Visual Servoing of Par4 using Leg Observation

Tej Dallej [1], Nicolas Andreff [1,2] and Philippe Martinet [1]

[1] LASMEA - 24 avenue des landais
63177, Aubière Cedex, France
firstname.lastname@lasmea.univ-bpclermont.fr
http://wwwlasmea.univ-bpclermont.fr/Control

[2] LaMI - BP 265 - 63175, Aubière Cedex, France

***Abstract*— In this paper, the original approach is proposed for the vision-based control of Par4 parallel robot. Observing the mechanism legs with a calibrated camera vision simplifies the control law based on the geometry of lines and the image projection of cylindrical legs. The method is illustrated and validated on a Par4 platform simulation.**

I. INTRODUCTION

Controlling parallel robots is a hard task. Indeed, simple joint control does not take into account the kinematic constraints due to the closed kinematic chains of such mechanisms. Hence, it may yield high internal forces [1] and can only achieve a good repeatability (relative positioning error). The latter is generally higher than in the serial case [2] and is due mainly to the mechanical constraints, while the achievable accuracy (absolute positioning error) is usually in the same order as serial robot accuracy [3]. The latter property comes from an inaccurate conversion from desired Cartesian pose to desired joint values, mainly due to the fact that the large number of links and passive joints makes it very hard to manufacture and assemble a physical parallel robot close to its CAD model. Therefore, one has to use advanced control techniques.

To do so, visual servoing techniques [4] are a good alternative for the control of parallel mechanisms, since they close the control loop over the vision sensor. The fundamental assumption is that external sensing of the end-effector pose in the feedback signal replaces advantageously the forward kinematic model, since the perception models are simpler than the kinematic models and contain less unmodelled physical phenomena. Visual servoing was however seldom applied to parallel robotics [5], [6], [7]. Those applications rely on 3D visual servoing [8] where the end-effector pose is indirectly measured and used for regulation. The main limitation of this approach is that it requires the estimation of the end-effector to tool transformation, the world to base transformation and the whole kinematic parameter set. Moreover, observing the end-effector of a parallel mechanism may be incompatible with its application. For instance, it is not wise to imagine observing the end-effector of a machine tool.

To overcome these problems, a new way to use vision which gathers the advantages of redundant metrology [9] and of visual servoing was presented in [10]. This method proposes to vision-based control the well-known six DOF Gough-Stewart platform [11], [12]. It has a reduced set of kinematic parameters and does not require any visual target. In that method, the leg orientations were chosen as visual primitives. The control law was based on their reconstruction from the image which might not be very accurate for intrinsic reasons.

Fig. 1. Par4-LIRMM.

To make control robust with respect to modelling errors, it was proposed [13] to servo the leg edges rather than the leg orientations which improved the practical robustness by servoing the legs in the image.

Consequently, we want to extend these previous results concerning edge-based visual servoing scheme validated in the Gough-Stewart platform to another parallel robot. The reason for such a choice is that many different robotic tasks such as the 'pick-and-place' task do not need six DOF which only needs three DOF in translation and one DOF in rotation. A lot of work deals with four degrees of freedom parallel mechanism [14] among which one can find the family derived from the H4 [15], namely the I4L [16] and the I4R [17]. Recently, a new parallel manipulator based on H4 and I4 architectures named Par4 (Fig. 1) was presented in [18].

The Par4 robot has cylindrical legs as in the case of the Gough-Stewart platform. Thus, those cylinders can be observed to extract, directly from the image, the leg edges and the latter can be used as visual primitives in the control. The expected advantages of this method is that it should not only reduce the kinematic parameter set but also simplify the kinematic models.

Therefore, the contribution of this paper is to present the vision-based control of Par4 by observing their leg edges using

1-4244-0136-4/06/$20.00©2006 IEEE

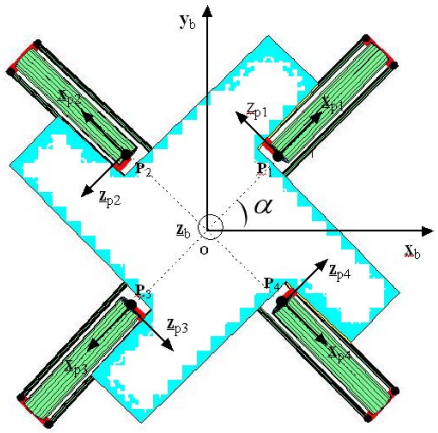

Fig. 2. Par4 parameters.

