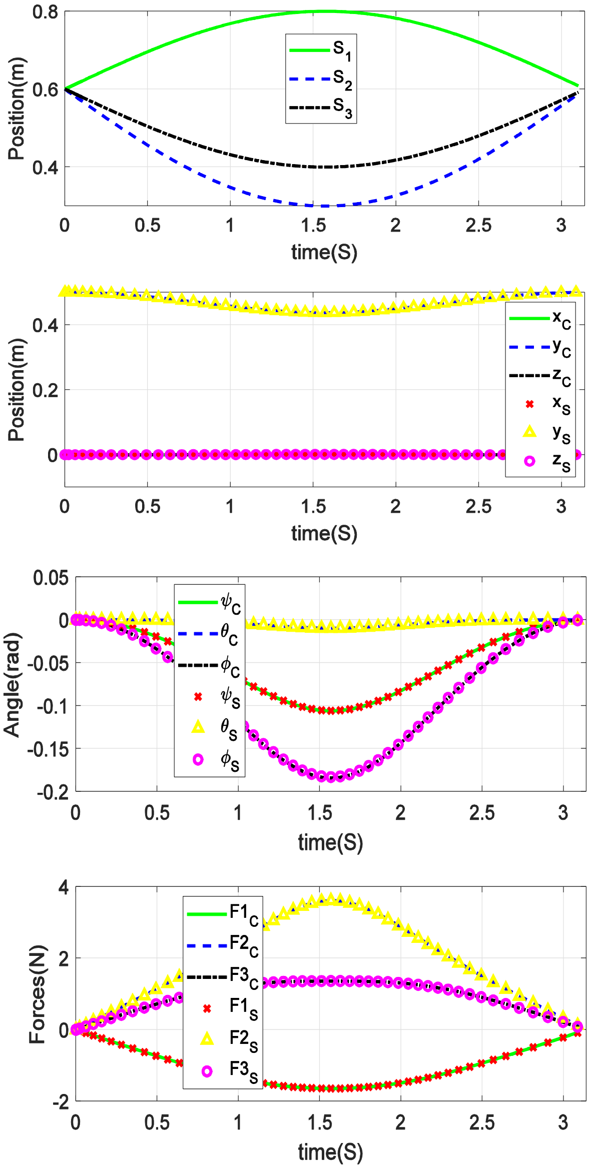

Fig. 5 Comparison of the generalized coordinates and generalized force between MATLAB (C) and SimMechanics (S).

It can be seen that an acceptable compatibility exists between the profiles of MATLAB and SimMechanics which proves the validity of robot modeling. Here in order to calculate the dynamic force of the mechanism, all of the generalized coordinates are extracted using inputs of $[s_i, \dot{s}_i, \ddot{s}_i]$ and considering the constraints of equation (26). Then using the above results and employing equation (42) the hydraulic force of the jacks are calculated. In order to verify the correctness of modeling, the actual kinematic results of the end-effector movement which is gained by the aid of solving the forward dynamic and using the proposed method of NOC is extracted and is compared with the desired trajectory as shown in “Fig. 6”.

