



HAL
open science

A new solution for machining with RA-PKMs: Modelling, control and experiments

Jonatan Martín Escorcia-Hernández, Ahmed Chemori, Hipólito
Aguilar-Sierra, Jesús Arturo Monroy-Anieva

► **To cite this version:**

Jonatan Martín Escorcia-Hernández, Ahmed Chemori, Hipólito Aguilar-Sierra, Jesús Arturo Monroy-Anieva. A new solution for machining with RA-PKMs: Modelling, control and experiments. *Mechanism and Machine Theory*, 2020, 150, pp.#103864. 10.1016/j.mechmachtheory.2020.103864 . lirmm-02514747

HAL Id: lirmm-02514747

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-02514747v1>

Submitted on 22 Mar 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A New Solution for Machining with RA-PKMs: Modelling, Control and Experiments

Jonatan Martín Escorcía-Hernández^a, Ahmed Chemori^b, Hipólito Aguilar-Sierra^{c,*}, Jesús Arturo Monroy-Anieva^d

^aUniversidad Politécnica de Tulancingo, Calle Ingenierías No. 100 C.P. 43629 Tulancingo Hidalgo México

^bLIRMM, Université de Montpellier - CNRS, 161 rue Ada, 34090 Montpellier France

^cCuerpo Académico de Control e Instrumentación. División de Ingenierías, Universidad Politécnica de Tulancingo, Calle Ingenierías No. 100 C.P. 43629 Tulancingo Hidalgo México

^dCONACYT-Centro de Investigación en Ciencias de la Información Geoespacial. Parque Científico Tecnológico de Yucatán, Biblioteca Central, Tercer piso. Carretera Sierra Papacal - Chuburná Pto. Km 5 C.P. 97302. Mérida, Yucatán, México

Abstract

In this paper, a novel 5-Degree of Freedom (DOF) Redundantly Actuated (RA) Parallel Kinematic Manipulator (PKM) called SPIDER4 is presented. The main purpose of this manipulator is to perform machining tasks such as drilling and milling. All the mathematical models including the forward and inverse kinematic models, as well as the inverse dynamic model were developed. Owing to machining tasks require high precision, a RISE Feedforward controller is proposed for desired trajectory tracking. To show the performance and effectiveness of the proposed control scheme, real-time experiments were performed. The obtained results of the proposed controller compared to the standard RISE controller are presented and discussed. They confirm that the proposed controller outperforms the standard one.

Keywords: Parallel Kinematic Manipulators, Kinematic Models, Dynamic Model, RISE Control, Actuation Redundancy

1. Introduction

Parallel Kinematic Manipulators, also called in the literature Parallel Kinematic Machines or Parallel Robots belong to a specific kind of manipulators formed by a fixed base connected to a mobile platform via several independent kinematic chains. These manipulators have been used in various areas ranging from medical applications to industrial applications. The main industrial applications developed by PKMs include among others flight simulators, food packaging and machining manipulators [1]. Machining is the broad term used to describe the removal of material from a workpiece and is one of the most important manufacturing processes. Machining operations can be applied to metallic and non-metallic materials such as polymers, wood, ceramics, composites and other materials. The most common machining operations are milling, turning and drilling, these operations require a high precision in positioning of the cutting tool as well as in the desired cutting path, hence the machine tools should satisfy these requirements [2]. The inclusion of the Computer Numerical Control (CNC) in machine tools has allowed this industry to obtain high quality products, with a shorter production time and a lower cost, compared to the first manually operated machine tool [3]. Conventional machine tools have mainly serial kinematic architecture

*Corresponding author

Email addresses: 1715002@upt.edu.mx (Jonatan Martín Escorcía-Hernández), ahmed.chemori@lirmm.fr (Ahmed Chemori), hipolito.aguilar@upt.edu.mx (Hipólito Aguilar-Sierra), jmonroy@centrogeo.edu.mx (Jesús Arturo Monroy-Anieva)

with each axial movement supporting the following axis and providing its motion. However a significant drawback of conventional machines lies the moving parts that should be heavy enough to provide the necessary stiffness to control the bending movements. This may impact the dynamic performance and reduces the operational flexibility. PKMs, due to their structure conformed by the closed-loop kinematic chain mechanism, have some key advantages, compared to their serial counterparts in terms of accuracy, stiffness, and moving masses and inertias; additionally they are relatively simpler in mechanical construction compared to the existing conventional machining tools [4] being thus more suitable for machining operations. However, the main drawbacks of PKMs lies in their small workspace and extensive singular configurations [5] [6]. In order to avoid the problem of singularities, the use of actuation redundancy may be investigated [7]. During last decades, several PKMs have been developed for machining tasks, ranging from 2 DOF to 6 DOF. Some notable examples can be mentioned, including: the Variax Hexacenter developed by Giddings & Levis company with 6 DOFs and based on the Gough platform [5], Orthoglide PKM which has 3 traslational DOFs with a fixed orientation [8], ARROW PKM which has 5 DOFs (3T-2R) [9], HexaM of Toyoda with 6 DOFs (3T-3R) [10], P800/P200 from Metrom, Okuma Cosmo Center and The Octahedral Hexapod from Ingersoll with 5 DOFs [5].

PKMs are highly nonlinear systems, then controlling them is a considerable challenge. In the literature several control schemes have been proposed for PKMs, where most of them rely on the Proportional Integral Derivative controller (PID), because of its simplicity in real-time implementation [11]. Nevertheless, if this controller is subjected to operational changes, such as external disturbances, changes in speed and acceleration of desired translation, its performance may significantly decrease [12]. **The Nonlinear PID controller (NLPID) is an improved version of the PID control, with the difference that its feedback gains are time-varying. The time-varying gains produce corrective actions through conditions based-on the system states allowing to this controller reject external disturbances and improve the overall performance of the system [13]. In order to cancel the inherent nonlinearities in PKMs, the dynamic model (totally or partially) should be included in the feedforward control design.** These kinds of control schemes are known as model-based controllers and may offer better performance than non-model-based controllers [14]. We can stand out different control schemes implemented on PKMs, either model-based or non-model-based, including: \mathcal{L}_1 adaptive control [15], Robust Integral Signal Error (RISE) [16], Active Disturbance Rejection Control (ADRC) [17], Adaptive Terminal Sliding Mode control (ATSMC) [12], Dual-Space adaptive control [18], and Desired Compensation Adaptation Law (DCAL) [19], among others.

In this paper a new RA-PKM called SPIDER4 is introduced, designed for machining applications such as milling and drilling. **The modelling of this new PKM is the main purpose of this paper.** In this work the inverse and forward kinematic models of the robot have been developed as well as a simplified inverse dynamic model. A RISE feedforward controller is then proposed for motion control of the cutting spindle. The paper is organized as follows: SPIDER4 concept and its mathematical models are introduced in Section 2. In Section 3 the description proposed RISE feedforward controller is introduced. To validate the performance of SPIDER4 PKM with the proposed controller, real-time experiments were carried out, their results are presented and discussed in Section 4. Finally, conclusions and future work are addressed in Section 5.

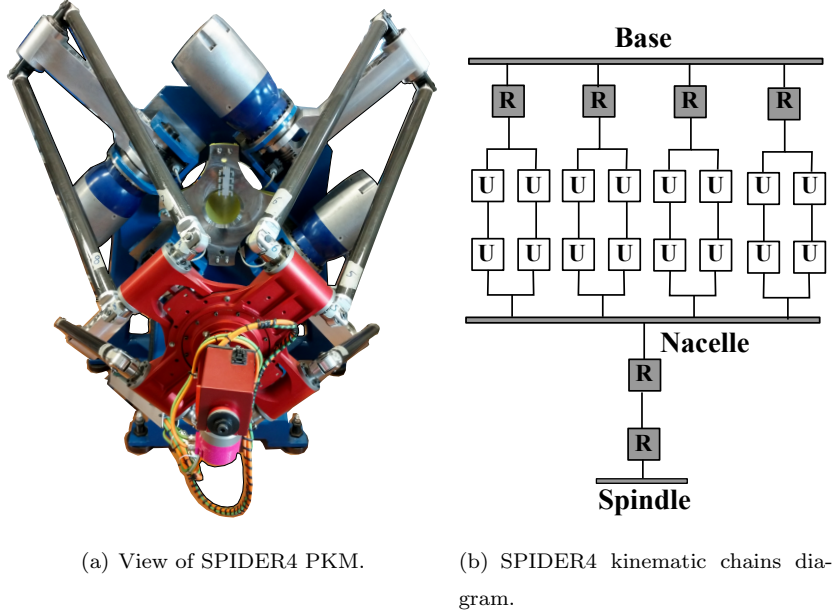


Figure 1: The SPIDER4 RA-PKM and its kinematics

2. SPIDER4 PKM: Description and modelling

SPIDER4 is a RA-PKM designed for CNC machining tasks of resin materials. It has 5 DOF: 3 translations denoted by (x, y, z) and 2 rotations denoted by (ϕ, ψ) . The translational motion is performed by the parallel structure consisting of four kinematic chains, each one formed by one arm and one forearm linked by a passive universal joint. Each arm is connected directly to an electric motor located in the fixed base, and the forearm is connected to the moving platform through two parallel bars, in a PKM Delta-like architecture. The rotational motions are generated by an independent serial wrist mechanism placed on the nacelle which is actuated through two electric motors. The machining spindle is placed at the end of this mechanism. Figure 1(a) shows SPIDER4 RA-PKM, Figure 1(b) illustrates the kinematic configuration diagram of SPIDER4 where, the gray boxes represent the active rotational joints and the white boxes represent the passive universal joints. The description of the main components of SPIDER4 is illustrated in Figure 2.

2.1. Inverse Kinematic Model

For SPIDER4, the inverse Kinematic Model (IKM) consists in finding the actuated joint vector position corresponding to a specific position of the spindle located on the nacelle. **The spindle position is expressed in the fixed reference frame $O-x_o, y_o, z_o$ as:** ${}^o\mathbf{S}_S = [x \ y \ z \ \phi \ \psi]^T \in \mathbb{R}^{5 \times 1}$. **The vector of generalized coordinates is denoted by** $\mathbf{Q} = [q_1 \ q_2 \ q_3 \ q_4 \ \phi \ \psi]^T \in \mathbb{R}^{6 \times 1}$ by considering all active joints located on the fixed base and the nacelle. It is worth mentioning that the variables ϕ, ψ are the same for operational and joint spaces; in addition, we can define a vector which only involves the SPIDER4 parallel structure joint variables $\mathbf{q} = [q_1 \ q_2 \ q_3 \ q_4]^T \in \mathbb{R}^{4 \times 1}$. Examples of modelling methods to compute the IKM of PKMs with delta-like architecture have been developed in [20] [21]. However, to apply any of these methods on SPIDER4, it is first necessary to find the nacelle position expressed by ${}^o\mathbf{N}_N = [x_n \ y_n \ z_n]^T \in \mathbb{R}^{3 \times 1}$ which depends on the spindle position ${}^o\mathbf{S}_S$. To understand the kinematics of SPIDER4 let us consider Figure 3.

