A novel wheel-leg parallel mechanism control for Kinematic Reconfiguration of an Environmental Hybrid Robot

Gustavo Freitas, Fernando Lizarralde, Liu Hsu, Vitor Paranhos and Ney R. Salvi dos Reis

Abstract—This paper addresses a control design method for parallel mechanisms based on the kinematic constraints, rather than the constraint equations. The method is applied on a new Environmental robot to control the position and angle of its wheels through 2DOF suspensions with actuated parallel mechanisms. This case study is discussed based on the concepts of kinematic control and forward and differential kinematics. Experimental tests illustrate implementation issues and the efficacy of the proposed design.

I. INTRODUCTION

Mobile robots are widely employed in unstructured and hazardous terrains for several applications, including mining, forestry, agricultural and military activities ([1], [9]). For effective remote operation in inhospitable environments, efficient locomotion systems, capable of working on irregular and rough terrains, must be considered.

This paper contemplates wheel-legged robots [8], which use wheels for propulsion and internal articulation to adapt their configuration in order to accommodate for obstacles, also repositioning the center of mass and influencing the contact forces against the terrain.

Accuracy, repeatability and payload capacities are fundamental requirements for the leg mechanism when advanced mobile robots are considered. Parallel mechanisms provide a stiff connection between the payload supported by the leg, with superior pose accuracy than serial chain mechanisms [10], [11]. The main disadvantages related to this type of structures are related to workspace limitation, difficulties in mapping the forward kinematic and complex singularity analysis [16], [13].

This paper presents a parallel mechanism control methodology based on a strategy proposed in [16], where the differential kinematic model is obtained considering the mechanism’s kinematic constrains from its structure equations, instead of using explicitly the constrains equations. Then, according to the system’s effective degrees-of-freedom (DOF), a control strategy with primary and auxiliary objectives is proposed considering the mechanism differential kinematic model null space.

The described methodology is applied, controlling the suspensions of Environmental Hybrid Robot’s (EHR) new prototype [4]. Each suspension is composed by a planar parallel mechanism with 2 DOF, allowing the wheel’s position and orientation to be independently controlled.

II. ENVIRONMENTAL HYBRID ROBOT

This work is motivated by the Environmental Hybrid Robot [5], [4] from Brazilian Oil company Petrobras S.A. (Fig. 1), which has four suspensions composed by parallel mechanisms. The robot is meant to help monitoring the Amazon rain forest region, being used as a maintenance tool for the Coari-Manaus pipeline [7].

An innovative locomotion system is designed according to the encountered conditions in Amazon. The wheel-legged architecture is adopted, allowing the robot to carry load and save energy, being able to work on irregular terrains.

Each wheel is coupled to an independent suspension system composed of a spring and electric linear actuators. The suspension’s motors are attached to screws, forming the active prismatic joints. The suspensions’ parallel mechanism is designed for structural rigidity, allowing the robot to overcome obstacles and increasing wheel’s traction.

The original suspension developed for the EHR [4] has only 1 DOF. When the coupled motor of this structure is actuated, both position and angle of its wheel are amended. By commanding the wheel’s height, it is possible to modify the robot mobility, e.g. the system’s stability and force distribution among the legs [5], [4]. Also, the suspension angle influences the contact point with the terrain and consequently the effective radio of the spherical wheels. Hence, the suspension’s reconfiguration affects the relation between velocity and torque on each wheel.

The capability of influencing the system’s stability, its force distribution and also the relation between velocity and torque is desirable. However, to optimize the robot’s missions, the suspension mechanism should allow this parameters to be independently controlled. Because of that, a new mechanism with 2 DOF is being developed, allowing the wheel’s height and angle to be decoupled. The new suspension also has a larger workspace, and even allows the robot to operate upside down, as presented in Fig. 2.

The robot uses 20 plastic bicycle wheels with solid tyres, which will be further covered by a profile as shown in Fig. 1 in order to...
increase traction and allow the EHR to buoy. These floating wheels are being designed to maximize traction considering the different operation environments.

