SolidWorks/Simscape Multibody Co-Simulation of the Dynamic Model of a Mobile Manipulator System

Jean Marie Lauhic Ndong Mezui[1], Jean Brice Mandatsy Moungomo[2*], Guy Richard Kibouka[3],
Francis Okou[4], Maarouf Saad[5]

[1,2,3]*Department of Mechanical Engineering, École Normale Supérieure de l'Enseignement Technique, Libreville, Gabon*
[4]*Department of Electrical and Computer Engineering, Royal Military College, Ontario, Canada*
[5]*Department of Electrical Engineering, École de Technologie Supérieure, Montréal, Canada*

***Abstract*: This article analyzes the behaviour of the dynamics of a physical model (PM) of the mobile manipulator system (MMS). This model uses computer-aided design (CAD). The SolidWorks model is exported to Matlab using the Simscape Multibody exporter to create the PM from the XML file. The mathematical model (MM) is developed from the methods of D-H and Euler-Lagrange. The goal here is to show that the dynamic behaviour of the PM designed for SolidWorks and Matlab software approaches the behaviour of the MMS. To analyze the behaviours of the PM and the MM, a co-simulation allowing to have a virtual environment is carried out with SolidWorks, Matlab and a control structure based on the PID. The tests carried out with the proposed control structure show that the angular velocities and the control forces are similar to the two models. The similarity of the two models also proves that the MM is sufficiently precise for the design of the control laws.**

***Keywords*: Mobile-manipulator, Computer-aided design, SolidWorks, Matlab, Simscape Multibody, Co-simulation.**

1. Introduction

Mobile manipulator systems (MMS) are robots consisting of two subsystems (a manipulator arm and a mobile robot). These robots have the ability to manipulate and move objects. They have the ability to improve task performance in dynamic environments unlike single robots like a manipulator arm or mobile robot. MMS is strongly coupled and highly nonlinear when the two subsystems are coupled. To optimize the behaviour of the MMS, robust control techniques become essential. Moreover, the experimental validation of the control laws can be complicated. Indeed, not only, the design of prototypes requires a lot of time and money. But also, robots (mobile, industrial, domestic, collaborative and humanoid robots) are almost inaccessible due to high purchase costs. In addition, research laboratories increasingly require great precision in the creation of models of these robots [1]. In addition, it is almost impossible to determine the dynamic parameters (Moments of inertia, frictions, coordinates of each linked element, etc.) of the robot with accuracy, due to the uncertainty to which the system is subjected [2]. Indeed,

information on the masses and inertia of an object of interest is a basic assumption for performing precise dynamic calculations [3]. However, it is important to develop the precise model of the MMS to set up effective control structures. The Modelica language [4], [5] is used to set up prototypes of mechatronic systems. However, this language only allows a one-way translation of models from computer-aided design (CAD) to Modelica with respect to geometric and inertial parameters [6]. Several studies have used CAD for the optimization of control structures. The advantage of such an approach is to develop the model of a given system without resorting to mathematical notions [7–9]. Adams and Matlab/Simulink software was used mainly in [10] to control the movement of a 6-DOF excavator using GPS. Then, in [1] for the modelling and dynamic simulation of the Stäubli TX40 industrial robot. The Simscape language is an alternative for some research work to understand the dynamic behaviour of physical systems. Indeed, the package integrated into Matlab's Simulink toolbox [6], [11], [12] allow the use of Simscape Multibody. However, it is difficult to model complex 3D objects directly in the Simscape Multibody environment. Some research even uses Autodesk Investors software to place a prototype of the Kuka KR5 robot [13] and the Scara robot [14], [15] whose models are validated with a PID controller. This command is also used to validate the model of a UAV-type Quadcopter in SolidWorks and Matlab environments [9]. The integration between SolidWorks and Simscape Multibody eliminates the need to calculate the mathematical model and provides a mobile platform for designing the controller for the system [16]. Often, co-simulation is used to be able to compare the results obtained with the virtual prototypes with other numerical and analytical methods. This modelling method provides a virtual environment for analysis purposes and does not require the use of mathematical methods to describe the dynamics of a system [17]. Also, the use of virtual prototypes designed with SolidWorks and Simscape Multibody increases the precision of control laws and ensures the efficiency of task coordination and

*Corresponding author: jeanbricemandatsy@gmail.com

trajectory tracking [18]. Several authors such as [19–22] have applied it to serial manipulator arms and even for exoskeleton robots. Rehabilitation [23–25] for the development of robust control laws. Other research directs this method towards mobile robots [26–28] to implement trajectory planning. Authors like [29] have made use of this modelling technique to identify parameters of online mobile robot dynamics. We notice that this modelling method is widely applied on manipulator arms and mobile robots. Moreover, very little research uses this method on MMS to experiment with control laws. Considering the fact that the subsystems (manipulator arm and mobile robot) are coupled, this article uses co-simulation to analyze the dynamic behaviour of the virtual prototype of a 4-DOF MMS. Computer-aided design (CAD) is used to deduce the physical model of the MMS with SolidWorks and Simscape Multibody software. Thus, the Simscape Multibody library which is part of the Matlab toolbox will make it possible to connect the PM to the command structure proposed in this article. The contributions of this work are around the following points:

- Develop the PM for which the two MMS subsystems are coupled using SolidWorks. Then, perform a co-simulation with Matlab and Simscape Multibody software;
- Find the kinematic and dynamic models of the MMS and set up a control structure based on the PID to analyze the behaviours of the PM and MM.

The rest of the article is structured as follows: Section II presents the mathematical models (kinematics and dynamics) of the MMS at 4DDL. It highlights the virtual prototype of this robot using SolidWorks and Matlab/Simscape Multibody software. The analysis of the results obtained is presented in section III. Finally, a conclusion is given in section IV.

2. Modeling of the Mobile Manipulator

To study the planning of MMS trajectories, the implementation of a model is necessary. Modelling this robot is an essential step in setting up the control structures. In this section, a virtual prototype of the MMS is first developed using SolidWorks and Matlab. This model makes the parameters (mass, friction and moments of inertia) accessible. Then, the mathematical model (MM) is designed to work out the control structure. Moreover, the comparison of the two models is made to justify that the mathematical model is sufficiently precise to propose a control law.

A. Co-simulation of MMS with Solidworks/Matlab

To develop a dynamic model of MMS, the application of virtual reality is an alternative that gives us the possibility to experiment on the analysis of MMS behaviour. Indeed, a robot can be tested without any hardware involvement by using a simulator and virtual robot models (Azimi et al., 2017) [30]. In this section the SolidWorks software is first used to design the physical model (PM) of the MMS shown in Fig.1. This model allows to introduce a control system and perform any motion analysis. Indeed, the PM was modelled on the SolidWorks software to facilitate the technological and computer relationship between the CAD and the Matlab environment.