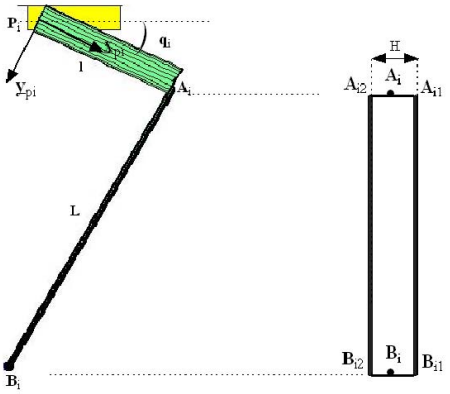

Fig. 3. Leg parameters.

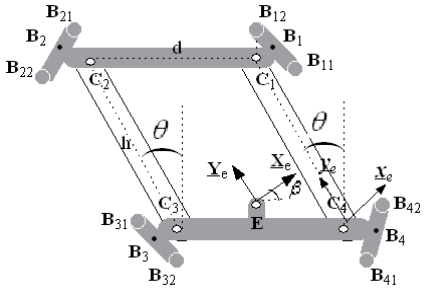

Fig. 4. Nacelle parameters.

a camera fixed with respect to the base.

Section II recalls the description of Par4 modelling. Section III recalls the vision-based framework for expressing the robot kinematics and control. Then, the visual servoing method is presented in Section IV and validated in Section V. Conclusions are to be found in Section VI.

II. Mechanism Modeling

The Par4 is based on four identical kinematic chains (Fig. 2). Each revolute motor, located in $\mathbf{P}_i$, moves an arm (parallelogram) i (Fig. 3) attached in points $\mathbf{P}_i$ and $\mathbf{A}_i$. Each arm is connected to a forearm made of two legs equipped with ball joints ($(\mathbf{A}_{i1}, \mathbf{A}_{i2})$ and $(\mathbf{B}_{i1}, \mathbf{B}_{i2})$). These forearms are connected, at each end, to the articulated traveling plate (nacelle) (Fig. 4) that is the difference with the architecture of H4 [15] or I4R [17]. As shown in [18], this nacelle is composed of four parts: two parts defined by $[\mathbf{C}_1\mathbf{C}_2]$ and $[\mathbf{C}_3\mathbf{C}_4]$ and linked by two rods with revolute joints.

To simplify the kinematic model, we choose $\mathbf{C}_4$ as the end-effector position. Indeed, the articulated nacelle is actually equipped by an amplification system to transform the relative rotation θ into a proportional rotation ($\beta = \kappa\theta$) in the actual end-effector $\mathbf{E}$. However, this amplification system is not very significant from a kinematic point of view.

As most parallel mechanism, the Par4 is designed with slim and rectilinear legs. Therefore, its legs observation by vision may exhibit a projective kinematic model for control [19].

The following notations are introduced:

- $i = 1..4$ denotes the legs and $j = 1..2$ denotes the edges.
- Boldface characters denote vectors. Unit vectors are underlined.
- $\mathcal{F}_b = (\mathbf{O}, \underline{\mathbf{x}}_b, \underline{\mathbf{y}}_b, \underline{\mathbf{z}}_b)$, $\mathcal{F}_e = (\mathbf{C}_4, \underline{\mathbf{x}}_e, \underline{\mathbf{y}}_e, \underline{\mathbf{z}}_e)$, $\mathcal{F}_c = (\mathbf{O}_c, \underline{\mathbf{x}}_c, \underline{\mathbf{y}}_c, \underline{\mathbf{z}}_c)$ and $\mathcal{F}_{pi} = (\mathbf{P}_i, \underline{\mathbf{x}}_{pi}, \underline{\mathbf{y}}_{pi}, \underline{\mathbf{z}}_{pi})$ denote respectively the base, end-effector, camera and i^{th} arm reference frames.
- We express all kinematic parameters in the camera frame $\mathcal{F}_c$.
- q_i is the articulated position of each arm.
- In the base frame we can write:

$$ {}^b\underline{\mathbf{x}}_{p_i} = \begin{pmatrix} \cos q_i \cos \alpha_i \\ \cos q_i \sin \alpha_i \\ -\sin q_i \end{pmatrix}, \; {}^b\underline{\mathbf{z}}_{p_i} = \begin{pmatrix} -\sin \alpha_i \\ \cos \alpha_i \\ 0 \end{pmatrix}, $$

$ {}^b\underline{\mathbf{y}}_{p_i} = {}^b\underline{\mathbf{z}}_{p_i} \times {}^b\underline{\mathbf{x}}_{p_i}$ and $\alpha_i = \alpha + (i-1)\frac{\pi}{2}$.