The mobile robot weighs 35 kg with 490 mm of wheelbase and track width of 710 mm. NiMH batteries supply energy for the system. A set of motors (70 W for each wheel and 10 W for each suspension active joint) and drivers (EPOs) from Maxon Motors are used, allowing the robot to navigate at firm terrains with maximum speed of 1.50 m/s.

The system is teleoperated. Communication with a base computer is accomplished through 900 MHz Ethernet radios. The embedded controls are executed by a PC/104 board with a 366 MHz processor. The robot’s orientation is obtained by an MTI Xsens inertial unit.

The new Environmental Robot prototype is still under development, and new equipments will be integrated into the system. The robot’s careen and floating wheels, which are under construction, must be fixed before tests in the Amazon Rain Forest.

III. LEG FORWARD KINEMATICS

The forward kinematics for a robotic system is described by the end-effector configuration specified for each chain. Here, the end-effector is represented by the contact point between wheel and terrain.

The contact point position is considered to be located in the wheel most inferior point as presented in Fig. 3.

Each wheel is coupled to an independent suspension system with 2 effective DOF (it can be verified using the Gluebler’s formula [12]), corresponding to an underactuated planar mechanism with 2 closed kinematic chains.

The mechanism’s geometry is presented in Fig. 3. It is a 7-bar-link, where link 0 is fixed to the robot’s structure and the wheel corresponds to the link segment 6. The leg has 2 active prismatic and 6 passive revolute joints.

In order to describe the leg’s forward kinematics, coordinate frames are attached to each link of it, where coordinate frame \(\{x_i, y_i\}\) is attached to the \(i\)th-link (see Fig. 4). The wheel-terrain contact position is defined by frame \(\{x_w, y_w\}\) and \(\vec{p}_{ij} \in \mathbb{R}^3\) is the vector from the \(i\)-frame origin to \(j\)-frame origin represented in base frame (frame 0). The mechanism’s links length are presented in Table I.

![Fig. 3. EHR’s suspension: seven-bar-link mechanism geometry with active and passive joints.](image)

![Fig. 4. EHR’s suspension: seven-bar-link mechanism frames.](image)

<table>
<thead>
<tr>
<th>Link #</th>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(a_0)</td>
<td>185</td>
<td>mm</td>
</tr>
<tr>
<td></td>
<td>(\psi_0)</td>
<td>117.0</td>
<td>deg</td>
</tr>
<tr>
<td>3</td>
<td>(a_{30})</td>
<td>80</td>
<td>mm</td>
</tr>
<tr>
<td></td>
<td>(a_{31})</td>
<td>40</td>
<td>mm</td>
</tr>
<tr>
<td></td>
<td>(a_{32})</td>
<td>40</td>
<td>mm</td>
</tr>
<tr>
<td>4</td>
<td>(a_4)</td>
<td>70</td>
<td>mm</td>
</tr>
<tr>
<td>6</td>
<td>(a_{60})</td>
<td>75</td>
<td>mm</td>
</tr>
<tr>
<td></td>
<td>(a_{61})</td>
<td>115</td>
<td>mm</td>
</tr>
<tr>
<td></td>
<td>(a_w)</td>
<td>275</td>
<td>mm</td>
</tr>
<tr>
<td></td>
<td>(\psi_6)</td>
<td>112.0</td>
<td>deg</td>
</tr>
</tbody>
</table>

**TABLE I**

EHR SUSPENSION MECHANISM LINKS LENGTH.
The forward kinematics is described by the following structure equation [12]:

\[
\begin{align*}
\vec{p}_{0w} &= \vec{p}_{02} + \vec{p}_{24} + \vec{p}_{46} + \vec{p}_{6w} \\
&= \vec{p}_{03} + \vec{p}_{34} + \vec{p}_{46} + \vec{p}_{6w} \\
&= \begin{cases} 
\theta_1 + \theta_2 + \theta_4 + \theta_6 = \theta_1 + \theta_2 + 2\pi + \theta_5 + \pi - \psi_0 & \text{chain 1} \\
\theta_3 + \psi_0 + \theta_4 + \theta_6 = \psi_0 + \theta_3 + \theta_4 + \theta_6 & \text{chain 2} \\
\psi_0 + \theta_3 + \theta_4 + \theta_6 & \text{chain 3}
\end{cases} \\
\theta_{0w} &= \begin{cases} 
\theta_1 + \theta_2 + \theta_4 + \theta_6 = \theta_1 + \theta_2 + 2\pi + \theta_5 + \pi - \psi_0 & \text{chain 1} \\
\psi_0 + \theta_3 + \theta_4 + \theta_6 & \text{chain 3}
\end{cases}
\end{align*}
\]

where \( \theta_{0w} \) represents the orientation of the wheel frame with respect to base frame.