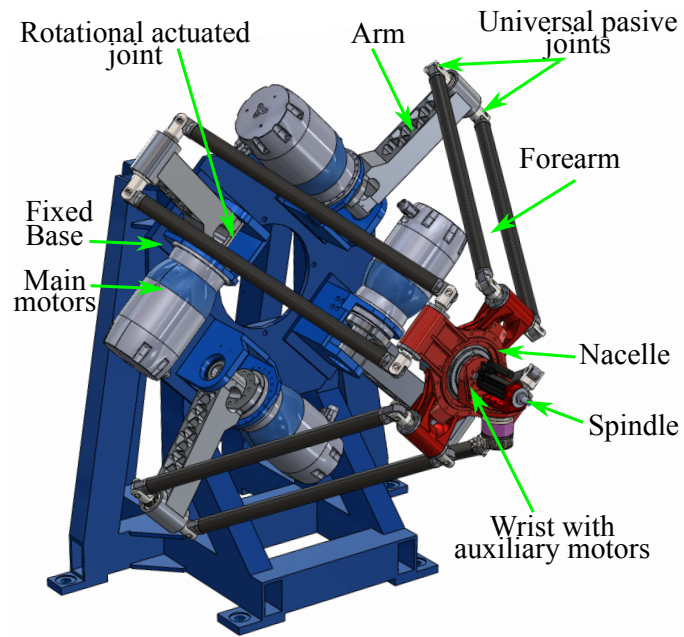


Figure 2: CAD View of SPIDER4 with its main components description

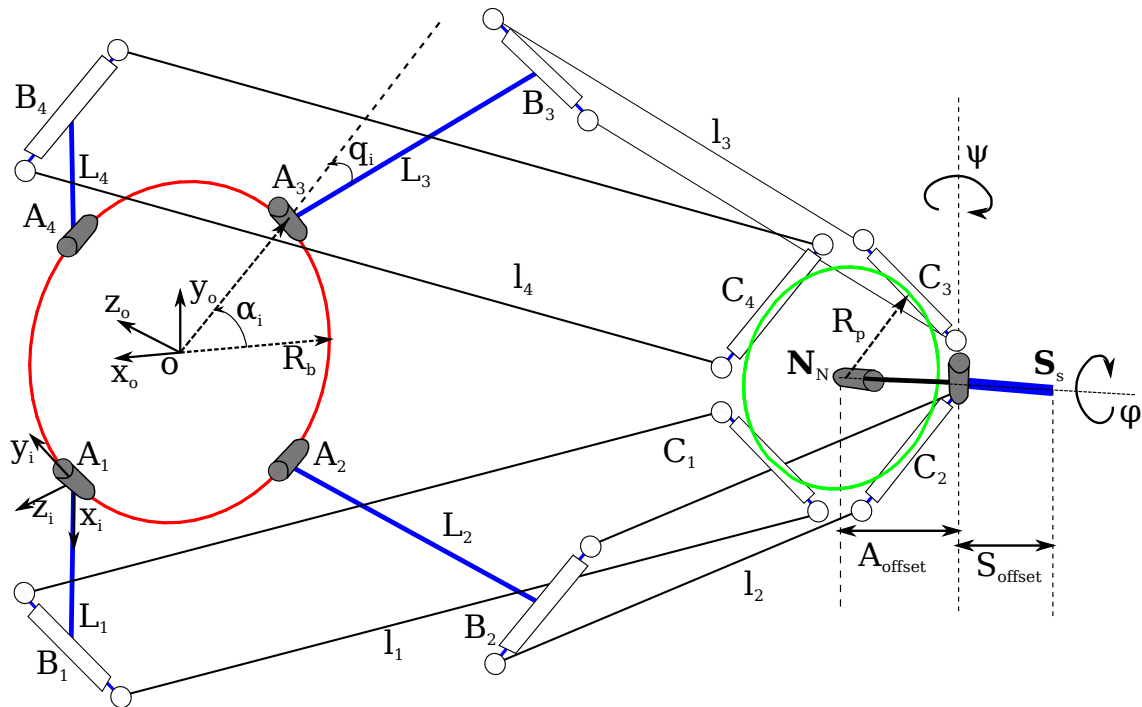


Figure 3: Illustration of the kinematics of SPIDER4

In order to find the nacelle position ${}^o\mathbf{N}_N$ it is fundamental to employ rotation matrices around the z and y axes.

These matrices are defined in this manner:

$$\mathbf{R}_Z = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{R}_Y = \begin{bmatrix} \cos(\psi) & 0 & \sin(\psi) \\ 0 & 1 & 0 \\ -\sin(\psi) & 0 & \cos(\psi) \end{bmatrix} \quad (1)$$

The above expressions involve the wrist mechanism variables (ϕ, ψ) , consequently ${}^o\mathbf{N}_N$ can be defined as follows:

$${}^o\mathbf{N}_N = \begin{bmatrix} x \\ y \\ z + A_{offset} \end{bmatrix} + \mathbf{R}_Z \mathbf{R}_Y \begin{bmatrix} 0 \\ 0 \\ S_{offset} \end{bmatrix} \quad (2)$$

Hence the final expression of ${}^o\mathbf{N}_N$ results in:

$${}^o\mathbf{N}_N = \begin{bmatrix} x + S_{offset} \cos(\phi) \sin(\psi) \\ y + S_{offset} \sin(\phi) \sin(\psi) \\ z + S_{offset} \cos(\psi) + A_{offset} \end{bmatrix} \quad (3)$$

where S_{offset} is the distance between the ψ joint axis and the tip of the spindle, and A_{offset} is the distance between the center of the nacelle and the ψ joint axis. Once ${}^o\mathbf{N}_N$ is computed, we can set the active and passive joints locations with respect to the Cartesian-fixed frame $O - x_o, y_o, z_o$.

The actuated joints located in the points \mathbf{A}_i , $\forall i = 1..4$, are expressed by the following vector:

$${}^o\mathbf{A}_i = R_b \begin{bmatrix} \cos(\alpha_i) & \sin(\alpha_i) & 0 \end{bmatrix}^T \quad (4)$$

where R_b is the fixed-base radius. The four main actuators are placed on the fixed base with the following angles with respect to the fixed Cartesian frame $\boldsymbol{\alpha} = \left[\frac{7\pi}{4} \quad \frac{5\pi}{4} \quad \frac{3\pi}{4} \quad \frac{\pi}{4} \right]^T$. The universal passive joints are located at the points \mathbf{B}_i and \mathbf{C}_i whose coordinates are expressed in the fixed reference frame $O - x_o, y_o, z_o$ as follows:

$${}^o\mathbf{B}_i = {}^o\mathbf{A}_i + L \begin{bmatrix} \cos(\alpha_i) \cos(q_i) & \sin(\alpha_i) \cos(q_i) & -\sin(q_i) \end{bmatrix}^T \quad (5)$$

$${}^o\mathbf{C}_i = \begin{bmatrix} R_p \cos(\alpha_i) + x_n & R_p \sin(\alpha_i) + y_n & z_n \end{bmatrix}^T \quad (6)$$

being L the arm length and R_p the nacelle radius. For further analysis an auxiliary frame located at ${}^o\mathbf{A}_i - x_i, y_i, z_i$ is defined, where the vectors ${}^i\mathbf{x}_i$ and ${}^i\mathbf{y}_i$ are defined as:

$${}^i\mathbf{x}_i = \begin{bmatrix} \cos(\alpha_i) & \sin(\alpha_i) & 0 \end{bmatrix}^T \quad (7)$$

$${}^i\mathbf{y}_i = \begin{bmatrix} -\sin(\alpha_i) & \cos(\alpha_i) & 0 \end{bmatrix}^T \quad (8)$$

The expression relating the operational nacelle variables ${}^o\mathbf{N}_N$ with the parallel structure joint variables \mathbf{q} is called the close-loop equation, which can be defined as:

$$\|{}^o\mathbf{C}_i - {}^o\mathbf{B}_i\|^2 = l_i^2 \quad (9)$$

According to [20] , [22] the expression (9) can be rewritten as follows to obtain the joint variables \mathbf{q} .

$$D_i \sin(q_i) + E_i \cos(q_i) + F_i = 0 \quad \forall i = 1, 2, 3, 4 \quad (10)$$

where $D_i = 2L_i({}^o\mathbf{A}_i^o\mathbf{C}_i \cdot \mathbf{z}_o)$, $E_i = 2L_i({}^o\mathbf{A}_i^o\mathbf{C}_i \cdot {}^i\mathbf{x}_i)$, and $F_i = l_i^2 - L_i^2 - \|{}^o\mathbf{A}_i^o\mathbf{C}_i\|^2$. Solving (10), representing the closed-loop equation, the values of q_i can be obtained by means of the following expression:

$$q_i = \arctan\left(\frac{-D_i \pm \sqrt{\Delta_i}}{F_i - E_i}\right) \quad (11)$$

Being $\Delta_i = D_i^2 + E_i^2 - F_i^2$.

The mechanism of the robot including all the coupling parts was design and assembled in SolidWorks Software. When the design of SPIDER4 was completed, it was built in collaboration with TECNALIA Company. Therefore, all kinematic parameters as the lengths of the limbs and the radius of the fixed base and nacelle were directly obtained from the CAD model. Thereby, the kinematic parameters of the robot are known with precision. These parameters are presented in Table 1

Table 1: Summary of the kinematic parameters of SPIDER4

Parameter	Description	Value
L	Rear-arm length	0.535 [m]
l	Forearm length	1.100 [m]
R_b	Fixed base radius	0.400 [m]
R_p	Nacelle radius	0.260 [m]

2.2. Jacobian analysis

Jacobian analysis plays an important role in the study of PKMs since it provides the relationship between the joint velocities $\dot{\mathbf{q}} = [\dot{q}_1 \quad \dot{q}_2 \quad \dot{q}_3 \quad \dot{q}_4]^T \in \mathbb{R}^{4 \times 1}$ and the operational velocities ${}^o\dot{\mathbf{N}}_N = [\dot{x}_n \quad \dot{y}_n \quad \dot{z}_n]^T \in \mathbb{R}^{3 \times 1}$. Furthermore, the Jacobian analysis constructs a transformation that relates the actuators' torques and the forces acting on the nacelle. The relationship between the joint and nacelle velocities can be expressed as follows:

$$\mathbf{J}_x^o \dot{\mathbf{N}}_N = \mathbf{J}_q \dot{\mathbf{q}} \quad (12)$$

where it is worth to note that the matrix $\mathbf{J}_x \in \mathbb{R}^{4 \times 3}$ is not square, then it is not invertible. However the matrix $\mathbf{J}_q \in \mathbb{R}^{4 \times 4}$ is square, and therefore invertible. The Jacobian matrices can be computed from the equiprojectivity principle [23], which establishes that the scalar product of the velocity of an extreme point of a vector by the vector is equal to the scalar product of the same vector by the velocity of the other extreme point. This principle is applied to the forearms as follows:

$$\mathbf{v}_{oB_i} \cdot {}^o\mathbf{B}_i^o\mathbf{C}_i = \mathbf{v}_{oC_i} \cdot {}^o\mathbf{B}_i^o\mathbf{C}_i \quad (13)$$

where \mathbf{v}_{oB_i} and \mathbf{v}_{oC_i} represent the velocities in \mathbf{B}_i and \mathbf{C}_i , respectively. Making use of Figure 3 we can note that velocities in the Nacelle are equal to those in the points \mathbf{C}_i so: $\mathbf{v}_{oC_i} = {}^o\dot{\mathbf{N}}_N$; however, the velocities in the points