Sketches (Line, Corner Rectangle, offset the surface, convert entities… etc.) and functions (Extruded Boss/Base, Extruded Cut, Reference Geometry, Revolved Boss/Base…etc.) were necessary to create the part models (the motors, the wheels, the links that connect the joints of the arm, the effector, the frame). To define the geometry of the MMS, the assembly of the MMS is developed using the local coordinates associated with each part in 3D whose links are ensured using standard, advanced and mechanical constraints.
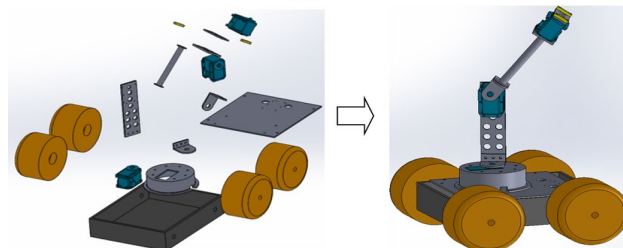

Fig. 1. SolidWorks model of the Manipulator-Mobile 4-DOF system

In order robotic systems, experimenting with control strategies on prototypes is a real challenge. The use of CAD makes it possible to carry out analyzes of the mechanical models of 3D systems. However, it is difficult to include control structures in these studies. To ensure co-simulation with a control structure, the PM shown in Fig. 1 is exported to Matlab. The export of the PM to Matlab takes into account all the bonds, dimensions and material properties of the MMS. Fig. 2 highlights the process of exporting the SolidWorks model to Matlab/Simscape Multibody. Indeed, the assembly of the MMS under the extension. sldam is exported in .xlm format to finally obtain prototype in .slx or .mdl format.
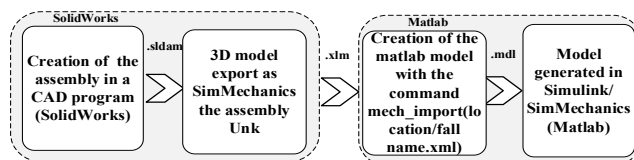

Fig. 2. Algorithm of generation of an exported model to Simscape Multibody/Matlab. Taken from: [9]

The model of the MMS that we obtain is illustrated in Fig. 3a where the kinematic and dynamic blocks are highlighted. The creation of the kinematic chain that defies movement between the mechanical parts of the MMS is done using four (4) rotary joints. Solid transformations and frames of reference are performed with rotation transform blocks. The flat joint is used to ensure the coupling of the mobile robot and the manipulator arm. The kinematics of the mobile platform are illustrated in Fig. 3b. In the dynamic block, the command and speed signals are well illustrated in Fig. 3c. This makes it possible to include the control system which allows to analyze and optimize the behaviour of the MMS. If we want to modify the parameters of the MMS such as: frictions, masses, moments of inertia, the MS model must be re-exported in Matlab. The explanations of the export procedure of the MS generating an XML file and the means of its import into the Matlab/Simulink environment which forms the Simscape Multibody model [9], [31], [32].
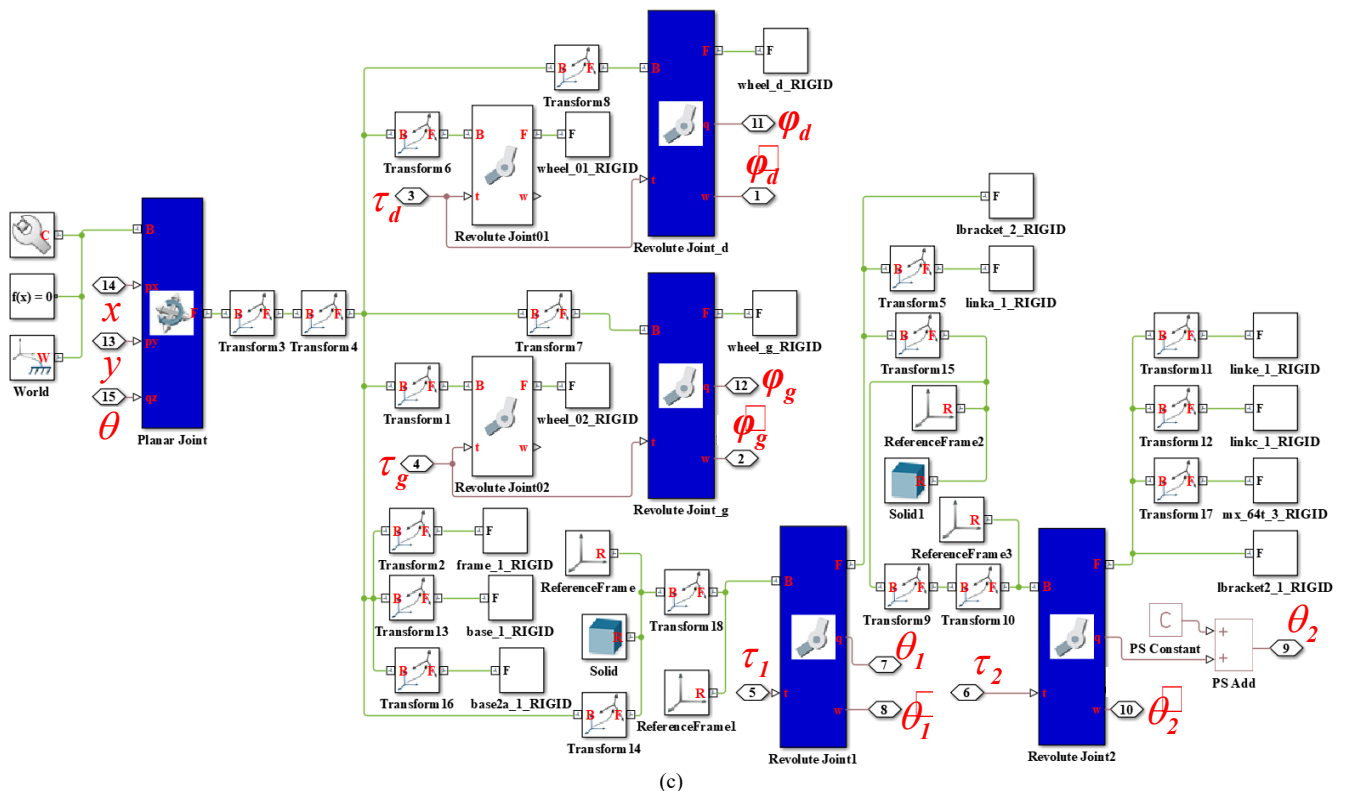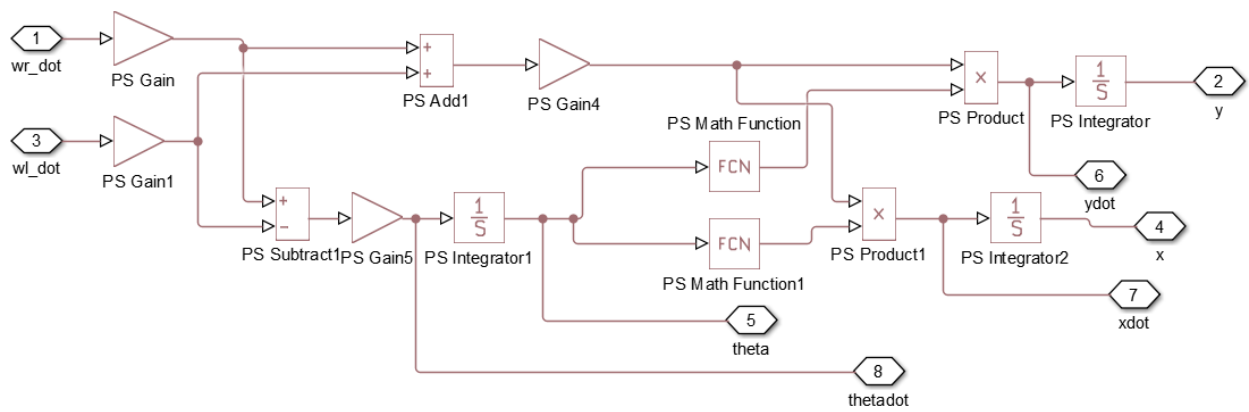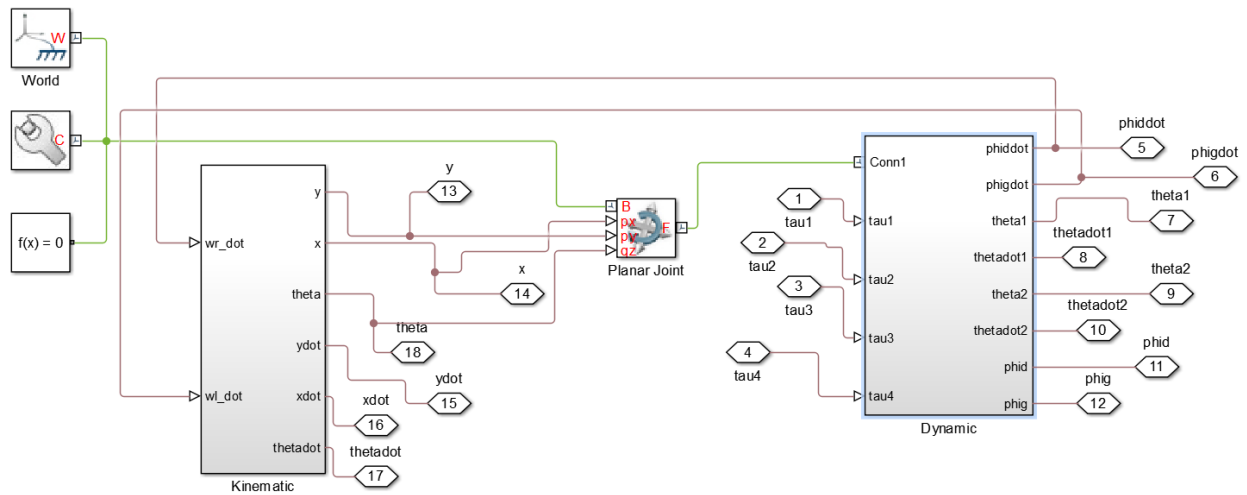
Fig. 3. Diagram of the 4-DOF MMS under Matlab/ Simscape Multibody (a), kinematic (b) and dynamic (c) model