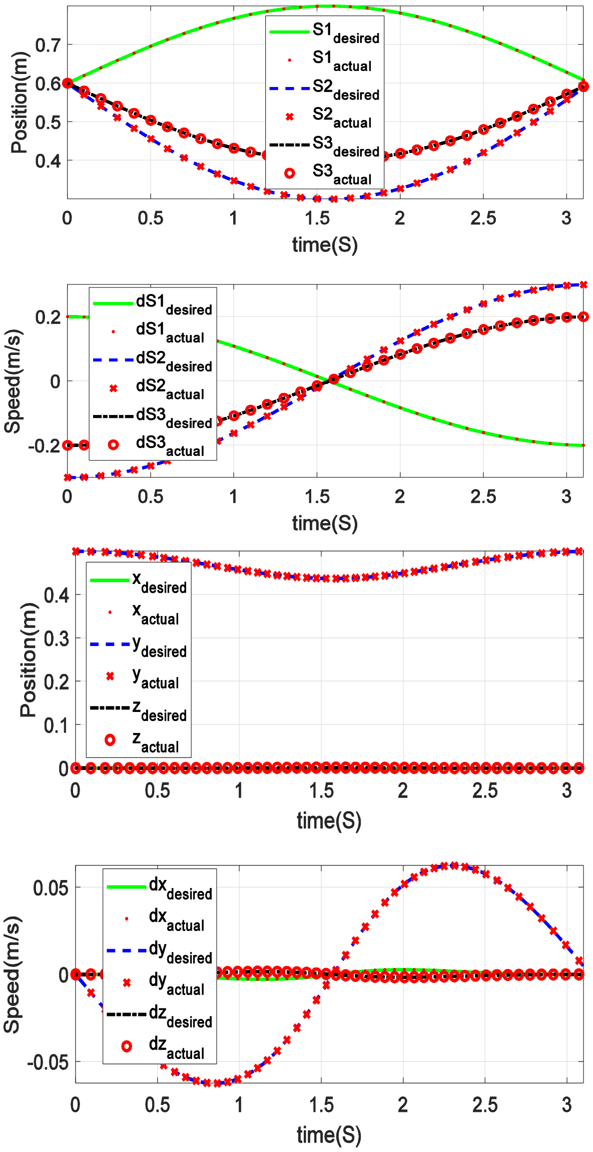

Fig. 6 Comparison of the actual and desired kinematic movement of the end-effector and its speed.

It can be seen that the actual movements of the end-effector which is calculated using the proposed NOC of this paper have a good compatibility with the desired trajectory of the end-effector with an error of less than 10^{-3} which is related to numerical error of the employed software. This compatibility shows the correctness and validity of the proposed method. Also the rotation of the platform and its speed is compared here between the inverse and direct dynamics as shown in “Fig.7”.

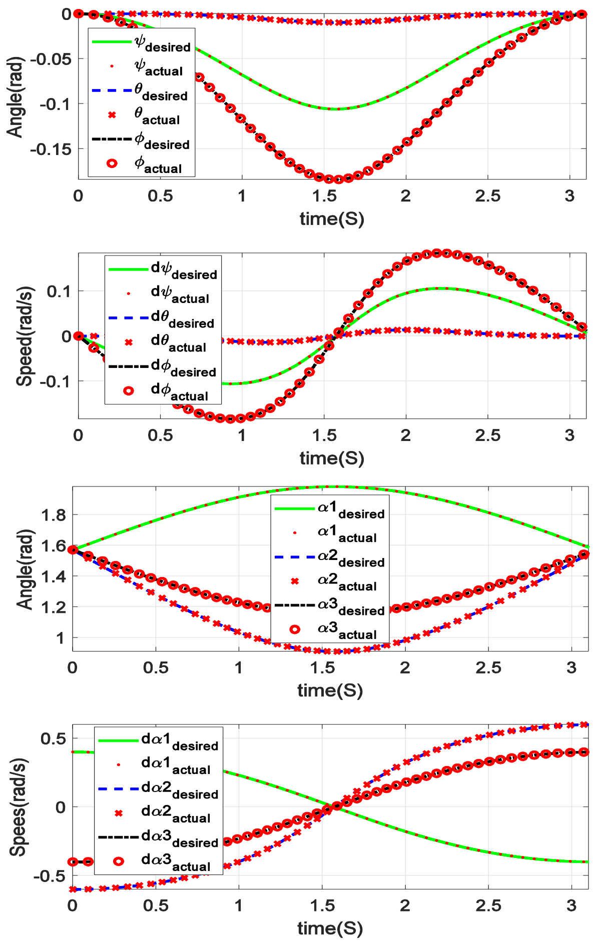

Fig. 7 Comparison of the actual and desired angular movement of the end-effector and its speed.