\mathbf{B}_i are related to velocities in the active joints as follows: $\mathbf{v}_{oBi} = (\dot{\mathbf{q}} \cdot {}^i \mathbf{y}_i \times {}^o \mathbf{A}_i^o \mathbf{B}_i) \cdot {}^o \mathbf{B}_i^o \mathbf{C}_i$. Substituting (13) in (12) and making use of the velocity equivalences, we defined the Jacobian matrices as follows:

$$\mathbf{J}_x = \begin{bmatrix} {}^o \mathbf{B}_1^o \mathbf{C}_1 \cdot \mathbf{x}_o & {}^o \mathbf{B}_1^o \mathbf{C}_1 \cdot \mathbf{y}_o & {}^o \mathbf{B}_1^o \mathbf{C}_1 \cdot \mathbf{z}_o \\ {}^o \mathbf{B}_2^o \mathbf{C}_2 \cdot \mathbf{x}_o & {}^o \mathbf{B}_2^o \mathbf{C}_2 \cdot \mathbf{y}_o & {}^o \mathbf{B}_2^o \mathbf{C}_2 \cdot \mathbf{z}_o \\ {}^o \mathbf{B}_3^o \mathbf{C}_3 \cdot \mathbf{x}_o & \mathbf{B}_1 \mathbf{C}_3 \cdot \mathbf{y}_o & {}^o \mathbf{B}_3^o \mathbf{C}_3 \cdot \mathbf{z}_o \\ {}^o \mathbf{B}_4^o \mathbf{C}_4 \cdot \mathbf{x}_o & {}^o \mathbf{B}_4^o \mathbf{C}_4 \cdot \mathbf{y}_o & {}^o \mathbf{B}_4^o \mathbf{C}_4 \cdot \mathbf{z}_o \end{bmatrix} \quad (14)$$

$$\mathbf{J}_q = \begin{bmatrix} ({}^o \mathbf{B}_1^o \mathbf{C}_1 \times {}^o \mathbf{A}_1^o \mathbf{B}_1) \cdot {}^1 \mathbf{y}_1 & & 0 \\ & \ddots & \\ 0 & & ({}^o \mathbf{B}_4^o \mathbf{C}_4 \times {}^o \mathbf{A}_4^o \mathbf{B}_4) \cdot {}^4 \mathbf{y}_4 \end{bmatrix} \quad (15)$$

where the Cartesian frame unit vectors can be expressed by $\mathbf{x}_o = [1 \ 0 \ 0]^T$, $\mathbf{y}_o = [0 \ 1 \ 0]^T$, and $\mathbf{z}_o = [0 \ 0 \ 1]^T$, then the Jacobian matrix can be defined as:

$$\mathbf{J} = \mathbf{J}_q^{-1} \mathbf{J}_x \quad (16)$$

It maps the joint velocities in terms of the nacelle velocities as follows:

$$\dot{\mathbf{q}} = \mathbf{J}^o \dot{\mathbf{N}}_N \quad (17)$$

In order to obtain the nacelle velocities in terms of the joint velocities, the calculation of the pseudoinverse Jacobian matrix should be used, owing to \mathbf{J} not being square.

$$\mathbf{H} = \mathbf{J}^+ = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \quad (18)$$

being \mathbf{H} a non-square 3×4 matrix.

2.3. Forward Kinematic Model

The Forward Kinematic Model (FKM) allows us to find the values of the operational variables as a function of the joint variables. FKM is computed in two steps; the first one is to find ${}^o \mathbf{N}_N$ when \mathbf{q} is given, this can be achieved by solving a four-equation system representing the intersection of four virtual spheres whose radii are equal to the forearm length l . Figure 4 illustrates the intersection algorithm of the four virtual spheres used to obtain the FKM.

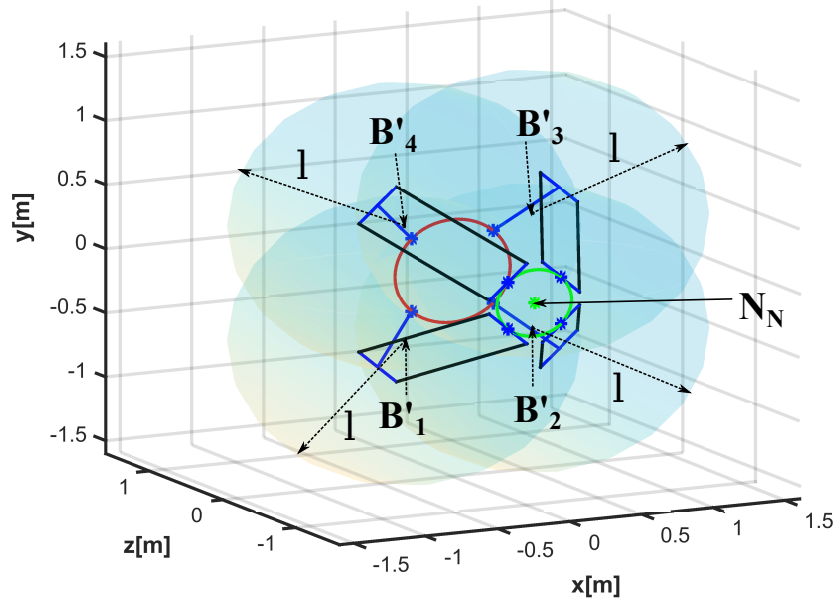


Figure 4: Illustration of the four virtual spheres intersection used to find ${}^o\mathbf{N}_N$ position

The second step lies in computing the spindle position ${}^o\mathbf{S}_S$ once ${}^o\mathbf{N}_N$ has been found. The sphere equation is given as follows:

$$x_n^2 + y_n^2 + z_n^2 - 2p_{xi}x_n - 2p_{yi}y_n - 2p_{zi}z_n + w_i = l^2 \quad \forall i = 1, \dots, 4 \quad (19)$$

where p_{xi} , p_{yi} , and p_{zi} represents the center coordinates of each one of the spheres and, $w_i = p_{xi}^2 + p_{yi}^2 + p_{zi}^2$. The centers of the virtual spheres are placed using the following expression:

$${}^o\mathbf{B}'_i = {}^o\mathbf{B}_i - {}^N\mathbf{N}_i \quad (20)$$

where

$${}^N\mathbf{N}_i = \begin{bmatrix} R_p \cos(\alpha_i) & R_p \sin(\alpha_i) & 0 \end{bmatrix}^T \quad (21)$$

solving system (18) leads to

$$y_n = d_1 z_n + e_1 \quad (22)$$

$$x_n = d_2 z_n + e_2 \quad (23)$$

$$z_n = \frac{-b - \sqrt{b^2 - 4ac}}{2a} \quad (24)$$

The auxiliary constants that are involved in expressions (21)-(23) are defined as follows:

$$\begin{aligned} a_1 &= \frac{p_{y1} - p_{y3}}{p_{x3} - p_{x1}} & b_1 &= \frac{p_{z1} - p_{z3}}{p_{x3} - p_{x1}} & c_1 &= \frac{w_3 - w_1}{2(p_{x3} - p_{x1})} & d_1 &= \frac{b_2 - b_1}{a_1 - a_2} & e_1 &= \frac{c_2 - c_1}{a_1 - a_2} \\ a_2 &= \frac{p_{y2} - p_{y3}}{p_{x3} - p_{x1}} & b_2 &= \frac{p_{z2} - p_{z3}}{p_{x3} - p_{x2}} & c_2 &= \frac{w_3 - w_2}{2(p_{x3} - p_{x2})} & d_2 &= a_1 d_1 + b_1 & e_2 &= a_1 e_1 + c_1 \end{aligned}$$

$$\begin{aligned}
a &= d_1^2 + d_2^2 + 1 \\
b &= 2[(e_1 - p_{y4})d_1 + (e_2 - p_{x4})d_2 - p_{z4}] \\
c &= e_1^2 + e_2^2 - 2(e_1 p_{y4} + e_2 p_{x4}) + w_4 - l^2
\end{aligned}$$

Once the positions of the nacelle (x_n, y_n, z_n) are found, an auxiliary vector is defined in order to find the spindle position; the auxiliary vector involves the variables of the wrist mechanism (ϕ, ψ) , and is expressed as:

$${}^o\mathbf{N}_{aux} = {}^o\mathbf{N}_N - \begin{bmatrix} S_{offset} \cos(\phi) \sin(\psi) \\ S_{offset} \sin(\phi) \sin(\psi) \\ S_{offset} \cos(\psi) + A_{offset} \end{bmatrix} \quad (25)$$

Finally, the spindle position is defined through the following expression:

$${}^o\mathbf{S}_S = \left[{}^o\mathbf{N}_{aux}^T \quad \phi \quad \psi \right]^T \quad (26)$$

2.4. Simplified Inverse Dynamic Model of the delta-like positioning mechanism of SPIDER4

SPIDER4 relies on the same mechanical structure as another RA-PKM called R4 [24]; nonetheless, some differences exist between both PKMs, including: the two rotational actuators incorporated at SPIDER4 nacelle, including the orientation of the kinematic chains and including the nacelle. In the case of R4 the z axis is oriented vertically, whereas the z axis is oriented horizontally in the case of SPIDER4. In [24] a simplified dynamic model was proposed for R4 RA-PKM, but this dynamic model does not take into account the gravity term, because this PKM was designed for high acceleration (up to 100G) pick-and-place tasks. However, SPIDER4 is designed to perform machining operations which are mainly executed at low speeds and accelerations so. For this case it is necessary to take into account the gravity acceleration term in the dynamic model. The developed simplified inverse dynamic model should be suitable for control tasks in real-time experiments. To this end the following simplifications to take into account discussed in more detail in [24] [9] are considered.

- The friction forces are neglected.
- The masses of the forearms are smaller than the other parts of SPIDER4, thus its inertia can be neglected. The mass of each forearm is divided into two parts, one half of each forearm mass is added to the mass of the corresponding arm and the other half is added to the nacelle mass.