When exporting the SolidWorks model to Matlab, all mass geometry of the MMS is taken into account. Furthermore, to avoid conflicts between the MS model and the analytical model as well as controller parameter issues in the hardware implementation [30], the model presented in the following subsection is designed using the exact parameters of the virtual prototype.

B. MMS Kinematic Model

The MMS illustrated in Fig. 4 is designed from the SolidWorks model shown in Fig. 1. It has a 2-dll mobile base under non-holonomic constraints and a 2-DOF manipulator arm. Note that $I=(O,\vec{I_1},\vec{I_2},\vec{I_3})$ is the absolute reference frame and $J=(O_\varphi,\vec{J_1},\vec{J_2},\vec{J_3})$ is the relative reference frame related to the robot. When the subsystems are paired, the MMS has 4-DOF. This coupling is established by an angle $(\theta+\Theta_1)$ where θ is the angle of curvature of the mobile robot and $\Theta1$ is the angular angle of the first articulation of the arm.

Note that the $\mathbf{R_{i-1}}=(O_{i-1},\vec{x}_{i-1},\vec{y}_{i-1},\vec{z}_{i-1})$ are relative reference frames respectively, of the joints {i} of the arm where i=1,2. Then, $\mathbf{R_e}=(O_e,\vec{x}_e,\vec{y}_e,\vec{z}_e)$ is the reference frame linked to the effector. Note also that the l_i and l_{ci} are distances between joints and distances between joints and their centers of mass. The MMS considered in this article has a moving base under non-holonomic constraints. Formulated in the works [33], [34], the expression of this constraint is written:

$$-\dot{x}sin(\theta)+\dot{y}cos(\theta)+l_c\dot{\theta}=0 \qquad (1)$$

The kinematics, which is taken into account the nonholonomic constraints can thus be rewritten in the form:

$$\dot{P}_m=\begin{bmatrix} \cos(\theta) & -l_c\sin(\theta) \\ \sin(\theta) & l_c\cos(\theta) \\ 0 & 1 \end{bmatrix}\begin{bmatrix} \dfrac{r}{2} & \dfrac{r}{2} \\ \dfrac{r}{2a} & \dfrac{-r}{2a} \end{bmatrix}\dot{q}_m \qquad (2)$$

Note that $\dot{\mathbf{P}}_m=[\dot{x},\dot{y},\dot{\theta}]^T$ is the speed vector of the mobile in the plane and $\dot{\mathbf{q}}_m=[\dot{\phi}_d,\dot{\phi}_g]^T$ is the vector of generalized coordinates

formed by the variables of the driving wheels. Where l_c is the distance between the center of the axle $\mathbf{O}_\varphi$ and $\mathbf{O}_c$ is the coupling point of the MMS subsystems. Also, $\mathbf{r}$ is the radius of the wheels and $\mathbf{a}$ is the distance between the wheel and the center of the axle. Table 1 represents the D-H parameters of the manipulator arm in the $\mathbf{R_0}$ reference frame.