- ${}^cV_e$ is the operational velocity of the end-effector $\mathbf{C}_4$ with coordinates $(\dot{x}, \dot{y}, \dot{z})^T$ and ${}^c\Omega_e$ its rotational velocity which is defined by the angular velocity ω_z around the fixed axis ${}^c\underline{\mathbf{z}}_e$. So, the cartesian velocity of the end-effector are minimally represented by:

$$ {}^c\mathcal{V}_e = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_z \end{pmatrix} $$

- ${}^c\overrightarrow{\mathbf{A}_{ij}\mathbf{B}_{ij}} = {}^c\overrightarrow{\mathbf{A}_{i1}\mathbf{B}_{i1}} = {}^c\overrightarrow{\mathbf{A}_{i2}\mathbf{B}_{i2}} = L\,{}^c\underline{\mathbf{u}}_i$.
- ${}^c\overrightarrow{\mathbf{P}_i\mathbf{A}_i} = l\,{}^c\underline{\mathbf{x}}_{p_i}$.
- ${}^c\overrightarrow{\mathbf{A}_{i1}\mathbf{A}_{i2}} = H\,{}^c\underline{\mathbf{z}}_{pi}$.

III. Vision-based Kinematics

A. Line representation

Let $\mathcal{L}$ be a 3D line. A point-independent representation of this line are the Plücker coordinates $(\underline{\mathbf{u}}, \mathbf{n})$ [20] (also known as normalized Plücker coordinates since us $\underline{\mathbf{u}}$ is a unit vector), where $\underline{\mathbf{u}}$ is the direction of the line and $\mathbf{n}$ encodes its position.

However, noticing that $\mathbf{n}$ is orthogonal to the so-called interpretation plane defined by $\mathcal{L}$ and the origin, one can split it into two parts: $\underline{\mathbf{n}}$, the unit vector defining the interpretation

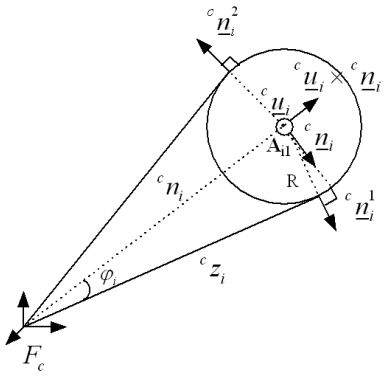

Fig. 5. Visual edges of a cylinder.

plane and n, its norm which is the orthogonal distance of $\mathcal{L}$ to the origin.

Doing so, [21] defined the so-called binormalized Plücker coordinates $(\underline{\mathbf{u}}, \underline{\mathbf{n}}, n)$ of a 3D line, where only $\underline{\mathbf{n}}$ is useful for image projection. Indeed, The projection of such a line in the image plane, expressed in the camera frame, has for characteristic equation:

$$ {}^c\underline{\mathbf{n}}^T {}^c\mathbf{p} = 0 \tag{1}$$

where ${}^c\mathbf{p}$ are the coordinates in the camera frame of any point in the image plane, lying on the line.

With the intrinsic parameters $\mathbf{K}$, one can easily obtain the conversion from the line equation in the camera frame ${}^c\underline{\mathbf{n}}$ to the same in the pixel coordinates ${}^p\underline{\mathbf{n}}$:

$$\left\{ \begin{array}{l} {}^p\underline{\mathbf{n}} = \frac{\mathbf{K}^{-T}{}^c\underline{\mathbf{n}}}{\|\mathbf{K}^{-T}{}^c\underline{\mathbf{n}}\|} \\ {}^c\underline{\mathbf{n}} = \frac{\mathbf{K}^T{}^p\underline{\mathbf{n}}}{\|\mathbf{K}^T{}^p\underline{\mathbf{n}}\|} \end{array} \right. \tag{2}$$

B. Image projection of a leg

As shown in Fig. 3, each forearm i is made of two legs $[\mathbf{A}_{i1}\mathbf{B}_{i1}]$ and $[\mathbf{A}_{i2}\mathbf{B}_{i2}]$. We choose to observe only the first leg designed by $[\mathbf{A}_{i1}\mathbf{B}_{i1}]$.