Equations (1)–(2) introduce constraints between the possible joint angles. These constraints allow the end-effector location control to be done by specifying only the active joints variables \( \theta_a = [d_1, d_2]^T \). The passive joints \( \theta_p = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T \) take on values making equations (1)–(2) satisfied.

From (1), the forward kinematics is given by:

\[
\vec{p}_{0w} = \vec{p}_{02} + \vec{p}_{24} + \vec{p}_{46} + \vec{p}_{6w} = d_2 \cdot R_{02}(\theta_1) \cdot x + (a_{30} + a_{31}) \cdot R_{01}(\theta_1) \cdot R_{23}(\theta_2) \cdot x + a_4 \cdot R_{01}(\theta_1) \cdot R_{23}(\theta_2) \cdot R_{34}(\theta_4) \cdot x + a_{61} \cdot R_{01}(\theta_1) \cdot R_{23}(\theta_2) \cdot R_{34}(\theta_4) \cdot R_{60}(\theta_6) \cdot x - a_w \cdot R_{01}(\theta_1) \cdot R_{23}(\theta_2) \cdot R_{34}(\theta_4) \cdot R_{60}(\theta_6) \cdot y \tag{3}
\]

where \( a_{30}, a_{31}, a_4, a_{61}, a_w \) are values from table 1, \( x = [1, 0, 0]^T \), \( y = [0, 1, 0]^T \) and the elementary rotation matrix \( R_{ij}(\theta_i) \in SO(3) \) is the orientation of \( j \)-frame with respect to \( i \)-frame, given by:

\[
R_{ij}(\theta_i) = \begin{bmatrix}
\cos(\theta_i) & -\sin(\theta_i) & 0 \\
\sin(\theta_i) & \cos(\theta_i) & 0 \\
0 & 0 & 1
\end{bmatrix}
\tag{4}
\]

Due to the parallel mechanism’s structure constraints, it is not trivial to obtain an analytical solution for the forward kinematics in terms of \( \theta_a \), i.e. \( \vec{p}_{0w} = k_1(d_1, d_2), \theta_{0w} = k_2(d_1, d_2) \).

The passive revolute joints \( \theta_p \in \mathbb{R}^6 \) are obtained in terms of the active joints \( d_1, d_2 \), i.e. \( \theta_1 = f_1(d_1), \theta_3 = f_3(d_1), \theta_4 = f_4(d_1), \theta_5 = f_5(d_2), \theta_6 = f_6(d_2) \). These functions can be calculated through algebraic identities and geometry (by using cosine formula).

Another possibility to determine \( \theta_p \) is to reduce the forward kinematic problem into appropriate subproblems whose solutions are known [12]. The passive joints \( \theta_p \) can be calculated through Paden-Kahan subproblems 1 and 3 [12, pag. 99–103], which are geometrically meaningful and numerically stable.

Knowing \( \theta_p \), it is possible to obtain the contact point position \( \vec{p}_{0w} \) through equation (3) and orientation \( \theta_{0w} \) using (2).

Figure 5 illustrates suspension’s configurations for \( d_1 \in [170, 220] \text{mm} \) and \( d_2 \in [45, 105] \text{mm} \). The mechanism’s dexterous workspace in terms of \( \vec{p}_{0w} \) and \( \theta_{0w} \) is presented in Fig. 6.