The frictional effects were neglected because it is complex to obtain an accurate friction model for a highly nonlinear PKM as SPIDER4. Since the friction forces depend of different factors as the type of materials in contact, the lubrication of the joints, and even the temperature. For these reasons, we selected a robust nonlinear controller as, namely RISE owing to guarantee asymptotic stability despite the presence of dynamic uncertainties. The details of the control law will be explained in Section 3. Concerning the second simplification, the ratio between the mass of the forearms and the mass of the rear-arms must be sufficiently small (that is less or equal than 0.3 equaling to a 2% error with respect to a complete dynamic model), as it is established on the analysis developed for PKM with delta-like architecture in [25]. Taking into account the dynamic parameters of SPIDER4 presented in Table 2, one

can note that the mass of one rear-arm is 17.6 Kg and the mass of one forearm is 4.64 Kg considering the two parallel bars yielding a ratio of 0.26. Therefore, the second simplification is suitable to be applied on the SPIDER4 dynamic analysis. The present simplified inverse dynamic model considers only the dynamics of the parallel structure of SPIDER4 without considering the dynamics of the wrist. Nonetheless, the mass of each element of the wrist is taken into account for the analysis. The inverse dynamic model formulation can be divided into forces and torques computations over the nacelle and the limbs. Considering the nacelle performing only translation movements, it would be possible avoid the torque calculation. The Nacelle dynamics is described by Newton-Euler formulation as follows:

$$\mathbf{M}_p({}^o\ddot{\mathbf{N}}_N - \mathbf{G}) = 0 \quad (27)$$

The gravity force acting on the nacelle can be expressed by:

$$\mathbf{G}_p = -\mathbf{M}_p\mathbf{G} \quad (28)$$

where $\mathbf{M}_p = \text{diag}\{m_t \ m_t \ m_t\} \in \mathbb{R}^{3 \times 3}$ is the mass matrix whose elements are given by:

$$m_t = m_{na} + m_{namotors} + 4\frac{m_f}{2} \quad (29)$$

where m_{na} is the nacelle mass, m_f is the forearm mass composed by the two parallelogram bars mass, $m_{namotors}$ is the mass of the three motors located at the nacelle and their coupling parts. The motors located at the nacelle include: the motor generating the movement around ϕ , the motor producing the movement over ψ , and the spindle motor. The gravity vector $\mathbf{G} \in \mathbb{R}^{3 \times 1}$ is denoted by $\mathbf{G} = [0 \ g \ 0]^T$ with $g = 9.81 \text{ m/s}^2$. It is worth to note that gravity acceleration is along the y axis instead of z axis due to the orientation of the parallel structure of the robot (Figure 2).

The inertial forces acting on the nacelle owing to the Cartesian acceleration ${}^o\ddot{\mathbf{N}}_N \in \mathbb{R}^{3 \times 1}$ are expressed by:

$$\mathbf{F}_p = \mathbf{M}_p{}^o\ddot{\mathbf{N}}_N \quad (30)$$

The contributions of \mathbf{G}_p and \mathbf{F}_p in joint space can be calculated using the pseudoinverse Jacobian matrix $\mathbf{H}(\mathbf{q}, {}^o\mathbf{N}_N) \in \mathbb{R}^{3 \times 4}$. These contributions are expressed through the following relationship:

$$\Gamma_{Gp} = -\mathbf{H}^T\mathbf{M}_p\mathbf{G} \quad (31)$$

$$\Gamma_{Fp} = \mathbf{H}^T\mathbf{M}_p{}^o\ddot{\mathbf{N}}_N \quad (32)$$

Having the dynamic equations for the nacelle, the next step consists to identify the dynamics of the acting mechanism which is formed by the set of actuators, rear-arms, and forearms. Similarly, as the nacelle, it should compute the torques produced by joint acceleration and gravity acceleration. The torques produced by joint acceleration on the kinematic chains $\Gamma_{Ta} \in \mathbb{R}^{4 \times 1}$ are expressed in this way.

$$\Gamma_{Ta} = \mathbf{I}_t\ddot{\mathbf{q}} \quad (33)$$

where $\mathbf{I}_t \in \mathbb{R}^{4 \times 4}$ is a diagonal matrix whose elements are formed by:

$$I = I_{act} + I_{arm} + \frac{L_i^2 m_f}{2} \quad (34)$$

where I_{act} , I_{arm} represents the actuators inertia and the arms inertia, respectively. The term $\frac{L_i^2 m_f}{2}$ corresponds to the inertial contribution of the forearms using the second simplification where, L is the arm's length and m_f is the forearm mass constituted by the parallelogram bars.

In order to obtain the torques produced by the gravity acceleration acting on the kinematic chains, we have to take into account the orientation of the manipulator since the gravity force affects differently each one of the kinematic chains as is appreciated in Figure 3. The gravity effect over the limbs of SPIDER4 is directly related with the fixed angular orientation of the linkages α_i . We may define the vector of torques produced by the gravitational forces acting on the kinematic chains in this form:

$$\mathbf{\Gamma}_{Ga} = -\mathbf{M}_{Ra}g \cos(q) \quad (35)$$

where $\cos(q) = [\cos(q_1) \quad \cos(q_2) \quad \cos(q_3) \quad \cos(q_4)]^T$ the matrix $\mathbf{M}_{Ra} \in \mathbb{R}^{4 \times 4}$ is composed as follows:

$$\mathbf{M}_{Ra} = \begin{bmatrix} m_{ra} \sin(\alpha_1) & 0 & 0 & 0 \\ 0 & m_{ra} \sin(\alpha_2) & 0 & 0 \\ 0 & 0 & m_{ra} \sin(\alpha_3) & 0 \\ 0 & 0 & 0 & m_{ra} \sin(\alpha_4) \end{bmatrix} \quad (36)$$

where $m_{ra} = m_a L_c + \frac{m_f L}{2}$, L_c is the distance to the center of mass of each arm, and m_f is the mass of each forearm. Relating the acting forces on the nacelle and the limbs, both inertial and non-inertial, as established in the principle of virtual works, leads to:

$$\mathbf{\Gamma} + \mathbf{\Gamma}_{Gp} + \mathbf{\Gamma}_{Ga} = \mathbf{\Gamma}_{Ta} + \mathbf{\Gamma}_{Fp} \quad (37)$$

To express the dynamic model of the delta-like positioning device of SPIDER4 in joint space, we proceed to use the following relationship, based on the pseudoinverse Jacobian matrix and its derivative:

$${}^o\ddot{\mathbf{N}}_N = \mathbf{H}\ddot{\mathbf{q}} + \dot{\mathbf{H}}\dot{\mathbf{q}} \quad (38)$$

By rearranging the terms, the simplified inverse dynamic model of the delta-like positioning mechanism of SPIDER4 is expressed terms of the redundant joint coordinates (\mathbf{q}) as follows [26]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{\Gamma} \quad (39)$$

with:

- $\mathbf{M}(\mathbf{q}) = \mathbf{I}_t + \mathbf{H}^T \mathbf{M}_p \mathbf{H}$
- $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{H}^T \mathbf{M}_p \dot{\mathbf{H}}$
- $\mathbf{G}(\mathbf{q}) = -\mathbf{\Gamma}_{Gp} - \mathbf{\Gamma}_{Ga}$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{4 \times 4}$ denotes the inertia matrix of the delta-like positioning device of SPIDER4, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{4 \times 4}$ is the Coriolis/Centripetal forces matrix satisfying the skew-symmetry property; $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^{4 \times 1}$ is the vector of forces and torques due to gravity being continuous and bounded.

The dynamic parameters of SPIDER4 as the masses of the rear-arms, forearms, the nacelle, coupling parts, and

rear-arms inertia were calculated through material assignation functionality of SolidWorks. The other dynamic parameters as the inertia of the motors and their masses were obtained from the datasheets of the manufacturers of the actuators. The values of dynamical parameters are provided in Table 2.

Table 2: Summary of the dynamic parameters of SPIDER4

Parameter	Description	Value
m_{na}	Nacelle mass	22.76 Kg
$m_{namotors}$	Mass of the three motors located on the Nacelle	19.5 Kg
m_a	Rear-arm mass	17.6 Kg
m_f	Forearm mass	4.64 Kg
I_{arm}	Rear-arm inertia	1.69 Kg m^2
I_{act}	Inertia of one of the principal actuators	0.00223 Kg m^2
m_s	Mass of the spindle motor	3.2 Kg
m_ψ	Motor mass that regulates movement in ψ	5.1 Kg
m_ϕ	Motor mass that regulates movement in ϕ	11.2 Kg

2.5. Inverse Dynamic Model of the Wrist

In the previous subsection, a simplified inverse dynamic model for the delta-like positioning device of SPIDER4 was derived. Notwithstanding, to derive the inverse dynamics of the whole manipulator, it is essential to obtain the inverse dynamic model of the wrist. The wrist is a serial mechanism formed by two links actuated by two rotational joints. The inverse dynamics model of this part of the robot is computed using Euler-Lagrange formulation. Therefore, we need first to obtain the forward kinematic model of the wrist, which is obtained considering the equation (25). The forward kinematic model of the wrist is expressed in the nacelle reference frame $N - x_n, y_n, z_n$ as follows:

$${}^N\mathbf{W}_W = \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = \begin{bmatrix} -S_{offset} \cos(\phi) \sin(\psi) \\ -S_{offset} \sin(\phi) \sin(\psi) \\ -S_{offset} \cos(\psi) + A_{offset} \end{bmatrix} \quad (40)$$

The Euler-Lagrange formulation requires the calculation of the kinetic and potential energy of the wrist. Taking the derivative of (40) the kinetic energy of the wrist can be written as:

$$\mathbf{K}_w = \frac{1}{2} m_s {}^N\dot{\mathbf{W}}_W^{TN} \dot{\mathbf{W}}_W + \frac{1}{2} I_{m\phi} \dot{\phi} \quad (41)$$

where m_s is the mass of the spindle motor and its coupling parts and $I_{m\phi}$ is the rotational inertia produced by the mass of the second motor on the nacelle and its coupling parts.

The Potential energy of the wrist may be expressed in the following form, considering that the gravity acceleration is over y axis.

$$\mathbf{U}_w = -m_s g S_{offset} \sin(\phi) \sin(\psi) \quad (42)$$

Therefore, the Lagrangian function is established in function of $\boldsymbol{\xi} = \begin{bmatrix} \phi & \psi \end{bmatrix}^T$, as follows [27]:

$$\mathcal{L}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) = \mathbf{K}_w(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) - \mathbf{U}_w(\boldsymbol{\xi}) \quad (43)$$

The Lagrangian function (43) is evaluated in the Euler-Lagrange dynamic equation, in order to obtain the inverse dynamic model of the wrist.

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}})}{\partial \dot{\boldsymbol{\xi}}} \right] - \left[\frac{\partial \mathcal{L}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}})}{\partial \boldsymbol{\xi}} \right] = \boldsymbol{\Gamma}_w \quad (44)$$

Being $\boldsymbol{\Gamma}_w \in \mathbb{R}^{2 \times 1}$ the torque vector of the wrist actuators. Developing the equation (44); the inverse dynamic model of the wrist is obtained in the function of the joint coordinates of the wrist as shown:

$$\mathbf{M}_w(\boldsymbol{\xi})\ddot{\boldsymbol{\xi}} + \mathbf{C}_w(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}})\dot{\boldsymbol{\xi}} + \mathbf{G}_w(\boldsymbol{\xi}) = \boldsymbol{\Gamma}_w \quad (45)$$

being

$$\mathbf{M}_w(\boldsymbol{\xi}) = \begin{bmatrix} m_s S_{offset}^2 \sin \psi + I_{m\phi} & 0 \\ 0 & m_s S_{offset}^2 \end{bmatrix} \quad \mathbf{C}_w(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) = \begin{bmatrix} 0.5 m_s S_{offset} \cos \psi \dot{\phi} & 0 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{G}_w(\boldsymbol{\xi}) = \begin{bmatrix} -m_s g S_{offset} \sin \psi \cos \phi \\ -m_s g S_{offset} \sin \phi \cos \psi \end{bmatrix}$$

where $\mathbf{M}_w(\boldsymbol{\xi}) \in \mathbb{R}^{2 \times 2}$ denotes the inertia matrix of the wrist, $\mathbf{C}_w(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) \in \mathbb{R}^{2 \times 2}$ is the Coriolis/Centripetal forces of the wrist matrix. and $\mathbf{G}_w(\boldsymbol{\xi}) \in \mathbb{R}^{2 \times 1}$ is the vector of gravity forces for the wrist.