Again here the good compatibility between the inverse and direct dynamics shows the correctness of modeling and null space calculation. The little deviation in the profile of rotational speed is due to closing to singularity region of Jacobian matrix which can be compensated by a more efficient setting of the calculation step size. In order to verify the satisfaction of the kinematic constraints, the time profile of equation (26) which is the error of kinematic constraint satisfaction is extracted in "Fig. 8".

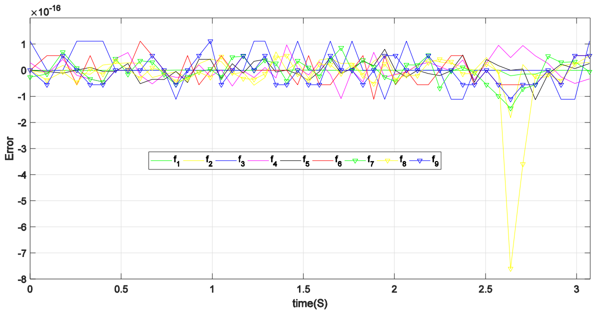

Fig. 8 The error of satisfaction of the kinematic constraints.

For the perfect model in which the kinematic constraints are perfectly satisfied, this profile should be zero during the simulation. As can be seen this condition is satisfied with a great accuracy of order 10^{-16} which shows that not only path tracking is conducted accurately but also the related kinematic constraints are completely satisfied simultaneously.

5.2. Proposed NOC Algorithm Verification

In order to verify the efficiency of the proposed method of elimination of Lagrange multipliers compared to conventional ones, the required calculations toward modeling of the mentioned 3PRS robot is evaluated using the proposed method and is compared with most conventional and traditional method of Gauss-Jordan elimination. The simulation is performed in MALAB. To compare these two methods, modeling of forward dynamics is considered in which elimination of Lagrange multipliers is required. Thus, the null space of matrix A needs to be calculated using the mentioned methods. This null space is calculated considering two approaches of numerical and analytical calculation methods. Required time of calculation for analytic approach is compared between the proposed method and traditional one using tic-toc code and its results are presented in "Table 3".

Table 3 Required time for calculating the null space matrix and its comparison between two methods using analytic method

Method	Time (sec)
Method of Gauss-Jordan elimination	298.14
Proposed method of NOC	2.16

It can be seen that there is a huge difference between the required mathematical calculation toward performing the elimination of multipliers between the conventional method of Gauss-Jordan elimination and the proposed method of NOC. Here the required time is decreased by about 138 times which is a significant optimization in required processing of modeling. Also required time of processing the mentioned calculations using numerical approach is compared using tic-toc code for a same CPU and the result is as below. This comparison is performed for 1000 times of calculation of null space. ("Table 4")

Table 4 Required time for calculating the null space matrix and its comparison between two methods using numerical method

Method	Time (sec)
Method of Gauss-Jordan elimination	0.7526
Proposed method of NOC	0.2001

It can be observed that again here required time of calculating the null space matrix is also decreased to some extent by about 3.7 times. As it was expected that this optimization for analytic method is more sever since here the mathematical processing is parametric. Thus for applications in which analytic and parametric modeling is required such as optimization and path planning processes, the proposed method can enormously decrease the time and increases the speed of processing. These two comparisons show the efficiency and superiority of the proposed method of modeling over the conventional algorithms since the heavy mathematical calculation of the constrained nonlinear modeling of the system is considerably reduced. From equation 46 it can be inferred that the required calculations for modeling the system is directly dependent to the required calculations of extracting its related null space matrix and this reduction in calculation of null space matrix can extremely save the time and effort of modeling calculations.

6 CONCLUSION

In this paper a new version of parallel robot was studied which has an acceptable workspace and a good capability for load transferring. Also a new method for optimum calculation of the null space of the system was proposed through which the modeling of the system can be performed with the minimum mathematical calculation and the maximum accuracy. In order to verify the efficiency of the algorithm, it was implemented on the mentioned 3PRS robot.