**IV. DIFFERENTIAL KINEMATICS**

The planar parallel mechanism end-effector’s (wheel-terrain contact point) velocity is related to the joints’ velocities by differentiating the structure equation (1). This gives a Jacobian matrix for each chain:

\[
v_w = S \cdot J_1 \begin{bmatrix}
\dot{\theta}_1 \\
\dot{d}_1 \\
\dot{\theta}_2 \\
\dot{d}_2 \\
\dot{\theta}_3 \\
\dot{d}_3 \\
\dot{\theta}_4 \\
\dot{d}_4 \\
\end{bmatrix} = S \cdot J_2 \begin{bmatrix}
\dot{\theta}_1 \\
\dot{d}_1 \\
\dot{\theta}_2 \\
\dot{d}_2 \\
\dot{\theta}_3 \\
\dot{d}_3 \\
\end{bmatrix} = S \cdot J_3 \begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2 \\
\dot{\theta}_3 \\
\end{bmatrix} \tag{5}
\]

where \( v_w = [\vec{p}_{0w}, \dot{p}_{0w}, \omega_{0w}]^T \in \mathbb{R}^3 \) is the planar linear velocity in \( x_0, y_0 \) axes and angular velocity \( \omega_{0w} = \dot{\theta}_{0w} \), \( S \in \mathbb{R}^{3 \times 6} \) is a selection matrix considering the planar mechanism:

\[
S = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]
and the 6-DOF Jacobian matrices are given by
\[
J_1 = \begin{bmatrix}
z \times \vec{p}_{2w} & R_{01} x & z \times \vec{p}_{2w} & z \times \vec{p}_{4w} & z \times \vec{p}_{6w} \\
0 & z & z & z & z \end{bmatrix}
\]
\[
J_2 = \begin{bmatrix}
z \times \vec{p}_{0w} & R_{01} x & z \times \vec{p}_{2w} & R_{01} R_{23} x & z \times \vec{p}_{6w} \\
0 & z & 0 & z & z \end{bmatrix}
\]
\[
J_3 = \begin{bmatrix}
z \times \vec{p}_{3w} & z \times \vec{p}_{4w} & z \times \vec{p}_{6w} \\
z & z & z & z & z \end{bmatrix}
\]
where \( z = [0, 0, 1]^T \), \( \vec{p}_{0w} \) is given in (3) and
\[
\vec{p}_{2w} = (a_{30} + a_{31}) R_{01} (\theta_1 R_{23} (\theta_2) + z \vec{p}_{4w})
\]
\[
\vec{p}_{3w} = a_{41} R (\psi (\theta_3) x + \vec{p}_{4w})
\]
\[
\vec{p}_{4w} = a_{40} R_{01} (\theta_1) R_{23} (\theta_2) R_{34} (\theta_4) x + \vec{p}_{0w}
\]
\[
\vec{p}_{6w} = a_{60} R_{01} (\theta_1) R_{23} (\theta_2) R_{34} (\theta_4) y
\]
\[
\vec{p}_{0w} = [a_{60} \cos (\psi + a_{61}) - a_{60} \sin (\psi - a_{62})] y
\]
(10)
\[
\vec{p}_{0w} = a_{60} R_{01} (\theta_1) R_{23} (\theta_2) R_{34} (\theta_4) R_{46} (\theta_6) x
\]
(11)
where it is possible to observe that all \( \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6 \) are duplicated in equation (12), and because of that, \( J_{ext} \) has more columns than necessary. Defining \( \theta_{ext} = M \theta \), with
\[
M = \begin{bmatrix}
I^3 & 0^{3 \times 2} & 0^{3 \times 3} \\
I^2 & 0^{2 \times 2} & 0^{2 \times 3} \\
I & 0^{1 \times 2} & 0^{1 \times 3} \\
I & 0^{1 \times 2} & 0^{1 \times 3} \\
I & 0^{1 \times 2} & 0^{1 \times 3} \\
I & 0^{1 \times 2} & 0^{1 \times 3} \\
I & 0^{1 \times 2} & 0^{1 \times 3} \\
I & 0^{1 \times 2} & 0^{1 \times 3} \\
\end{bmatrix}
\]
where \( I^i \) is an identity matrix and \( 0^{i \times j} \) is a zero matrix with \( i \) rows and \( j \) columns, equation (12) is rewritten as:
\[
J \dot{\theta} = A v_w
\]
(14)
with \( J \in \mathbb{R}^{9 \times 8} = J_{ext} M \) and \( \dot{\theta} = [\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6]^T \).
Considering \( A^T \in \mathbb{R}^{9 \times 9} \) the pseudo-inverse of \( A \) such that \( A^T A = I \), and \( A \in \mathbb{R}^{9 \times 9} \) the annihilator matrix as \( AA = 0 \), the differential kinematic relationship (14) can be equivalently written as [17], [14]:
\[
v_w = J_1 \dot{\theta}, \quad J_2 \dot{\theta} = 0
\]
(15)
where \( J_1 = A^T J \) and \( J_2 = A J \).
Considering the active joints \( [d_1, d_2]^T = \dot{\theta}_a \) and collecting all the passive joints velocities together in \( \dot{\theta}_p \) and then partitioning \( J_p \in \mathbb{R}^{3 \times 8} \) and \( \dot{\theta}_p \in \mathbb{R}^{6 \times 8} \) accordingly
\[
J_1 = [J_{ta}; J_{tp}], \quad J_2 = [J_{ca}; J_{cp}]
\]
(16)
considering (15) and (16), one has that:
\[
v_w = J_{ta} \dot{\theta}_a + J_{tp} \dot{\theta}_p, \quad J_{ca} \dot{\theta}_a + J_{cp} \dot{\theta}_p = 0
\]
(17)
where \( J_{ta} \in \mathbb{R}^{3 \times 2}, J_{tp} \in \mathbb{R}^{3 \times 6}, J_{ca} \in \mathbb{R}^{6 \times 2} \) and \( J_{cp} \in \mathbb{R}^{6 \times 6} \).