2.6. Inverse Dynamic Model of SPIDER4 RA-PKM

The whole inverse dynamic model of SPIDER4 can be expressed in function of the generalized coordinates, as a result of the union of the inverse dynamic model of the positioning delta-like device (39), and the inverse dynamic model of the wrist (45), yielding:

$$\mathbf{M}(\mathbf{Q})\ddot{\mathbf{Q}} + \mathbf{C}(\mathbf{Q}, \dot{\mathbf{Q}})\dot{\mathbf{Q}} + \mathbf{G}(\mathbf{Q}) = \boldsymbol{\Gamma}_T \quad (46)$$

being:

$$\mathbf{M}(\mathbf{Q}) = \begin{bmatrix} \mathbf{M}(\mathbf{q})_{[4 \times 4]} & \mathbf{0}_{[2 \times 4]} \\ \mathbf{0}_{[2 \times 4]} & \mathbf{M}_w(\boldsymbol{\xi})_{[2 \times 2]} \end{bmatrix} \quad \mathbf{C}(\mathbf{Q}, \dot{\mathbf{Q}}) = \begin{bmatrix} \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})_{[4 \times 4]} & \mathbf{0}_{[2 \times 4]} \\ \mathbf{0}_{[2 \times 4]} & \mathbf{C}_w(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}})_{[2 \times 2]} \end{bmatrix}$$

$$\mathbf{G}(\mathbf{Q}) = \begin{bmatrix} \mathbf{G}(\mathbf{q})_{[4 \times 1]} \\ \mathbf{G}_w(\boldsymbol{\xi})_{[2 \times 1]} \end{bmatrix} \quad \boldsymbol{\Gamma}_T = \begin{bmatrix} \boldsymbol{\Gamma}_{[4 \times 1]} \\ \boldsymbol{\Gamma}_w_{[2 \times 1]} \end{bmatrix} \quad \mathbf{Q} = \begin{bmatrix} \mathbf{q}_{[4 \times 1]} \\ \boldsymbol{\xi}_{[2 \times 1]} \end{bmatrix}$$

where $\mathbf{M}(\mathbf{Q}) \in \mathbb{R}^{6 \times 6}$ denotes the inertia matrix of SPIDER4, $\mathbf{C}(\mathbf{Q}, \dot{\mathbf{Q}}) \in \mathbb{R}^{6 \times 6}$ is the Coriolis/Centripetal forces matrix of SPIDER4, $\mathbf{G}(\mathbf{Q}) \in \mathbb{R}^{6 \times 1}$ is the vector of gravity forces and, $\boldsymbol{\Gamma}_T \in \mathbb{R}^{6 \times 1}$ is the torque vector of the whole actuators of SPIDER4.

Note: The inverse dynamic model introduced in (39) is taken for subsequent control development instead of (46) owing to the delta-like, and the wrist mechanisms are controlled independently. The software of the robot allows us to modify only the control scheme for the delta-like mechanism; however, for the wrist, that is not possible. In the next section, the control scheme for the delta-like positioning will be presented.

3. Motion Control

3.1. The actuation redundancy

Actuation redundancy in robotics means that the system has more actuators than DOF [7]. Robotics systems with this type of configuration have advantages compared with their non-redundant counterparts, such as higher accuracy and improved stiffness. Furthermore, actuation redundancy can also lead to singularity-free large workspaces. However, this configuration leads to the generation of internal forces that may create pre-stress in the mechanism without operational motions and can damage the mechanical structure of the robot. According to [7] the internal forces can be caused by geometric uncertainties and may be amplified by the use of decentralized control techniques. This may lead to uncoordinated control of the individual actuators, since such a control law does not account for kinematic constraints. In order to avoid such an issue, the use of a projection matrix can be proposed, it is based on the pseudo inverse jacobian matrix evaluated with the desired variables $\mathbf{H}(\mathbf{q}_d, {}^o\mathbf{N}_{Nd}) \in \mathbb{R}^{3 \times 4}$. The projection operator is defined by:

$$\mathbf{R}_H = (\mathbf{H}^+)^T \mathbf{H}^T \quad (47)$$

The projection matrix \mathbf{R}_H eliminates the control inputs in the null-space of \mathbf{H}^T . Hence, all control inputs applied to the SPIDER4 have to be 'regularized' using this projection matrix as follows.

$$\mathbf{\Gamma}^* = \mathbf{R}_H \mathbf{\Gamma} \quad (48)$$

where $\mathbf{\Gamma}$ denotes the torques vector generated by the control scheme.

3.2. Controller description

Milling and drilling tasks require a high precision in order to obtain good quality end products. To satisfy this demand, besides having an adequate mechanical structure, it is also necessary to design a good controller that allows SPIDER4 end-effector to follow a reference trajectory with the minimum possible tracking error. RISE is a robust feedback control strategy developed for uncertain nonlinear systems with a unique integral signum term that can ensure semi-global asymptotic stability in the presence of general uncertain disturbances [28]. The main advantage of RISE is that, based on limited assumptions about the system, it can compensate for a large class of uncertainties; this is of great importance when the dynamic parameters of the system to be controlled are totally or partially unknown. To understand concept of limited assumptions it is necessary to express the dynamic model of the PKM in the following form:

$$\ddot{\mathbf{q}} = -\mathbf{M}(\mathbf{q})^{-1}[\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q})] + \mathbf{M}(\mathbf{q})^{-1}\mathbf{\Gamma} \quad (49)$$

From the previous equation we let us define $\mathbf{E}(\cdot) = -\mathbf{M}(\mathbf{q})^{-1}[\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q})]$ and $\mathbf{D}(\cdot) = \mathbf{M}(\mathbf{q})^{-1}$. According to [28] the limited assumptions to guarantee asymptotic stability using RISE controller are: $\mathbf{E}(\cdot)$ and $\mathbf{D}(\cdot)$ must be second-order differentiable and $\mathbf{D}(\cdot)$ must be a positive-definite symmetric matrix. According to these established

assumptions, we can see that our system is suitable to be regulated by the RISE controller. Let, the position tracking error in joint space $\mathbf{e}_q(t) \in \mathbb{R}^{4 \times 1}$, be defined as:

$$\mathbf{e}_q = \mathbf{q}_d - \mathbf{q} \quad (50)$$

RISE control uses filtered tracking errors, denoted as follows:

$$\mathbf{e}_1 = \dot{\mathbf{e}}_q + \mathbf{\Lambda}_1 \mathbf{e}_q \quad (51)$$

$$\mathbf{e}_2 = \dot{\mathbf{e}}_1 + \mathbf{\Lambda}_2 \mathbf{e}_1 \quad (52)$$

where $\mathbf{\Lambda}_1, \mathbf{\Lambda}_2 \in \mathbb{R}^{4 \times 4}$ are positive-definite, diagonal gain matrices. RISE feedback control law is defined by the following expression:

$$\mathbf{\Gamma}_{RISE} = (\mathbf{K}_s + \mathbf{I})\mathbf{e}_1(t) - (\mathbf{K}_s + \mathbf{I})\mathbf{e}_1(t_0) + \int_0^t [(\mathbf{K}_s + \mathbf{I})\mathbf{\Lambda}_2 \mathbf{e}_1(\tau) + \mathbf{\beta} \mathbf{sgn}(\mathbf{e}_1(\tau))] d\tau \quad (53)$$

where $\mathbf{K}_s, \mathbf{\beta} \in \mathbb{R}^{4 \times 4}$ are positive-definite, diagonal matrices, $\mathbf{I} \in \mathbb{R}^{4 \times 4}$ is the identity matrix, and $\mathbf{sgn}(\cdot)$ is the vector of the sign functions of the first filtered tracking error. The term $(\mathbf{K}_s + \mathbf{I})\mathbf{e}_1(t_0)$ is used to ensure a zero initial control input at $t = 0$. To improve the performance of the RISE feedback controller, it is possible to add a model-based feedforward term, as it has been reported in [16], [29]. If the robot is intended to perform tasks where its dynamic parameters may vary, e.g. pick and place tasks where the payload changes, it is possible to integrate an adaptive feedforward model-based compensation term. However, for machining tasks the dynamic parameters of the robot are not time-varying, then if the dynamic parameters are known or well approximated, they can be included in the feedforward compensation term. Hence the feedforward compensation term is expressed as [14]:

$$\mathbf{M}(\mathbf{q}_d)\ddot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d)\dot{\mathbf{q}}_d + \mathbf{G}(\mathbf{q}_d) = \mathbf{\Gamma}_{FW} \quad (54)$$

The above equation uses the desired joint variables instead of the measured ones, which is a great advantage owing to the majority of manipulators are equipped with only encoders for joint positions measurements. The velocities and accelerations are usually unmeasurable, therefore they have to be estimated through observers to subsequently be used in the inverse dynamic model implying more computational time. The feedforward part is used to enhance the overall performance of the feedback control scheme (in our case RISE controller), reducing the magnitude of overshoots in the transient response and stabilizing deviations from the desired trajectory [30], [31]. Moreover, the feedforward part is evaluated with the desired joint trajectories instead of the measured ones reducing the calculation time of the control action and avoiding the inclusion of noise that may come from the measurements [14], [32]. Considering the RISE controller (51), the feedforward compensation term (52), and the projection matrix (39), the final control law applied to SPIDER4 can be expressed as follows:

$$\mathbf{\Gamma}^* = \mathbf{R}_H(\mathbf{\Gamma}_{RISE} + \mathbf{\Gamma}_{FW}) \quad (55)$$

The control block-diagram is illustrated in Figure 5.