Complete kinematic modeling of the robot was represented which is perquisite for dynamics, control and optimization of the robot. Also implementing Lagrange multiplier, the dynamic equation of the constrained system was extracted. It was explained that the inverse dynamic of the system can be employed as the feedforward controlling term of the system while the forward kinetic of the robot can be used as the plant of the system in order to verify the designed controller and the proposed optimization of the null space calculation.

Afterwards, a new method of calculating the null space of a system was presented through which a lower amount of mathematical calculation and time is required. Also the accuracy of the proposed formulation can be maximized by orthogonal zing the calculated null space. It was seen that to implement the proposed optimization on the constrained case study of this paper, a new technique can be employed through which the dependent generalized coordinates of the robot and related multipliers were eliminated using the null space of the system. All of the mentioned modeling, controlling and optimization processes were verified by conducting some analytic and comparative simulation scenarios in MATLAB for a 3PRS robot.

It was seen that the designed kinematic and kinetic model of the system is correct since the actual and desired path of forward and inverse models are greatly compatible with 99.7 percent accuracy. The mentioned comparison in dynamic modeling also showed the efficiency of the designed controller based on computed torque method. Afterwards the proposed methods of optimization of calculating the null space was verified by comparing the required mathematical calculation and time consumption for the proposed method and the conventional method of Gauss-Jordan elimination in MATLAB.

The correctness of modeling and the satisfaction of kinematic constraints were verified by the aid of SimMechanics and some analytic simulation scenarios. The rate of optimization by the aid of proposed method was also checked through two main approaches of analytic and numerical calculations. It was seen that the required time of processing for calculating the related null space is significantly decreased by about 138 times for analytic approach and about 3.7 times for numerical one. It was explained that this difference is contributed to the fact that in analytic approach the calculation should be performed in a parametric way.

Moreover, it was shown that by orthogonal zing the null space in this method the accuracy of the modeling can be increased simultaneously. Therefore, it was seen that using the results of this paper, the null space of 3PRS robot can be easily extracted and its related model can be developed while the minimum calculation and time is required and the maximum accuracy can be achieved.

7 APPENDIX

Here an example is presented for better comprehension of the efficiency of the proposed optimization method of this paper. Suppose the following matrices:

$$A = \begin{bmatrix} 1 & 2 & 0 & 1 \\ 5 & 8 & 1 & 3 \end{bmatrix} \rightarrow$$

$$C = \begin{bmatrix} 0 & 1 \\ 1 & 3 \end{bmatrix}, B = \begin{bmatrix} 1 & 2 \\ 5 & 8 \end{bmatrix}, |B| = -2$$

$$N = \begin{bmatrix} -\frac{\begin{vmatrix} 0 & 2 \\ 1 & 8 \end{vmatrix}}{-2} & -\frac{\begin{vmatrix} 1 & 0 \\ 5 & 1 \end{vmatrix}}{-2} & 1 & 0 \\ -\frac{\begin{vmatrix} 1 & 2 \\ 3 & 8 \end{vmatrix}}{-2} & -\frac{\begin{vmatrix} 1 & 1 \\ 5 & 3 \end{vmatrix}}{-2} & 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 0.5 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}^T \qquad \text{A-1}$$

As can be seen, the resultant of (AN) is equal to zero, and the matrix N is the expected null space matrix. Now if we would like to calculate the N in a way that its base would be orthogonal, we can implement the mentioned algorithm of this paper as follow:

$$\begin{cases} A=\begin{bmatrix}1&2&0&1\\5&8&1&3\end{bmatrix} \\ N_1=\begin{bmatrix}-1&0.5&1&0\end{bmatrix}^T\end{cases} \rightarrow \hat{A}=\begin{bmatrix}1&2&0&1\\5&8&1&3\\-1&0.5&1&0\end{bmatrix}$$

$$C=\begin{bmatrix}1\\3\\0\end{bmatrix}, B=\begin{bmatrix}1&2&0\\5&8&1\\-1&0.5&1\end{bmatrix}, \left|B\right|=-4.5 \qquad \text{A-2}$$