If \( J_{cp} \) is full rank, the passive joints’ velocities can be given in terms of active joint velocity \( \dot{\theta}_a \):
\[
\dot{\theta}_p = -J_{cp}^{-1} J_{ca} \dot{\theta}_a
\]
(18)

It is possible to verify that \( J_{cp} \) is not singular for the considered mechanism.
Finally, the wheel velocity \( v_w \) is given in terms of the active joints’ velocities \( d_1, d_2 \) by:
\[
v_w = (J_{ta} - J_{tp} J_{cp}^{-1} J_{ca}) \dot{\theta}_a
\]
(19)
where \( J_p \in \mathbb{R}^{3 \times 2} \) is the leg’s Jacobian.

V. KINEMATIC CONTROL

A kinematic control approach is proposed to change the suspension’s parallel mechanism posture in order to accomplish an specific task. Considering the system with 2 effective DOF, the adopted control has two objectives:

1) control the vertical distance \( p_{0w} \) between wheel-terrain contact point and the suspension frame:
\[
p_{0w} \rightarrow p_{0w}^* (t) \quad e_h = p_{0w}^* - p_{0w} \rightarrow 0
\]
(20)

2) regulate the wheel’s orientation \( \theta_{0w} \) acting on the mechanism null space velocities with respect to the primary objective:
\[
\theta_{0w} \rightarrow \phi_{0w}, \quad e_h = \theta_{0w}^* - \theta_{0w} \rightarrow 0
\]
(21)
The proposed scheme consists in commanding the robot’s joints’ velocities \( d_1, d_2 \), i.e. \( u(t) = \dot{\theta}_a \), adjusting the contact point’s pose to cancel the errors (20) and (21).
The actuated planar suspension’s differential kinematic model (19) can be decoupled in 2 components: a linear \( \vec{p}_{0w} \) and an angular velocity \( \omega_{0w} \) as:
\[
\begin{bmatrix}
\vec{p}_{0w} \\
\theta_{0w}
\end{bmatrix} = J_p \dot{\theta}_a, \quad J_p = \begin{bmatrix} J_{py} \\ J_{py}
\end{bmatrix}
\]
where \( J_{py} \in \mathbb{R}^{1 \times 2} \) is thus the joints’ velocities can be obtained from \( \dot{\theta}_a = \dot{\theta}_a \), adjusting the contact point’s pose to cancel the errors (20) and (21).
From Fig. 6, it is possible to observe that the suspension’s workspace has boundary singularities in terms of orientation. Considering \( p_{0w} < -320 mm \), the orientation singularity is achieved for \( \phi_{0w} = 6 \pi rad \).
However, in these configurations, it is still possible to control \( p_{0w} \). The vertical distance between suspension frame and wheel-terrain contact point directly influences the robot’s orientation, force distribution among legs and tip over stability as presented in [4], [3]. Thus it corresponds to the main control objective, avoiding orientation singularities.