Assuming that each cylinder edge is a line in space, with binormalized Plücker coordinates expressed in the camera frame $({}^c\underline{\mathbf{u}}_i, {}^c\underline{\mathbf{n}}_i, {}^c n_i)$, as illustrated in Fig. 5, one can compute the interpretation planes (${}^c\underline{\mathbf{n}}_i^1$ and ${}^c\underline{\mathbf{n}}_i^2$) of the two edges of a cylinder $[\mathbf{A}_{i1}\mathbf{B}_{i1}]$:

$$\left\{ \begin{array}{rcl} {}^c\underline{\mathbf{n}}_i^1 & = & \cos\varphi_i \ {}^c\underline{\mathbf{n}}_i - \sin\varphi_i \ {}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i \\ {}^c\underline{\mathbf{n}}_i^2 & = & -\cos\varphi_i \ {}^c\underline{\mathbf{n}}_i - \sin\varphi_i \ {}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i \end{array} \right. \tag{3}$$

where $\cos\varphi_i = {}^c z_i / {}^c n_i$, $\sin\varphi_i = R/{}^c n_i$, ${}^c z_i = \sqrt{({}^c n_i)^2 - R^2}$ and R is the cylinder radius.

Let us remark now that the attachment point $\mathbf{A}_{i1}$ (Fig. 5) is lying on the revolution axis of the leg with radius R. Consequently, a cylinder edge is also defined by the following constraints expressed in the camera frame:

$$\left\{ \begin{array}{l} {}^c\underline{\mathbf{n}}_i^{j\,T} {}^c\underline{\mathbf{u}}_i = 0 \\ {}^c\underline{\mathbf{n}}_i^{j\,T} {}^c\underline{\mathbf{n}}_i^j = 1 \\ {}^c\mathbf{A}_{i1}{}^T {}^c\underline{\mathbf{n}}_i^j = -R \end{array} \right. \tag{4}$$

where, in the present case,

$${}^c\mathbf{A}_{i1} = {}^c\mathbf{P}_i + l {}^c\underline{\mathbf{x}}_{pi} + {}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}} \tag{5}$$

C. The differential inverse kinematic model of Par4

If we observe only the edges of each leg $[\mathbf{A}_{i1}\mathbf{B}_{i1}]$, the expression of the kinematic chains closure (Fig. 3) yields the so-called implicit kinematic model in vector form [19]:

$$L {}^c\underline{\mathbf{u}}_i = {}^c\overrightarrow{\mathbf{A}_{i1}\mathbf{B}_{i1}} = {}^c\overrightarrow{\mathbf{P}_i\mathbf{B}_{i1}} - {}^c\overrightarrow{\mathbf{P}_i\mathbf{A}_{i1}} \tag{6}$$

Notice that this expression is available also using the second leg $[\mathbf{A}_{i2}\mathbf{B}_{i2}]$ or a fictive leg $[\mathbf{A}_i\mathbf{B}_i]$.

Time differentiating (6) yields:

$$L {}^c\dot{\underline{\mathbf{u}}}_i = \frac{d}{dt}({}^c\overrightarrow{\mathbf{P}_i\mathbf{B}_{i1}}) - \frac{d}{dt}({}^c\overrightarrow{\mathbf{P}_i\mathbf{A}_{i1}}) \tag{7}$$

Using (5) and

$${}^c\overrightarrow{\mathbf{P}_i\mathbf{B}_{i1}} = {}^c\overrightarrow{\mathbf{P}_i\mathbf{O}} + {}^c\overrightarrow{\mathbf{O}\mathbf{C}_4} + {}^c\overrightarrow{\mathbf{C}_4\mathbf{B}_{i1}} \tag{8}$$

and assuming that ${}^c\overrightarrow{\mathbf{P}_i\mathbf{O}}$ and ${}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}}$ are constant, (7) becomes:

$$L {}^c\dot{\underline{\mathbf{u}}}_i = {}^cV_e + \frac{d}{dt}({}^c\overrightarrow{\mathbf{C}_4\mathbf{B}_{i1}}) - l\frac{d}{dt}({}^c\underline{\mathbf{x}}_{pi}) \tag{9}$$