Table 1
Modified arm D-H parameters

i	α_{i-1}	a_{i-1}	d_i	Θ_i
1	0	0	0	Θ_1
2	$\pi/2$	l_1	0	$\Theta_2+\pi/2$

Given the general homogeneous transformation matrix [35], [2], [36] and Table 1, by setting: $\cos(\theta_i)=c_{\theta i}$ and et $\sin(\theta_i)=s_{\theta i}$, the following matrices are derived:

$$^0_1T=\begin{bmatrix} c_{\Theta_1} & -s_{\Theta_1} & 0 & 0 \\ s_{\Theta_1} & c_{\Theta_1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^1_2T=\begin{bmatrix} -s_{\Theta_2} & -c_{\Theta_2} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ c_{\Theta_2} & -s_{\Theta_2} & 1 & -l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (3)$$

The geometry model which makes it possible to describe the behaviour of the manipulator arm is obtained by setting: $^{O_c}_{O_e}T=^{O_c}_1T.^1_2T$, to reach the position of the tool, we base ourselves on Table 1 and Fig. 4. The passage matrix between the {$\mathbf{R_e}$} and {$\mathbf{I}$} markers is written:

$$^{O_c}_{O_e}T=\left[\begin{array}{c|c} ^{O_c}_{O_e}R(q_e) & ^{O_c}P_e(q_e) \\ \hline 0_{1\times3} & 1 \end{array}\right] \qquad (4)$$

where

$$^{O_c}_{O_e}R=\begin{bmatrix} -c_{\Theta_1}c_{\Theta_2} & -c_{\Theta_1}s_{\Theta_2} & s_{\Theta_1} \\ -s_{\Theta_1}c_{\Theta_2} & -s_{\Theta_1}s_{\Theta_2} & c_{\Theta_1} \\ c_{\Theta_2} & -s_{\Theta_2} & 0 \end{bmatrix}; \quad ^{O_c}P_e=\begin{bmatrix} -c_{\Theta_1}s_{\Theta_2}l_2 \\ -s_{\Theta_1}s_{\Theta_2}l_2 \\ L_1+c_{\Theta_1}s_{\Theta_2}l_2 \end{bmatrix}$$

The passage matrix described by Eq. (4) makes it possible to express the effector linked to the {$\mathbf{R_0}$} frame of the MMS in the {$\mathbf{I}$} frame.
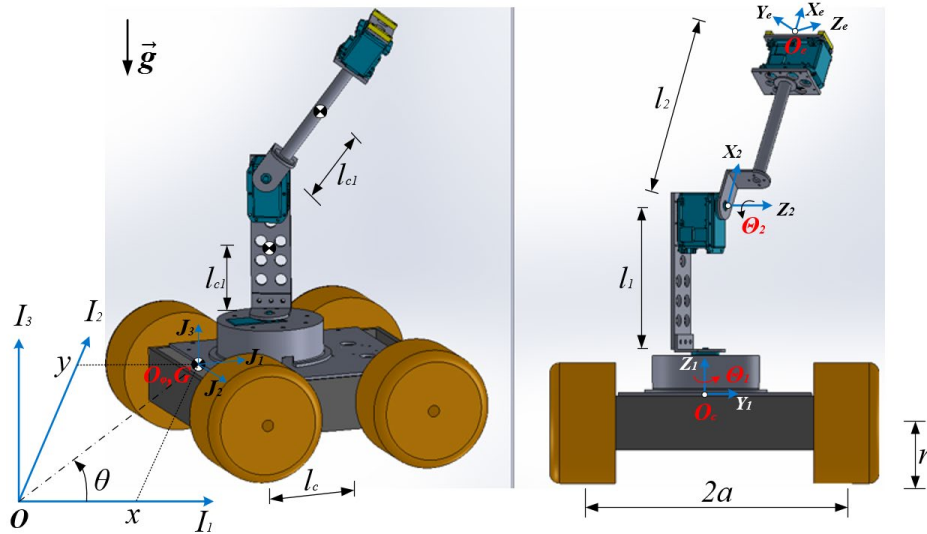

Fig. 4. Identification of the mobile manipulator system

$$_{o_e}^{O}T = \begin{bmatrix} \boldsymbol{R}(\theta) & ^O P_{O_c} \\ \hline 0_{1\times3} & 1 \end{bmatrix} \cdot \begin{bmatrix} ^{O_c}_{o_e}R(q_e) & ^{O_c}P_2(q_e) \\ \hline 0_{1\times3} & 1 \end{bmatrix} \qquad (5)$$

where

$$\boldsymbol{R}(\theta) = \begin{bmatrix} c_\theta & -s_\theta & 0 \\ s_\theta & c_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad ^O\boldsymbol{P}_{O_c} = \begin{bmatrix} x + l_c c_\theta \\ y + l_c s_\theta \\ 0 \end{bmatrix}$$

When the MMS is coupled, it is possible to define the geometric model of the arm for the development of Eq. (5). therefore:

$$^O\boldsymbol{P}_e = \begin{bmatrix} -c_{\Theta_1} s_{\Theta_2} c_\theta l_2 + s_{\Theta_1} s_{\Theta_2} s_\theta l_2 + x + l_c c_\theta \\ -c_{\Theta_1} s_{\Theta_2} s_\theta l_2 - s_{\Theta_1} s_{\Theta_2} s_\theta l_2 + y + l_c s_\theta \\ l_1 + c_{\Theta_1} s_{\Theta_2} l_2 \end{bmatrix} \qquad (6)$$

The kinematic model that takes into account the coupling effect in the {**I**} frame is expressed as in the reference [37]:

$$\dot{\boldsymbol{P}}_e = \boldsymbol{R}(\theta) K_m \dot{\boldsymbol{q}}_m + \boldsymbol{R}(\theta) J_e \Sigma \dot{\boldsymbol{q}}_m + \boldsymbol{R}(\theta) J_e \dot{\boldsymbol{q}}_e \qquad (7)$$

where

$$J_e = \frac{\partial}{\partial \boldsymbol{q}_e}\left(^{O_c}P_{O_e}(q_e)\right), \quad \Sigma^T = \begin{bmatrix} \overline{\Sigma} & O_{2\times2} \end{bmatrix}$$

With $\overline{\Sigma} = \left[\dfrac{r}{2a}, \dfrac{r}{2a}\right]$. Note that $\dot{\boldsymbol{P}}_e = [\dot{p}_x, \dot{p}_y, \dot{p}_z]^T$ is the velocity vector of the effector in Cartesian space. Also, $\dot{\boldsymbol{q}}_e = [\dot{\Theta}_1, \dot{\Theta}_2]^T$ represents the vector of generalized coordinates formed by the variables of the joints of the manipulator arm. The coupled kinematic model of the MMS can also be written:

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{R}(\theta) J_e & \boldsymbol{R}(\theta)\left(K_m + J_e \Sigma\right) \\ O_{3\times2} & \boldsymbol{R}(\theta) K_m \end{bmatrix} \cdot \boldsymbol{q} \qquad (8)$$

Note that $\boldsymbol{P} = [\boldsymbol{P_e}, \boldsymbol{P_m}]^T$ is the MMS position vector and $\boldsymbol{q} = [\boldsymbol{q_m}, \boldsymbol{q_e}]^T$ is the generalized coordinate vector formed by the wheel and joint variables.