$$\rightarrow N_2=\begin{bmatrix}0.33\\-0.66\\0.66\\1\end{bmatrix}$$

$$N=[N_1 \quad N_2]=\begin{bmatrix}-1&0.5&1&0\\0.33&-0.66&0.66&1\end{bmatrix}^T$$

$$\rightarrow N^T N=\begin{bmatrix}2.25&0\\0&2\end{bmatrix}$$

As can be seen the resultant of (AN) is equal to zero. Also in $(N^T N)$, all of the elements are zero except than the main diagonal of the matrix.

REFERENCES

[1] Stewart, D., A Platform with Six Degrees of Freedom, Proceedings of the Institution of Mechanical Engineers, Vol. 180, No. 1, 1965, pp. 371-386.

[2] Ruiz, F., Campa, C., Roldán-Paraponiaris, and Altuzarra, O., Dynamic Model of a Compliant 3PRS Parallel Mechanism for Micromilling, in Microactuators and Micromechanisms: Springer, 2017, pp. 153-164.

[3] Li, Y., Xu, Q., Kinematics and Inverse Dynamics Analysis for a General 3-PRS Spatial Parallel Mechanism, Robotica, Vol. 23, No. 02, 2005, pp. 219-229.

[4] Li, Y., Staicu, S., Inverse Dynamics of a 3-PRC Parallel Kinematic Machine, Nonlinear Dynamics, Vol. 67, No. 2, 2012, pp. 1031-1041.

[5] Altuzarra, O., Gomez, F. C., Roldan-Paraponiaris, C., and Pinto, C., Dynamic Simulation of a Tripod Based in Boltzmann-Hamel Equations, in Proceedings of ASME International Design Engineering Technical Conferences, Vol. 5, 2015, pp. 8-22.

[6] Nikravesh, P., Haug, E., Generalized Coordinate Partitioning for Analysis of Mechanical Systems with Nonholonomic Constraints, ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 105, 1983, pp. 379-384.

[7] Singh, R., Likins, P., Singular Value Decomposition for Constrained Dynamical Systems, Journal of Applied Mechanics, Vol. 52, No. 4, 1985, pp. 943-948.

[8] Kim, S., Vanderploeg, M., A General and Efficient Method for Dynamic Analysis of Mechanical Systems using Velocity Transformations, Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 108, No. 2, 1986, pp. 176-182.

[9] Liang, Lance, G. M., A Differentiable Null Space Method for Constrained Dynamic Analysis, Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 109, No. 3, 1987, pp. 405-411.

[10] Angeles, J., Lee, S. K., The Formulation of Dynamical Equations of Holonomic Mechanical Systems using a Natural Orthogonal Complement, Journal of Applied Mechanics, Vol. 55, 1988, pp. 243-249.

[11] Angeles, J., Ma, O., Dynamic Simulation of N-Axis Serial Robotic Manipulators using a Natural Orthogonal Complement, The International Journal of Robotics Research, Vol. 7, No. 5, 1988, pp. 32-47.

[12] Saha, S. K., Angeles, J., Dynamics of No Holonomic Mechanical Systems using a Natural Orthogonal Complement, Journal of Applied Mechanics, Vol. 58, No. 1, 1991, pp. 238-243.

[13] Terze, Z., Lefeber, D., and Muftić, O., Null Space Integration Method for Constrained Multibody Systems with no Constraint Violation, Multibody System Dynamics, Vol. 6, No. 3, 2001, pp. 229-243.

[14] Pendar, H., Vakil, M., and Zohoor, H., Efficient Dynamic Equations of 3-RPS Parallel Mechanism through Lagrange Method, in Proceedings of the Robotics, Automation and Mechatronics, IEEE Conference, Vol. 2, 2004, pp. 1152-1157.