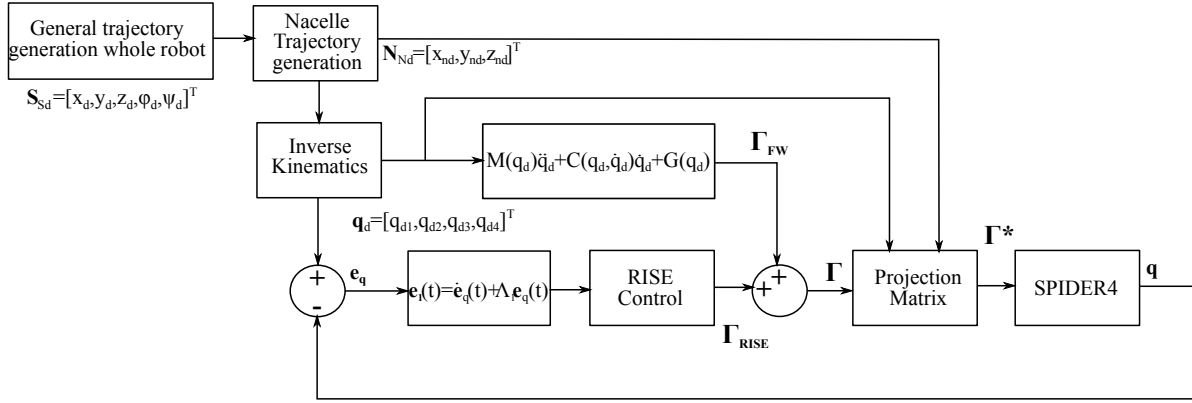


Figure 5: Block-diagram of the proposed control strategy for SPIDER4 PKM

4. Experimental results

4.1. Description of the experimental testbed

The proposed control scheme, presented in section 3, has been implemented on the experimental platform of SPIDER4, while considering the mathematical models, developed in section 2. To perform the movements, SPIDER4 has different type of actuators; the main four actuators are WITTESTEIN TPMA110S-022M-6KB1-220H-W6 motors, each actuator can provide a peak torque of 3100 Nm ; besides, this kind of actuators include a gear ratio of 1:22. To measure the joint positions these motors are equipped with multiturn absolute encoders. The joint velocities of SPIDER4 are not measured directly because the robot is not equipped with sensors to measure the joint velocities. Notwithstanding, they are calculated through numerical derivatives to form the measurements of the joint positions with a sampled time of 0.0004 sec. On the nacelle there are three actuators which perform the independent angular motions ϕ and ψ as well as the movement of the spindle. The motor responsible of the movement in the ϕ axis is STOBER EZH501USVC4P097 motor, this actuator can generate a peak torque of 200 Nm . To perform the angular motion in the ψ axis, a HARMONIC DRIVE CHA-20A-30-H-M1024-B is used, this motor can provide a peak torque of 27 Nm . Finally, the B&R 8JSA24.E4080D000-0 is used for actuation of the spindle machining tool, it can provide a torque of 1.41 Nm and a related speed of 8000 $r.p.m$. The proposed control scheme is developed in MATLAB-SIMULINK and B&R AUTOMATION STUDIO. The programmed control scheme is executed through the B&R Automation PC 910 and X20 system with a sampling time of 0.4 msec. The experimental setup is illustrate in Figure 6.

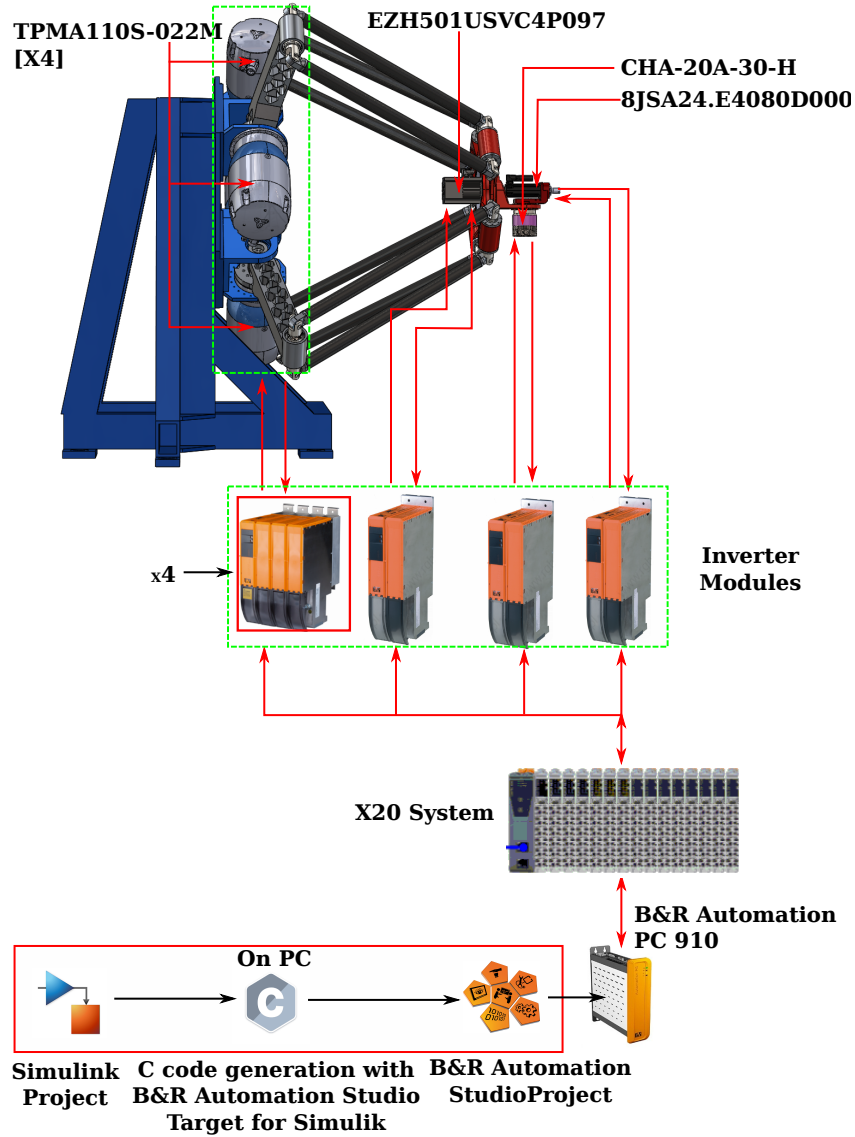


Figure 6: Illustration of experimental setup of the SPIDER4

4.2. RISE feedforward evaluation

The proposed RISE feedforward controller is evaluated and compared with the standard RISE controller through a trajectory tracking task. The desired trajectories are generated using the CNC functionality of B&R AUTOMATION STUDIO in G code. The most important G commands for the trajectory generation are the $G01$, $G02$ and $G03$ which are functions for linear interpolation, clockwise circular interpolation and counterclockwise circular interpolation respectively. Each line for interpolation in the G code must specify the coordinates of the desired point in mm and the Desired Cartesian Velocity (F) in mm/min. The desired trajectories were designed to contemplate the limits of the workspace. We had to use the CNC functionality of B&R Automation Studio to set the desired trajectories; however, due to software restrictions, it was not possible to modify the interpolation algorithm for the desired points. It is worth mentioning that the desired operational trajectories involves the whole actuated joints of SPIDER4. Nevertheless the proposed controller ensures only the trajectory tracking for nacelle of SPIDER4;

since the two motors located on the nacelle being controlled independently. The desired trajectories with respect to time for the whole joints of SPIDER4 are illustrated in Figure 7 including the parallel structure and the wrist mechanism. Additionally Figure 7 shows also a 3D representation of the desired trajectory. The desired trajectory for the nacelle is generated using (3) and is shown in Figure 8. Since to SPIDER4 is a highly nonlinear system, then does not exist an analytical method to tune the feedback gains of the controller. Therefore we tuned experimentally each one of the gains via trial-and-error method. Firstly we established a value large enough for α_1 and K_s , then we set a value of 1 for α_2 subsequently, that value was progressively increased until obtaining the minimum possible tracking error. The value of the gain β for the robustness part of the controller was selected in such a way to minimize the effect of chattering. The controllers gains are summarized in Table 3.

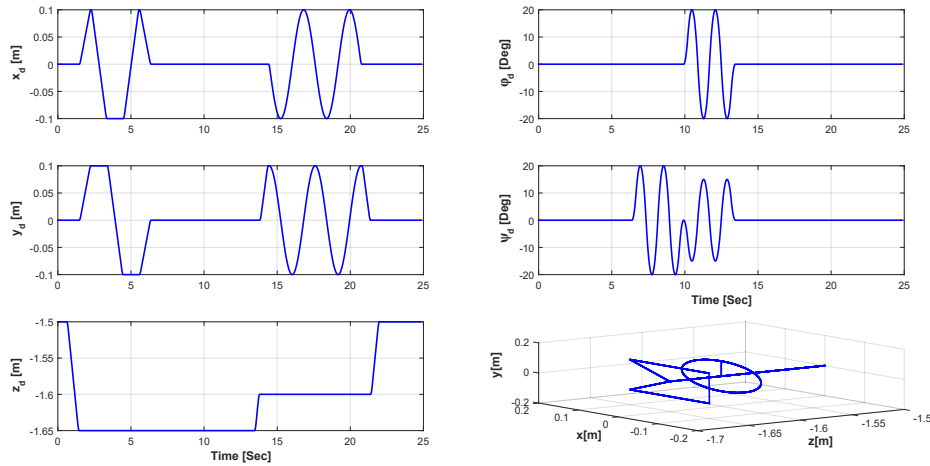


Figure 7: Evolution of the desired trajectories in Cartesian Space versus time

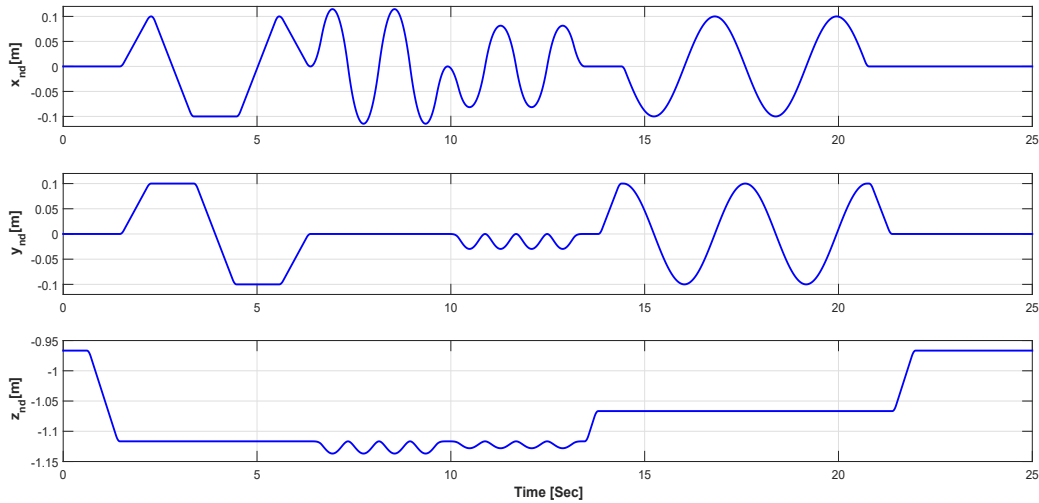


Figure 8: Desired trajectory of the nacelle in Cartesian space versus time

Table 3: Controller Gains

$$\mathbf{K}_s = \text{diag}[35 \ 35 \ 35 \ 35]$$

$$\mathbf{\Lambda}_1 = \text{diag}[110 \ 110 \ 110 \ 110]$$

$$\mathbf{\Lambda}_2 = \text{diag}[10 \ 10 \ 10 \ 10]$$

$$\boldsymbol{\beta} = \text{diag}[8 \ 8 \ 8 \ 8]$$

The different tracking errors in Cartesian and joint space for the Standard RISE and RISE feedforward are depicted in Figures 9 and 10 respectively. It can be seen that the second one greatly reduces the overshoots with respect to the first one and even though both tracking errors for whole trajectory are small. The tracking errors in Cartesian space depicted in Figure 9 were calculated using the FKM because the controllers are in function of the joint variables since there is no sensor available to measure the nacelle position directly.