C. MMS Inverse Kinematics

This section presents the inverse kinematics to find the angular positions of the MMS from its Cartesian coordinates. We start by determining the angles $\boldsymbol{q}_e$ of the arm, for this we solve Eq.(6) in which the angles $\boldsymbol{q}_e$ and $\boldsymbol{q}_m$ are unknown and the position $\boldsymbol{P}_e$ is known. Solving Eq. (6) makes it possible to find the angular positions of the arm as a function of the Cartesian positions of the tool. Thereby:

$$\Theta_2 = \pm A\cos\left(\frac{p_z - l_1}{l_2}\right) \qquad (9)$$

$$\Theta_1 = A\tan2(\vartheta_2, \vartheta_1) - A\tan2(-l_2 s_2 c_\theta, -l_2 s_2 s_\theta) \qquad (10)$$

where

$$\vartheta_1 = P_y - y - l_c s_\theta, \quad \vartheta_2 = P_y - x - l_c c_\theta$$

It is clear, because of the coupling effect, Eqs. (9), (10) depends on the steering angle of the mobile robot. Furthermore,

this angle is a function of the angular positions of the wheels. To define these positions, we integrate the velocities defined in [38], [39].

$$\dot{\boldsymbol{q}}_m = \begin{bmatrix} \dfrac{r}{2} & \dfrac{r}{2} \\ \dfrac{r}{2a} & \dfrac{-r}{2a} \end{bmatrix}^{-1} \begin{bmatrix} k_x e_x + v_r \cos(e_\theta) \\ v_r k_\theta \sin(e_\theta) + v_r k_y e_y + \omega_r \end{bmatrix} \qquad (11)$$

with

$$\begin{bmatrix} e_x, e_y, e_\theta \end{bmatrix}^T = [x_d - x, y_d - y, \theta_d - \theta]^T$$

Note that $k_x > 0$, $k_y > 0$ and $k_\theta > 0$ are gains of Eq. (11). Then, v_r and ω_r are referencing linear and angular velocities. Also, x_d, y_d and θ_d are referencing positions. The angular positions of the mobile robot are obtained by integrating Eq. (11). Therefore:

$$\boldsymbol{q}_m = \int \dot{\boldsymbol{q}}_m \cdot dt \qquad (12)$$

D. Dynamic Model

The dynamic modelling of MMS systems still remains a complex problem in control, according to the coupling or coupling hypotheses. Taking into account the coupling effect, the dynamic model of the MMS system is developed by applying the Euler-Lagrange method [37,40,41], Therefore

$$\boldsymbol{M}(q)\ddot{\boldsymbol{q}} + \boldsymbol{V}(q, \dot{\boldsymbol{q}}) + G(q) + F(\dot{\boldsymbol{q}}) + \tau_d = \tau \qquad (13)$$

Note that τ_d and $\tau \in \mathbf{R}^{4\times1}$ are respectively the torque applied to the joints and the disturbance torque of the MMS. $\mathbf{G}(q) \in \mathbf{R}^{4\times1}$ is the gravity vector. Note also that $\mathbf{F} \in \mathbf{R}^{4\times4}$ is the frictional force. The mass matrix $\mathbf{M}(q) \in \mathbf{R}^{4\times4}$ and the vector of centrifugal and Coriolis forces $\mathbf{V}(q, \dot{q}) \in \mathbf{R}^{4\times1}$. The following properties are necessary for the study of the stability of MMS. Also, the satisfaction of these properties confirms the realization of Eq.(13).

Property 1: *The mass matrix* $\boldsymbol{M}(q)$ *is symmetric and positive definite for all* $q \in \boldsymbol{R}^{4\times1}$ *[35,42].*

Property 2: *The matrix* $\boldsymbol{V_m}$ *is always chosen so that* $\dot{\boldsymbol{M}}(q)-2\boldsymbol{V_m}(q,\dot{q})$ *is antisymmetric such that* $S^T[\dot{\boldsymbol{M}}(q)-2\boldsymbol{V_m}(q,\dot{q})]S=0$, *for all S and* $q \in \boldsymbol{R}^{4\times1}$ *[43,44].*

3. Results and Discussions

When comparing the behaviour of the physical model and the mathematical model, it is important to obtain similar results [6]. To achieve this similarity, the choice of an efficient control is necessary. The co-simulation of 4-DOF MMS is performed by applying the control structure based on PID control. This allowed us to show the control law that makes the mathematical model sufficiently accurate. The MMS parameters are shown in Table 2.

To analyze the behaviour of the MS, the command structure proposed in Fig. 5 is used. Indeed, the MM is replaced by the prototype of the MMS given in Fig. 1 to test this command structure. We show that the command structure used in the co-simulation, allows to show the similarity of the two models of the MMS presented in this article. Better convergence
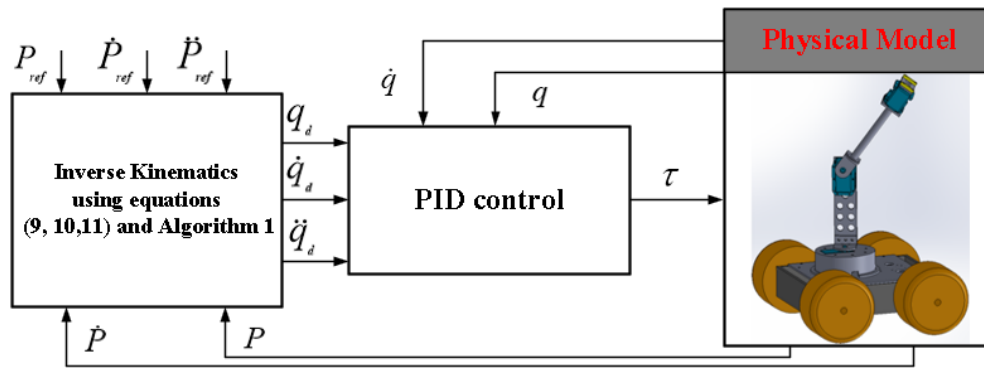
Fig. 5. Mobile manipulator system control structure

performances of speeds and control forces are analyzed for this purpose.