The adopted approach consists on given priority to control the vertical distance, considering orientation as an auxiliary control. Therefore, the mechanism is considered redundant, with 1 effective DOF and 2 active joints. The differential kinematic is given by \( \vec{p}_{0w} = J_{py} \dot{\theta}_a \), and the joints’ velocities are given by [15], [2]:
\[
\dot{\theta}_a = J_{py}^T v + \left(I - J_{py}^T J_{py}\right) \dot{\theta}_a
\]
(22)
where \( \text{col}(P) \) spans the null space of \( J_{py} \) and \( \dot{\theta}_a \) is an arbitrary vector of active joints’ velocities. The right hand side of (22) can be interpreted as null space velocity which provides internal movements with respect to the redundant mechanism’s model.
Then, using a proportional law as

\[ u = J_p^T K_h \dot{e}_h + (I - J_p^T J_p) \bar{u}, \quad (23) \]

where \( \bar{u} \) is the auxiliary control signal, the position error dynamic is determined by \( \dot{e}_h + K_h \dot{e}_h = 0 \), since the right hand side of (23) spans the null space of \( J_p \). Considering \( K_h > 0 \) and nonsingular \( J_p \), we have \( \lim_{t \to \infty} e_h(t) = 0 \).

The auxiliary control \( \bar{u} \) is chosen to improve the mechanism’s performance during tasks execution. A typical choice is

\[ \bar{u} = \bar{K} \left( \frac{\partial f(\theta_a)}{\partial \theta_a} \right)^T, \quad (24) \]

where \( \bar{K} > 0 \) is a gain factor and \( f(\theta_a) \) is an objective function\(^1\) in terms of \( \theta_a \), here chosen to cancel the orientation error: \( f(\theta_a) = e_o^2 \).

Considering \( f(\theta_a) = e_o^2 \), the auxiliary control is given by:

\[ \bar{u} = K_o \dot{e}_o \quad \text{where} \quad K_o \gg 0 \]

Finally, the active joints’ velocities \( \dot{\theta}_a \) are defined as:

\[ u = \dot{\theta}_a = J_p^T K_h \dot{e}_h + (I - J_p^T J_p) K_o J_p^T e_o \quad (26) \]

VI. CONTROL IMPLEMENTATION AND EXPERIMENTAL RESULTS

This section describes the kinematic control’s implementation in the Environmental Robot. The control algorithm is executed on a PC/104 single board computer featuring a low-power, 366 MHz AMD Geode GX500 processor running Linux (Debian 5.0.1, Kernel 2.6.34), which is embedded in the robot.

To achieve real-time code execution, Xenomai 2.5.3 is used side-by-side with Linux [6]. Xenomai uses the Adaptive Domain Environment for Operating Systems (ADEOS) to share hardware interrupts with Linux and guarantee real-time operation. The implementation described here uses Xenomai’s native API.

The EHR’s software is implemented in C/C++. The user-space application spawns the following real-time task that executes the control algorithm:

```c
for(allMechanisms){
    readActualPositions();
    computeForwardKinematics();
    getReferences();
    calculateErrors();
    computeVelocities();
}
for(allMechanisms){
    sendHardwareTheVelocities();
    logValues();
}
```

Among the several custom functions that comprise the code, the following two are particularly relevant:

- `forwardKinematics()`: Calculates the leg’s forward kinematics \( p_{w_0}^{\text{ref}} \) from the active joint position vector \( q_a \).
- `suspensionControl()`: Calculates the control command \( u = \dot{q}_a \) from the joint position vectors \( q_a, q_p \) and closed-loop position and orientation errors \( e_h, e_o \).

\(^1\) \( f(\cdot) \) is continuous, differentiable and convex.