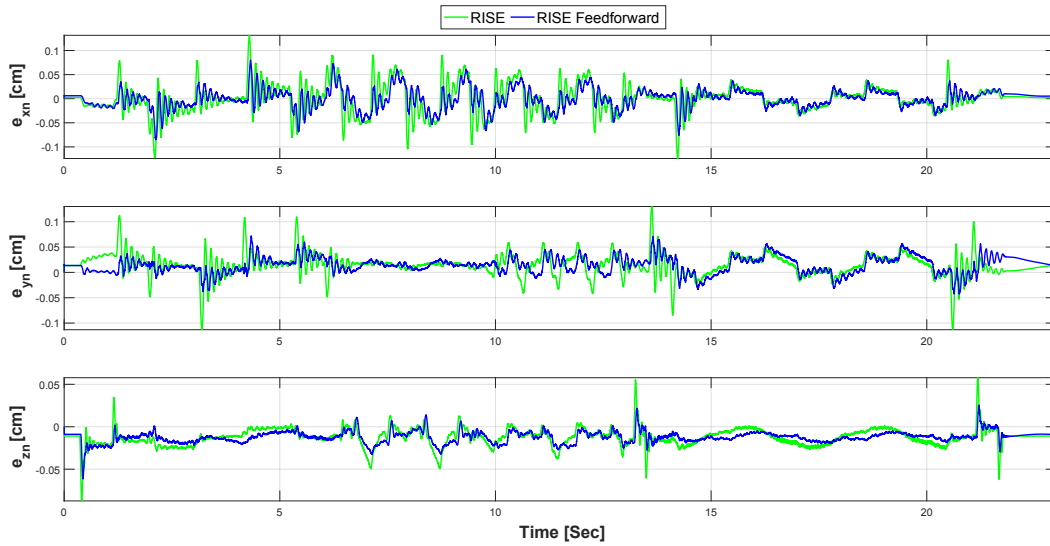


Figure 9: Evolution of the tracking errors in Cartesian Space versus time

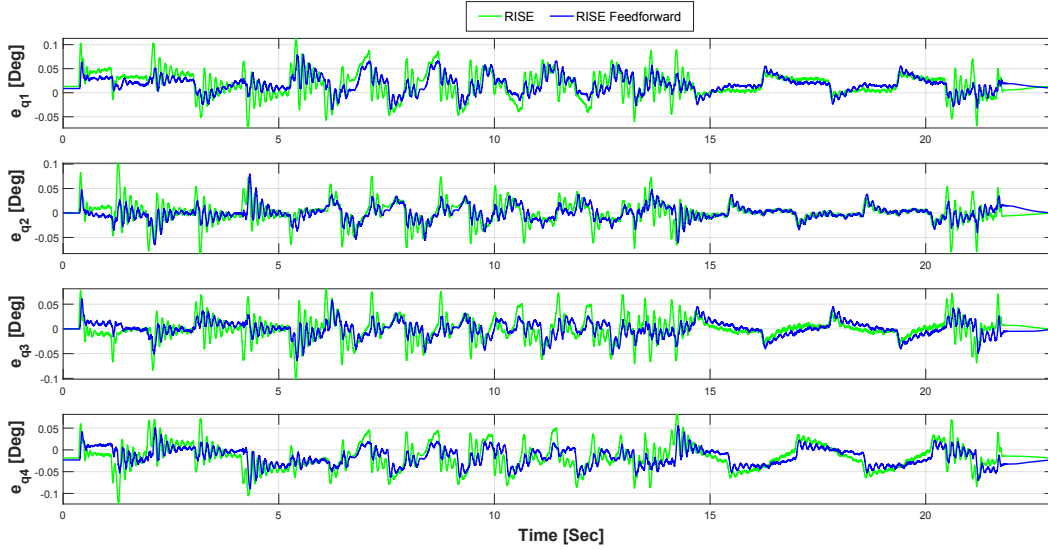


Figure 10: Evolution of the tracking error in joint space versus time

The generated torques by the RISE feedforward controller are illustrated in Figure 11, where we can note that the steady state values for the produced torques Γ_1 and Γ_2 are positive, while the torques Γ_3 , and Γ_4 are negative due to the orientation of the kinematic chains and the gravity affects which are different for the upper and the lower chains. From Figure 11 also we can note the presence of noise in the control signals. The noise phenomena came from the fact that measured encoder signals yield noise even if it is not very noticeable thereby, when these signals are processed in the control loop the numerical derivative operations amplify the effect of the noise, resulting in noisy control signals. We did not use a filter at the output of the controller since it may affect the robustness of the controller, in addition to that the computational cost may also increase without significant improvement in the performance of the overall system.

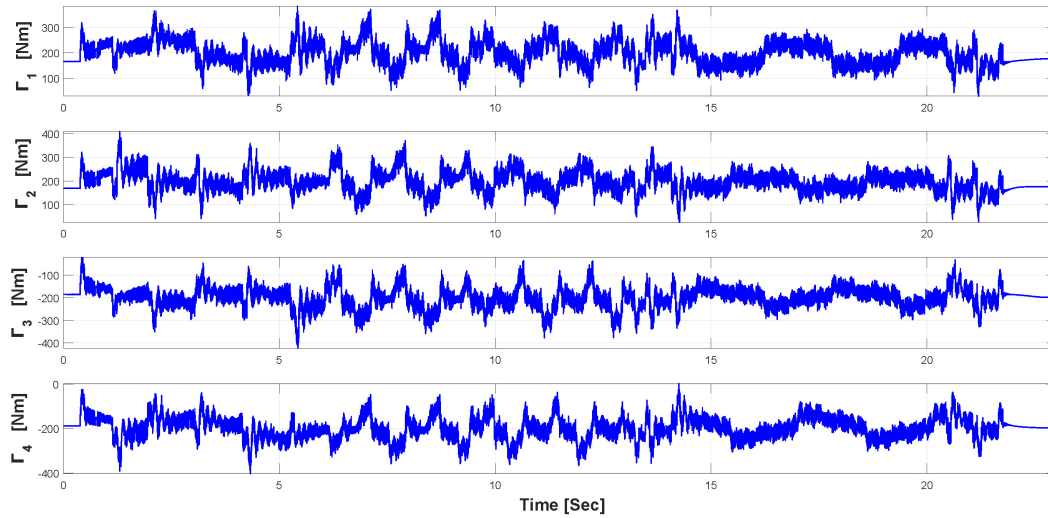


Figure 11: Evolution of the torques generated by RISE Feedforward controller versus time

To quantify the performance of the proposed controller, the Root Mean Square Error (RMSE) formula is used. It

enables to quantify how much the actual trajectory is close to the desired one. The RMSE expressions for Cartesian and joint spaces are given as follows respectively:

$$RMSE_C = \sqrt{\frac{1}{N} \sum_{k=1}^N (e_{xn}^2(k) + e_{yn}^2(k) + e_{zn}^2(k))} \quad (56)$$

$$RMSE_J = \sqrt{\frac{1}{N} \sum_{k=1}^N (e_{q1}^2(k) + e_{q2}^2(k) + e_{q3}^2(k) + e_{q4}^2(k))} \quad (57)$$

where e_{xn}, e_{yn}, e_{zn} denote the Cartesian position tracking errors of the nacelle along the x, y, z axes, while $e_{q1}, e_{q2}, e_{q3}, e_{q4}$ are the different joint space tracking errors. N is the number of samples. As can be seen in Table 4, the RISE feedforward improvement with respect to the standard RISE controller is up to 23%.

Table 4: Controllers Performance Evaluation

	RMSE_C (cm)	Improvement Cartesian	RMSE_J (Deg)	Improvement Joint
RISE	0.0432	0 %	0.0552	0 %
RISE Feedforward	0.0330	23.51 %	0.0420	23.95 %

To reinforce the superiority of the proposed RISE feedforward with respect to standard RISE, the following performance indices are computed: The Integral Time multiplied by the Absolute Error (ITAE), the Integral Time multiplied by the Absolute Squared Error (ITASE), and the Integral of the Absolute Square Error (IASAE) [33]. The following expressions correspond to the performance indices evaluated in Cartesian space (first column), and joint space (second column.)

$$\begin{aligned} ITAE_C &= \int t. |e_{xn}(t) + e_{yn}(t) + e_{zn}(t)| dt & ITAE_J &= \int t. |e_{q1}(t) + e_{q2}(t) + e_{q3}(t) + e_{q4}(t)| dt \\ ITASE_C &= \int t. |e_{xn}(t) + e_{yn}(t) + e_{zn}(t)|^2 dt & ITASE_J &= \int t. |e_{q1}(t) + e_{q2}(t) + e_{q3}(t) + e_{q4}(t)|^2 dt \\ IASE_C &= \int |e_{xn}(t) + e_{yn}(t) + e_{zn}(t)|^2 dt & IASE_J &= \int |e_{q1}(t) + e_{q2}(t) + e_{q3}(t) + e_{q4}(t)|^2 dt \end{aligned}$$

The performance evaluation between the standard RISE and RISE feedforward controller using the aforementioned performance indices is presented in Table 5. The behaviors of each one of the performance indices are shown in Figure 12. It is appreciated in the results for Cartesian space that the RISE feedforward control improves the standard RISE control by 36% considering $ITAE_C$, 62% if $ITASE_C$ is taken into account, and 59% using $IASE_C$. The acquired enhancements in joint space are 17%, 62%, and 51% involving $ITAE_J$, $ITASE_J$, and $IASE_J$ respectively. The obtained results using these criteria confirm that the proposed RISE feedforward controller far exceeds the standard RISE controller as a result of the integration of our proposed inverse dynamic model in the feedforward part of the control scheme together with the RISE feedback controller.