Table 2
Parameters of the MMS to 4DOF

Physical variables	Values	Units
I_{z1} and I_{z2}	0.05 and 0,1	$K_g.m^2$
I_r, I_ω and I_φ	1.411, 972,6e-6 and 1407,76e-6	$K_g.m^2$
m_1 and m_2	27,04e-3 and 38,23e-3	Kg
m_r and m_ω	3.8 and 0,76	Kg
l_1 and l_2	145e-3 and 206e-3	m
l_{c1} and l_{c2}	$l_1/2$ and $l_2/2$	m
f_1, f_2, f_d and f_g	0,1, 0,1, 0,25 and 0,1	
a, r and l_c	0,133, 61e-3 and 10,94 e-3	m

The test that is proposed in this part of the article aims to demonstrate from a powerful command that the virtual prototype and the mathematical model have a similar behaviour even in the presence of strong disturbances. Note that during the test, the dynamic parameters are known when the virtual prototype is available. The execution of the control structure required the choice of gains proposed by Table 2. The calculation of the PID gains used for this simulation is detailed in [36].

Table 3
Gain values

Gains	Values
Kp	19.98diag([1,1,1,1])
Ki	90.06diag([1,1,1,1])
Kd	109.04diag([1,1,1,1])
K_x, K_y and K_θ	0.4, 0.6 et 0.2

Considering the MMS to be coupled makes this robot highly nonlinear. In addition to nonlinearities, singularities are generated in the workspace. These singularities are taken into account by Algorithm 1.

Exporting the SolidWorks model to the Matlab/Simscape Multibody environment allowed us to obtain the co-simulation of the command structure illustrated in Fig. 6. This presentation of the co-simulation thus allowed us to visualize the behaviour of the MMS during the proposed control structure simulation.

The ultimate goal of the test is to show the similarity between the behaviours of the PM and MM. To achieve this, we first compared the angular velocities of the four MMS joints. We can see that in Fig. 7 eight angular velocities are similar. Note that the first speed of the MM arm struggles to follow that of the PM at the start of the simulation. But the similarity of these

speeds is very quickly caught up in a steady state. Secondly, we verified this similarity by comparing the MMS command efforts. We can also notice that as for the speeds, the MM control forces converge towards the PM control forces as shown in Fig. 8.

Algorithm 1 Movements under constraint of the MM.

```
1:  function Q_FINAL(q[4])
2:      q1 ← q[0]; q2 ← q[1]; q3 ← q[2]; q4 ← q[3]
3:      q3 ← q3 mod 2π
4:      q4 ← q4 mod 2π
5:          if q3 > π then
6:              q3 = q3 − 2π
7:          else if q3 < −π then
8:              q3 = q3 + 2π
9:          end if
10:         if q4 > π then
11:             q4 = q4 − 2π
12:         else if q4 < −π then
13:             q4 = q4 + 2π
14:         end if
15:         return {q1;q2;q3;q4}
16: end function
```
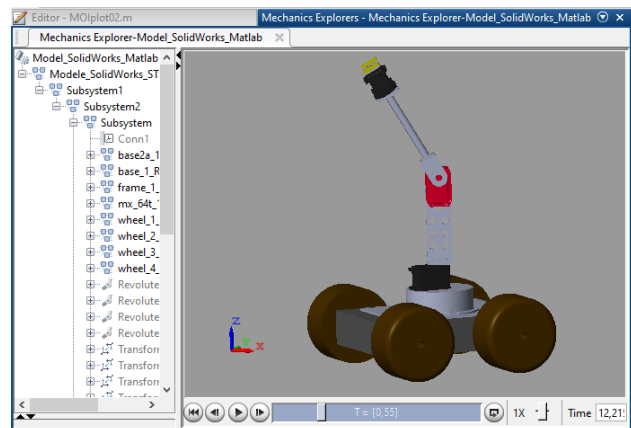

Fig. 6. Visualisation du modèle SolidWorks dans Matlab/Simscape Multibody

It can be observed that with the coupling effect of the MMS subsystems, the similarity of the models is guaranteed when the co-simulation is done with a PID control. In the model designed

from SolidWorks, the friction forces of the angled joints of the MMS are taken into account. Which makes this model more realistic. The fact of having the speeds and the efforts of the MM controller, which converge towards those of the PM prove that the modelling developed with the analytical methods is sufficiently precise. Indeed, this approach makes it possible to show that the MM reflects the dynamic behaviour of the MP. This confirms that MM is well suited to the design of control laws and the analysis of co-simulation developed with the proposed control structure.
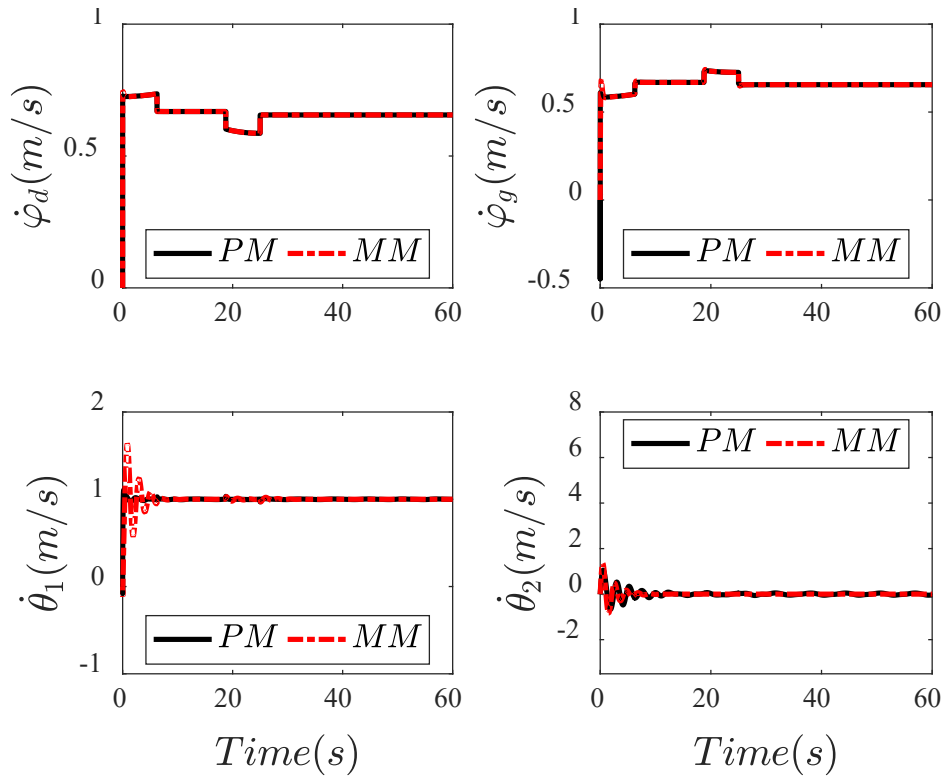
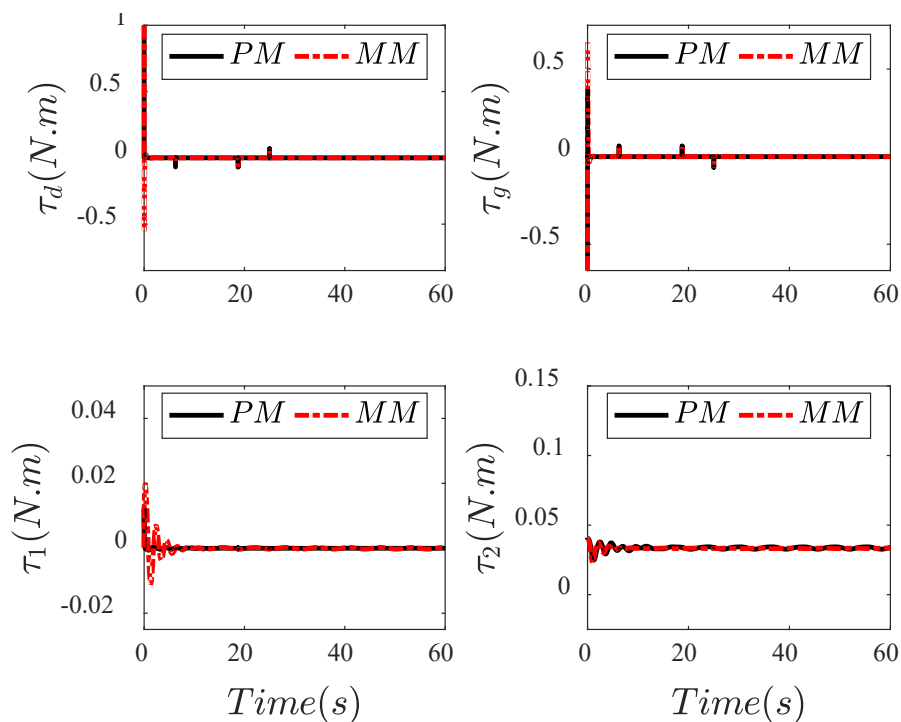Fig. 7. Comparison of the Angular Velocities of PM and MM