The function `forwardKinematics` is obtained with Matlab’s `ccode` command. The mathematical expression for the forward leg kinematics is declared as a symbolic expression in Matlab, and `ccode` is called to generate the C code implementing the expression. Because `ccode` does not always generate computationally-efficient code, we hand-customized the code to reduce computation speed, by storing in memory those variables that are calculated frequently and reusing them during code execution. This manual optimization step resulted in a code that runs 80% faster (see Table II).

The `suspensionControl` function is also obtained via `ccode` and manual optimization. The matrix inversion \( J_p^{-1} \) is computed using the open source computer vision library named OpenCV. The library provides functions for inverting matrices using singular value decomposition (SVD) as well as Gaussian elimination with optimal pivot element selection (LU). The computational effort for both the original and optimized codes using each of the matrix inversion methods is shown in Table III. The optimized code with LU inversion runs 38% faster than its original counterpart.

![Table II](image)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>E[t]</th>
<th>σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Optimized forwardKinematics</td>
<td>165.79µs</td>
<td>74.72µs</td>
</tr>
<tr>
<td>Optimized forwardKinematics</td>
<td>35.19µs</td>
<td>6.50µs</td>
</tr>
</tbody>
</table>

![Table III](image)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>E[t]</th>
<th>σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Optimized with LU inversion</td>
<td>353.60µs</td>
<td>99.79µs</td>
</tr>
<tr>
<td>Non-Optimized with SVD inversion</td>
<td>1074µs</td>
<td>158.93µs</td>
</tr>
<tr>
<td>Optimized with LU inversion</td>
<td>219.07µs</td>
<td>65.05µs</td>
</tr>
<tr>
<td>Optimized with SVD inversion</td>
<td>942.57µs</td>
<td>173.61µs</td>
</tr>
</tbody>
</table>

In the experiments reported here the wheel’s vertical position reference \( p_{w_0}^{\text{ref}} \) is a series of step signals, and the orientation reference is such that the wheel is perpendicular to the ground.

The results are presented in Fig. 7. It can be seen that the primary control objective of the vertical position of the wheel-terrain contact point is accomplished, with \( e_h \to 0 \) within a time of about 3 s.

Likewise, the secondary control objective is also achieved, with the orientation error \( e_o \) tending to zero at every change in reference position. Because of the reduced priority placed on orientation control, the time constant is longer, on the order of 20 s. Note that the orientation error is not canceled when the active joint \( d_1 \) reaches its minimum position at \( t = [140, 160] \) s. This, however, is a physical limitation of the leg’s mechanism and not an intrinsic limitation of the controller. More importantly, even at these configurations the null-space control methodology still allows for full control of the vertical distance \( p_{w_0}^{\text{ref}} \), since the orientation control acts on the null space of \( J_p \) and therefore does not interfere with the control of \( p_{w_0}^{\text{ref}} \).
This paper presents a control methodology for parallel mechanisms, based on a modeling procedure proposed in literature which consider the mechanism’s kinematic constraints from its structure equations, instead of explicitly using the constraints equations. A kinematic control is then proposed considering the mechanism's DOF to improve achieving primary and auxiliary objectives.

The presented methodology is demonstrated considering the parallel 2 DOF suspension mechanism of the Environmental Hybrid Robot’s new prototype. The mechanism’s forward and differential kinematics are obtained. Then, a strategy is presented to control firstly the vertical distance between wheel and base frame and then the wheel’s orientation.

Experimental results are presented considering a constant orientation reference $\theta_{\text{ref}}$ and a sequence of steps as vertical position reference $p_{\text{ref}}$. The control implementation is described, presenting details related to real time computation. Tests with the EHR are also presented.

Future works will consist on commanding the combined suspensions in order to improve the robot mobility, i.e. traction and stability, as proposed in [4], [3].

The Environmental Hybrid Robot’s new prototype is still under development. Some preliminary tests have been accomplished in laboratory and field, and are presented in Fig. 8. The robot’s complete functionality will be achieved when the floating wheels are integrated into the system.

VIII. ACKNOWLEDGMENTS

The authors would like to thank all the Petrobras S.A. Robotics Laboratory team.

This work was partially supported by FAPERJ, CNPq, CAPES and Petrobras S.A.

REFERENCES