Table 5: Controllers performance Evaluation Using Performance Indices

Tracking Error	Performance Index	RISE	RISE Feedforward
Cartesian Space	ITAE _C	7.899	6.555
	ITSAE _C	0.418	0.227
	IASE _C	0.041	0.020
Joint Space	ITAE _J	8.048	5.076
	ITSAE _J	0.507	0.189
	IASE _J	0.047	0.019

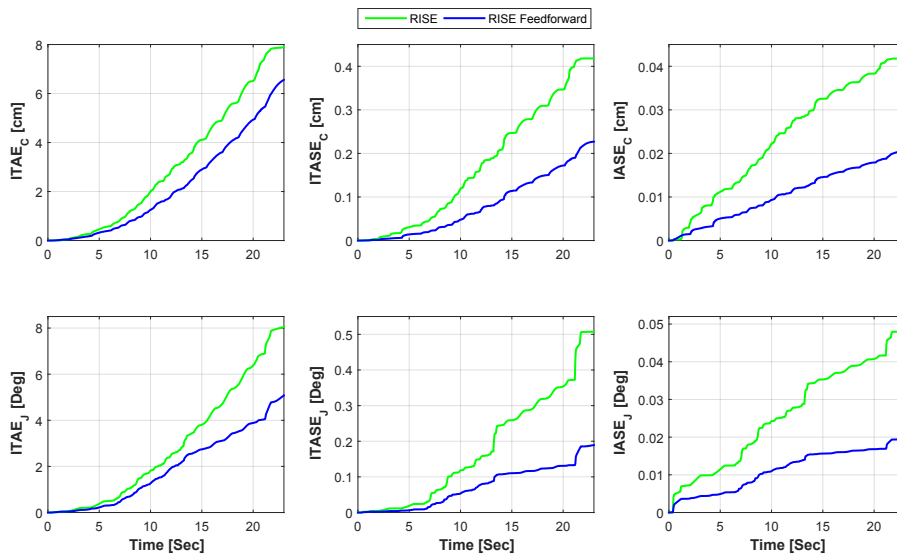


Figure 12: Evolution of the different performance indices for Cartesian and joint space with respect to time

4.3. Results discussion with respect to literature review

RISE feedforward was implemented on SPIDER4 owing to the following features: On the one hand standard RISE is a robust nonlinear feedback controller that may ensure asymptotic stability in presence of uncertain nonlinearities, unknown dynamics, external disturbances; and on the other hand the feedforward part is employed to enhance the tracking performance of the RISE controller. This control scheme is relative easy to be implemented in real-time experimentation. The previously mentioned features make to the selected control scheme suitable for wide tasks that require high precision. Comparing this control scheme with other control schemes as were mentioned in the introduction section some of them may suffer the lack of robustness or they are difficult to be implemented. For example, the linear PID controller is easy to be implemented in real-time and it can provide fairly acceptable performance on PKMs however, this performance is highly affected if the PKM is subjected to abrupt changes in the acceleration. Besides, if the operating conditions change, it is necessary to re-tune the controller gains [12].

Even if the NLPID improves the overall performance of the system compared to standard PID. The nature of its time-varying nonlinear gains may increment the energy consumption of the controller [13]. The DCAL controller shares the similarity with RISE feedforward in having a feedforward compensation term which is used to reinforce the feedback controller [19]. For DCAL the feedback term is a linear PD controller notwithstanding, as we have seen before, the RISE controller surpasses a PD control in terms of robustness. ADCR is another control technique that has been implemented successfully in PKMs [17]. This control scheme does not need to have complete knowledge about the system dynamics. The unknown dynamics are considering in a set of disturbances, involving internal perturbations, external disturbance inputs. The aim of this controller is the on-line estimation of the unknown disturbances to cancel them. To obtain the estimations ADCR employs extended state observers however, the accuracy of disturbances estimation depends on the good selection or estimation of the observer gains which is not a trivial and easy task. \mathcal{L}_1 adaptive control is a control technique that does not require knowledge dynamic knowledge and also it has been implemented satisfactorily on PKMs [15]. This control scheme control features a unique filtering technique that enables a decoupling between the estimation and the control loops. This controller is formed by several elements as state predictors, adaptation laws for estimation of unknown functions, and the decoupling low pass filter. However, the appropriate selection of the gains makes this controller quite complicated for its implementation in real-time. Finally, higher-order sliding mode controllers are robust nonlinear control strategies; they can yield finite-time stability even with unknown dynamics. These control schemes were designed in order to reduce the chattering phenomenon presented on standard sliding mode controllers [12]. Nevertheless, for PKMs as SPIDER4, the control signals are amplified greatly to move the big actuators consequently, the chattering phenomenon also may increase considerably. Therefore, the selection of appropriate control gains results in a difficult task.

5. Conclusions and future work

In this work, a new RA-PKM with five DOFs called SPIDER4 was presented. The three translational DOF are performed by the parallel structure mechanism and the two remaining degrees of freedom are produced by the independent wrist mechanism located on the nacelle. SPIDER4 is designed to perform machining operations such as milling and drilling; the main advantages in choosing a PKM as machining solution are the improvement in the mechanism stiffness, the enhanced accuracy, and the low cost compared to the conventional machine tool. Before carrying out CNC machining tasks its necessary to have knowledge about the different mathematical models. To this end, in this paper, we presented an IKM based on the formulation for delta-like PKMs, an analytical FKM, and a simplified inverse dynamic model developed for controller design purpose. Due to this PKMs with actuation redundancy may avoid some singularities; however, this redundancy may lead to the generation of some internal forces which can damage the mechanical structure. To avoid this issue the control torques were regularized using a projection operator. In order to track the desired trajectories, a RISE feedforward controller was proposed. The achieved performance using this control scheme was compared to the standard RISE controller. The obtained results showed that the proposed control scheme overcomes the standard one thanks to the inclusion of the proposed inverse dynamic model in the feedforward part. For future work, it is intended to test this platform in one real-time machining tasks and also to implement an adaptive controller in order to obtain an estimation of the unknown parameters of the system such as the friction.

Acknowledgments

This research was supported by the project Equipex Robotex. The first author was supported by The Mexican Council of Science and Technology (CONACYT); Award no. 593804 and PADES program; Award no. 2017-13-011-053.

References

- [1] H. D. Taghirad, *Parallel robots: mechanics and control*, CRC press, 2013.
- [2] J. P. Davim, *Machining: fundamentals and recent advances*, Springer Science & Business Media, 2008.
- [3] A. Overby, *CNC machining handbook: building, programming, and implementation*, McGraw-Hill, Inc., 2010.
- [4] D. Zhang, *Parallel robotic machine tools*, Springer Science & Business Media, 2009.
- [5] J.-P. Merlet, *Parallel robots*, Vol. 128, Springer Science & Business Media, 2006.
- [6] F. Park, J. W. Kim, Singularity analysis of closed kinematic chains, *Journal of mechanical design* 121 (1) (1999) 32–38.
- [7] T. Hufnagel, A. Muller, A projection method for the elimination of contradicting decentralized control forces in redundantly actuated pkm, *IEEE Transactions on Robotics* 28 (3) (2012) 723–728.
- [8] D. Chablat, P. Wenger, A new three-dof parallel mechanism: milling machine applications, arXiv.
- [9] S. Shayya, *Towards rapid and precise parallel kinematic machines*, Ph.D. thesis, Ph. D. thesis, University of Montpellier, Montpellier (2015).
- [10] F. Pierrot, T. Shibukawa, From hexa to hexam, in: *Parallel Kinematic Machines*, Springer, 1999, pp. 357–364.
- [11] J. G. Ziegler, N. B. Nichols, Optimum settings for automatic controllers, *trans. ASME* 64 (11).
- [12] M. Bennehar, G. El-Ghazaly, A. Chemori, F. Pierrot, A novel adaptive terminal sliding mode control for parallel manipulators: Design and real-time experiments, in: *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 6086–6092.
- [13] Y. X. Su, B. Y. Duan, C. H. Zheng, Nonlinear pid control of a six-dof parallel manipulator, *IEE Proceedings - Control Theory and Applications* 151 (1) (2004) 95–102.
- [14] R. Kelly, V. Santibáñez Davila, J. A. Loría Perez, *Control of robot manipulators in joint space*, Springer Science & Business Media, 2006.
- [15] M. Bennehar, A. Chemori, S. Krut, F. Pierrot, Adaptive control of parallel manipulators: Design and real-time experiments, in: *Parallel Manipulators: Design, Applications and Dynamic Analysis*, Nova Science Pub Inc, 2017.

- [16] M. Bennehar, A. Chemori, M. Bouri, L. Jenni, F. Pierrot, A new rise-based adaptive control of pkms: design, stability analysis and experiments, *International Journal of Control* 91 (3) (2018) 593–607.
- [17] L. A. Castañeda, A. Luviano-Juárez, I. Chairez, Robust trajectory tracking of a delta robot through adaptive active disturbance rejection control, *IEEE Transactions on Control Systems Technology* 23 (4) (2015) 1387–1398.
- [18] G. Sartori Natal, A. Chemori, F. Pierrot, Dual-space control of extremely fast parallel manipulators: Payload changes and the 100g experiment, *IEEE Transactions on Control Systems Technology* 23 (4) (2015) 1520–1535.
- [19] M. Bennehar, A. Chemori, F. Pierrot, A new revised desired compensation adaptive control for enhanced tracking: application to ra-pkms, *Advanced Robotics* 30 (17-18) (2016) 1199–1214.
- [20] F. Pierrot, Robots pleinement parallèles légers: conception, modélisation et commande, Ph.D. thesis, Université Montpellier 2 (1991).
- [21] A. Codourey, Contribution à la commande des robots rapides et précis, Ph.D. thesis, École polytechnique fédérale de Lausanne (1991).
- [22] R. Clavel, Conception d' un robot parallèle rapide à 4 degrés de liberté, Ph.D. thesis, École polytechnique fédérale de Lausanne (1991).
- [23] G. Sartori Natal, Control of parallel robots: towards very high accelerations, Ph.D. thesis, Montpellier 2 (2012).
- [24] D. Corbel, M. Gouttefarde, O. Company, F. Pierrot, Towards 100g with pkm. is actuation redundancy a good solution for pick-and-place?, in: *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 4675–4682.
- [25] V. Nabat, Robots parallèles à nacelle articulée: Du concept à la solution industrielle pour le pick-and-place, Ph.D. thesis, Montpellier 2 (2007).
- [26] A. Müller, T. Hufnagel, Model-based control of redundantly actuated parallel manipulators in redundant coordinates, *Robotics and Autonomous systems* 60 (4) (2012) 563–571.
- [27] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, *Robotics: modelling, planning and control*, Springer Science & Business Media, 2010.
- [28] B. Xian, D. M. Dawson, M. S. de Queiroz, J. Chen, A continuous asymptotic tracking control strategy for uncertain nonlinear systems, *IEEE Transactions on Automatic Control* 49 (7) (2004) 1206–1211.
- [29] P. M. Patre, W. MacKunis, C. Makkar, W. E. Dixon, Asymptotic tracking for systems with structured and unstructured uncertainties, *IEEE Transactions on Control Systems Technology* 16 (2) (2008) 373–379.
- [30] V. Hagenmeyer, E. Delaleau, Exact feedforward linearization based on differential flatness, *International Journal of Control* 76 (6) (2003) 537–556.
- [31] M. Spong, S. Hutchinson, M. Vidyasagar, *Robot Modeling and Control*, Wiley, 2005.

- [32] M. Bennehar, Some contributions to nonlinear adaptive control of pkms: from design to real-time experiments, Ph.D. thesis (2015).
- [33] I. Rosales-Díaz, J. R. López-Gutiérrez, A. E. Z. Suárez, S. R. Salazar, A. Osorio-Cordero, H. Aguilar-Sierra, R. Lozano, Comparison of control techniques in a weight lifting exoskeleton, *Journal of Bionic Engineering* 16 (4) (2019) 663–673.