Fig. 8. Comparison of the PM and MM control efforts

4. Conclusion

In this article, a physical model of the 4-DOF MMS developed using SolidWorks software is presented. Then, the co-simulation is carried out with the model obtained with SolidWorks and Simscape Multibody to create a virtual environment. Indeed, the physical model of the MMS is generated by exporting the SolidWorks model from the XTL file. This virtual prototype is developed taking into account the fact that the MMS subsystems is strongly coupled. This type of model not only makes it possible to deduce the physical parameters (moments of inertia, friction, masses, etc.) of the MMS, but also to produce visualizations of the real movements of the robot. But also to introduce a command structure that allows to analyze and visualize the dynamic behaviour of the MMS. To this end, a mathematical model based on the method of D-H and Euler Lagrange is calculated to be able to design control laws. To verify the similarity between the physical and mathematical models, the analysis of the test comparing the speeds and the control efforts of the PM and MM is carried out a control structure based on the PID control. These results show that the angular velocities of the MM converge towards those of the PM. The same observation is also made with the control forces. Moreover, as the analysis has shown, the convergence errors of speed $\dot{\theta}_1$ and control force τ_1 increase to less than 5 seconds. This error is justified by the uncertainties of the MM. But, after this time, we observe a similarity of these signals in steady state. The rest of the speed and effort signals better show the similarity between MM and PM. Given the strong nonlinearities and the coupling effect of MMS, we observed that there is a good similarity between the two models presented in this article. This also demonstrates that the MM is sufficiently precise to elaborate the control laws of the type of model presented in this article.

References

[1]   Cheraghpour, F., Vaezi, M., Jazeh, H. E. S., & Moosavian, S. A. A. (2011, April). Dynamic modelling and kinematic simulation of Stäubli© TX40 robot using Matlab/Adams co-simulation. In *2011 IEEE International Conference on Mechatronics* (pp. 386-391). IEEE.

[2]   Brahmi, B., Saad, M., Luna, C. O., Archambault, P. S., & Rahman, M. H. (2017, June). Sliding mode control of an exoskeleton robot based on time delay estimation. In *2017 International Conference on Virtual Rehabilitation (ICVR)* (pp. 1-2). IEEE.

[3]   ŞAFAK, K. K. (2014). Dynamics, stability, and actuation methods for powered compass gait walkers. Turkish Journal of Electrical Engineering and Computer Sciences, 22(6), 1611-1624.

[4]   Mattsson, S. E., & Elmqvist, H. (1998, April). An overview of the modelling language Modelica. In *Eurosim'98 Simulation Congress*. Helsinki.

[5]   Ferretti, G., Magnani, G., & Rocco, P. (2004). Virtual prototyping of mechatronic systems. *Annual Reviews in Control*, 28(2), 193-206.

[6]   Zatopek, J., Machado, J., & Sousa, J. (2018). Dynamic simulation of the CAD model in SimMechanics with multiple uses. *Turkish Journal of Electrical Engineering & Computer Sciences*, 26(3), 1278-1290.

[7]   Lu, P., & Geng, Q. (2011, August). Real-time simulation system for UAV based on Matlab/Simulink. In *2011 IEEE 2nd international conference on computing, control and industrial engineering* (Vol. 1, pp. 399-404). IEEE.

[8]   Frankovský, P., Hroncová, D., Delyová, I., & Virgala, I. (2013). Modelling of dynamic systems in simulation environment Matlab/Simulink–SimMechanics. *American Journal of Mechanical Engineering*, 1(7), 282-288.

[9]   Jatsun, S., Lushnikov, B., Emelyanova, O., & Leon, A. S. M. (2021). Synthesis of SimMechanics Model of Quadcopter Using SolidWorks CAD Translator Function. In *Proceedings of 15th International Conference on Electromechanics and Robotics" Zavalishin's Readings"* (pp. 125-137). Springer, Singapore.

[10] Makkonen, T., Nevala, K., & Heikkilä, R. (2006). A 3D model based control of an excavator. *Automation in Construction*, 15(5), 571-577.

[11] Ryadchikov, I., Sechenev, S., Mikhalkov, N., Biryuk, A., Svidlov, A., Gusev, A., ... & Nikulchev, E. (2020). Feedback control with equilibrium revision for cmg-actuated inverted pendulum. In *Proceedings of 14th International Conference on Electromechanics and Robotics "Zavalishin's Readings"* (pp. 431-440). Springer, Singapore.

[12] Kathpal, A., & Singla, A. (2017, January). SimMechanics™ based modelling, simulation and real-time control of Rotary Inverted Pendulum. In *2017 11th International Conference on Intelligent Systems and Control (ISCO)* (pp. 166-172). IEEE.

[13] Udai, A. D., Rajeevlochana, C. G., & Saha, S. K. (2011). Dynamic simulation of a Kuka KR5 industrial robot using MATLAB SimMechanics. In *15th National conference on machines and mechanisms* (Vol. 96, pp. 1-8).

[14] Ibrahim, B. S. K. K., & Zargoun, A. M. (2014). Modelling and control of SCARA manipulator. *Procedia Computer Science*, 42, 106-113.