[15] Rao, K., Saha, S., and Rao, P., Dynamics Modelling of Hexaslides using the Decoupled Natural Orthogonal Complement Matrices, Multibody System Dynamics, Vol. 15, No. 2, 2006, pp. 159-180.

[16] Phong, V., Hoang, N. Q., Singularity-Free Simulation of Closed Loop Multibody Systems by using Null Space of Jacobian Matrix, Multibody System Dynamics, Vol. 27, No. 4, 2012, pp. 487-503.

[17] Marino, Parker, L. E., Antonelli, G., and Caccavale, F., A Decentralized Architecture for Multi-Robot Systems Based on the Null-Space-Behavioral Control with Application to Multi-Robot Border Patrolling, Journal of Intelligent & Robotic Systems, Vol. 71, No. 3-4, 2013, pp. 423-444.

[18] Raoofian, Kamali, A., and Taghvaeipour, A., Forward Dynamic Analysis of Parallel Robots using Modified Decoupled Natural Orthogonal Complement Method, Mechanism and Machine Theory, Vol. 115, 2017, pp. 197-217.

[19] Coleman, T. F., Sorensen, D. C., A Note on the Computation of an Orthonormal Basis for the Null Space of a Matrix, Mathematical Programming, Vol. 29, No. 2, 1984, pp. 234-242.

[20] Berry, M., Heath, M., Kaneko, I., Lawo, M., Plemmons, R., and Ward, R., An Algorithm to Compute a Sparse Basis of the Null Space, Numerische Mathematik, Vol. 47, No. 4, 1985, pp. 483-504.

[21] Coleman, T. F., Pothen, A., The Null Space Problem I. Complexity, SIAM Journal on Algebraic Discrete Methods, Vol. 7, No. 4, 1986, pp. 527-537.

[22] Dai, J. S., Jones, J. R., Null–Space Construction using Cofactors from a Screw–Algebra Context, in

Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, Vol. 458, No. 2024, 2002, pp. 1845-1866.

[23] Bourdoux, Khaled, N., Joint TX-RX Optimisation for MIMO-SDMA Based on a Null-Space Constraint, Proceedings of the Vehicular Technology Conference, Vol. 1, 2002, pp. 171-174.

[24] Nie, P. Y., A Null Space Method for Solving System of Equations, Applied Mathematics and Computation, Vol. 149, No. 1, 2004, pp. 215-226.

[25] Ye, J., Xiong, T., Computational and Theoretical Analysis of Null Space and Orthogonal Linear Discriminant Analysis, Journal of Machine Learning Research, Vol. 7, No. Jul, 2006, pp. 1183-1204.

[26] Betsch, P., The Discrete Null Space Method for the Energy Consistent Integration of Constrained Mechanical Systems: Part I: Holonomic Constraints, Computer Methods in Applied Mechanics and Engineering, Vol. 194, No. 50, 2005, pp. 5159-5190.

[27] Betsch, P., Leyendecker, S., The Discrete Null Space Method for the Energy Consistent Integration of Constrained Mechanical Systems, Part II: Multibody Dynamics, International Journal for Numerical Methods in Engineering, Vol. 67, No. 4, 2006, pp. 499-552.

[28] Leyendecker, S., Betsch, P., and Steinmann, P., The Discrete Null Space Method for the Energy-Consistent Integration of Constrained Mechanical Systems. Part III: Flexible Multibody Dynamics, Multibody System Dynamics, Vol. 19, No. 1, 2008, pp. 45-72.

[29] Leyendecker, S., Ober-Blöbaum, S., Marsden, J. E., and Ortiz, M., Discrete Mechanics and Optimal Control for Constrained Systems, Optimal Control Applications and Methods, Vol. 31, No. 6, 2010, pp. 505-528.

[30] Spivak, M., Calculus on Manifolds: a Modern Approach to Classical Theorems of Advanced Calculus. CRC Press, 2018.

[31] Spong, M. W., Hutchinson, S., and Vidyasagar, M., Robot Modeling and Control, Jon Wiley & Sons, 2005, ISBN-100-471-649.