[15] Zhuo, Z., & Cheng, Z. (2021, June). SCARA Modeling and Simulation based on SimMechanics and Solidworks. In *2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)* (Vol. 4, pp. 1559-1563). IEEE.

[16] Alkamachi, A. (2020). Integrated SolidWorks and Simscape platform for the design and control of an inverted pendulum system. *Journal of Electrical Engineering*, 71(2), 122-126.

[17] Sosa-Méndez, D., Lugo-González, E., Arias-Montiel, M., & Garcia-Garcia, R. A. (2017). ADAMS-MATLAB co-simulation for kinematics, dynamics, and control of the Stewart–Gough platform. *International Journal of Advanced Robotic Systems*, 14(4), 1729881417719824.

[18] Matoui, F., Boussaid, B., & Abdelkrim, M. N. (2017, January). Path planning of two robots in dynamic workspace based on potential field. In *2017 International Conference on Advanced Systems and Electric Technologies (IC_ASET)* (pp. 267-272). IEEE.

[19] Brezina, T., Hadas, Z., & Vetiska, J. (2011, June). Using of Co-simulation Adams-Simulink for development of mechatronic systems. In *14th International Conference Mechatronika* (pp. 59-64). IEEE.

[20] Baran, D., & Lisowski, W. (2013). Numerical simulation and co-simulation in analysis of manipulators' dynamics. *Mechanics and Control*, 32(4).

[21] [21] Sudharsan, J., & Karunamoorthy, L. (2016). Path planning and co-simulation control of 8 DOF anthropomorphic robotic arm. *International journal of simulation modelling*, 15(2), 302-312.

[22] Bhatia, V., Ram, R. G., Kalaichelvi, V., & Karthikeyan, R. (2015, December). Application of Model Predictive Controller for 2-DOF robot manipulator. In *2015 10th International Symposium on Mechatronics and its Applications (ISMA)* (pp. 1-5). IEEE.

[23] Li, Y., Wang, X., Xu, P., Zheng, D., Liu, W., Wang, Y., & Qiao, H. (2013). SolidWorks/SimMechanics-based lower extremity exoskeleton modelling procedure for rehabilitation. In *World Congress on Medical Physics and Biomedical Engineering May 26-31, 2012, Beijing, China* (pp. 2058-2061). Springer, Berlin, Heidelberg.

[24] Chen, Y., Liu, J., Wang, H., Pan, Z., & Han, S. (2019, June). Model-free based adaptive RBF neural network control for a rehabilitation exoskeleton. In *2019 Chinese Control and Decision Conference (CCDC)* (pp. 4208-4213). IEEE.

[25] Han, S., Wang, H., Tian, Y., & Christov, N. (2020). Time-delay estimation based computed torque control with robust adaptive RBF neural network compensator for a rehabilitation exoskeleton. *ISA transactions*, 97, 171-181.

[26] Angel, L., Hernandez, C., & Diaz-Quintero, C. (2013, July). Modelling, simulation and control of a differential steering type mobile robot. In *Proceedings of the 32nd Chinese Control Conference* (pp. 8757-8762). IEEE.

[27] Christiansen, M. P., Larsen, P. G., & Jørgensen, R. N. (2015). Robotic design choice overview using co-simulation and design space exploration. *Robotics*, 4(4), 398-420.

[28] Matoui, F., Boussaid, B., & Abdelkrim, M. N. (2019). Distributed path planning of a multi-robot system based on the neighborhood artificial potential field approach. *Simulation*, 95(7), 637-657.

[29] Hoang, N. B., & Kang, H. J. (2015). Observer-based dynamic parameter identification for wheeled mobile robots. *International Journal of Precision Engineering and Manufacturing*, *16*(6), 1085-1093.

[30] Azimi, M. S., Shukri, Z. A., & Zaharuddin, M. (2017). Virtual reality based mobile robot navigation in greenhouse environment. *Advances in Animal Biosciences*, *8*(2), 854-859.

[31] Cekus, D., Posiadała, B., & Warys, P. (2014). Integration of modelling in SolidWorks and Matlab/Simulink environments. *Archive of Mechanical Engineering*, *61*(1).

[32] Benotsmane, R., Dudás, L., & Kovács, G. (2020). simulation and trajectory optimization of collaborating robots by application of solidworks and matlab software in industry 4.0. *academic journal of manufacturing engineering*, 18(4).

[33] Seraji, H. (1998). A unified approach to motion control of mobile manipulators. *The International Journal of Robotics Research*, *17*(2), 107-118.

[34] Peng, J., Yu, J., & Wang, J. (2014). Robust adaptive tracking control for nonholonomic mobile manipulator with uncertainties. *ISA transactions*, *53*(4), 1035-1043.

[35] Craig, J. J. (2005). Introduction to robotics: mechanics and control. Pearson Educacion.

[36] Mezui, J. M. L. N., Nganga-Kouya, D., Maarouf, S. A. A. D., & Okou, A. F. (2021). Modélisation, commande et simulation d'un robot industriel. Sciences Appliquées et de l'Ingénieur, 2(1), 16-23.

[37] Tzafestas, S. G. (2013). *Introduction to mobile robot control*. Elsevier.

[38] Maalouf, E., Saad, M., & Saliah, H. (2006). A higher level path tracking controller for a four-wheel differentially steered mobile robot. *Robotics and autonomous systems, 54*(1), 23-33.

[39] Mezui, J.M.L.N., Nganga-Kouya, D., Okou, A.F., Saad, M. and Hernandez, B. (2019) 'A novel control structure for a pioneer mobile robot: simulation and practical implementation', *Int. J. Modelling, Identification and Control*, Vol. 33, No. 2, pp.108–119.

[40] Li, Z., & Ge, S. S. (2013). Fundamentals in modelling and control of mobile manipulators (Vol. 49). crc press.

[41] Brahmi, A., Saad, M., Gauthier, G., Zhu, W. H., & Ghommam, J. (2019). Adaptive backstepping control of multi-mobile manipulators handling a rigid object in coordination. *International Journal of Modelling, Identification and Control*, *31*(2), 169-181.

[42] Brahmi, B., M. Saad, C. Ochoa-Luna, M. H. Rahman, et S. Di Gennaro. 2018. « Adaptive control of upper limb exoskeleton robot based on a new modified Function Approximation Technique (FAT) ». *In 2018 European Control Conference (ECC)*. pp. 608–613. IEEE.

[43] Lewis, F. L., Dawson, D. M., & Abdallah, C. T. (2003). *Robot manipulator control: theory and practice*. CRC Press.

[44] Fareh, R., Saad, M. R., Saad, M., Brahmi, A., & Bettayeb, M. (2019). Trajectory tracking and stability analysis for mobile manipulators based on decentralized control. *Robotica*, *37*(10), 1732-1749.