Taking into account that the parallelogram form of nacelle introduces a circular translation [18], one can easily obtain:

$$\left\{ \begin{array}{l} \frac{d}{dt}({}^c\underline{\mathbf{x}}_{pi}) = \dot{q}_i {}^c\underline{\mathbf{y}}_{pi} \\ \\ \frac{d}{dt}({}^c\overrightarrow{\mathbf{C}_4\mathbf{B}_{i1}}) = (\omega_z {}^c\underline{\mathbf{z}}_e) \times \varepsilon_i (h {}^c\underline{\mathbf{y}}_e) = -\varepsilon_i h \omega_z {}^c\underline{\mathbf{x}}_e \\ \\ with \\ \varepsilon_3 = \varepsilon_4 = 0 \\ \varepsilon_1 = \varepsilon_2 = 1 \end{array} \right. \tag{10}$$

where ε_i denotes whether $\mathbf{B}_i$ is located on the same part as $\mathbf{C}_4$ or not (Fig. 4). Hence, we obtain:

$$L {}^c\dot{\underline{\mathbf{u}}}_i = {}^cV_e - \varepsilon_i h \omega_z {}^c\underline{\mathbf{x}}_e - l\dot{q}_i {}^c\underline{\mathbf{y}}_{pi} \tag{11}$$

which can be simplified into matrix form:

$$L {}^c\dot{\underline{\mathbf{u}}}_i = \left(\begin{array}{cc} \mathbf{I}_3 & -\varepsilon_i h {}^c\underline{\mathbf{x}}_e \end{array} \right) {}^c\mathcal{V}e - l\dot{q}_i {}^c\underline{\mathbf{y}}_{pi} \tag{12}$$

with ${}^c\mathcal{V}_e = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_z \end{pmatrix}$

Using ${}^c\underline{\mathbf{u}}_i{}^T {}^c\dot{\underline{\mathbf{u}}}_i = 0$, we can obtain, from (12), the expression of the differential inverse kinematic model, associated to joint velocities, of Par4:

$$\dot{q}_i = \frac{1}{l {}^c\underline{\mathbf{y}}_{pi}^T {}^c\underline{\mathbf{u}}_i} \left(\begin{array}{cc} {}^c\underline{\mathbf{u}}_i^T & -\varepsilon_i h {}^c\underline{\mathbf{u}}_i^T {}^c\underline{\mathbf{x}}_e \end{array} \right) {}^c\mathcal{V}_e = {}^c\mathbf{D}_{ei}^{invc} \mathcal{V}_e \tag{13}$$

The differential inverse kinematic model associated to joint velocities is:

$$\dot{q} = {}^c\mathbf{D}_e^{invc} \mathcal{V}_e \tag{14}$$

where

$${}^c\mathbf{D}_e^{inv} = \begin{pmatrix} \frac{1}{l {}^c\underline{\mathbf{y}}_{p1}^T {}^c\underline{\mathbf{u}}_1} & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \frac{1}{l {}^c\underline{\mathbf{y}}_{p4}^T {}^c\underline{\mathbf{u}}_4} \end{pmatrix} \begin{pmatrix} {}^c\underline{\mathbf{u}}_1 T & -h {}^c\underline{\mathbf{u}}_1^T {}^c\underline{\mathbf{x}}_e \\ {}^c\underline{\mathbf{u}}_2^T & -h {}^c\underline{\mathbf{u}}_2^T {}^c\underline{\mathbf{x}}_e \\ {}^c\underline{\mathbf{u}}_3^T & 0 \\ {}^c\underline{\mathbf{u}}_4^T & 0 \end{pmatrix} \tag{15}$$

To complete the differential kinematic model [19], we need to define the differential relation between ${}^c\dot{\underline{\mathbf{u}}}_i$ and ${}^c\mathcal{V}_e$:

$$ {}^c\dot{\underline{\mathbf{u}}}_i = \mathbf{M}_i {}^c\mathcal{V}_e \tag{16} $$

This can be derived by inserting (13) in (12):

$$ L {}^c\dot{\underline{\mathbf{u}}}_i = (\mathbf{I}_3 - \frac{{}^c\underline{\mathbf{y}}_{pi} {}^c\underline{\mathbf{u}}_i^T}{{}^c\underline{\mathbf{y}}_{pi}^T {}^c\underline{\mathbf{u}}_i}) \begin{pmatrix} \mathbf{I}_3 & -\varepsilon_i h {}^c\underline{\mathbf{x}}_e \end{pmatrix} {}^c\mathcal{V}_e \tag{17} $$

The differential kinematic model associated to a leg orientation is hence:

$$ \mathbf{M}_i = \frac{1}{L}(\mathbf{I}_3 - \frac{{}^c\underline{\mathbf{y}}_{pi} {}^c\underline{\mathbf{u}}_i^T}{{}^c\underline{\mathbf{y}}_{pi}{}^T {}^c\underline{\mathbf{u}}_i}) \begin{pmatrix} \mathbf{I}_3 & -\varepsilon_i h {}^c\underline{\mathbf{x}}_e \end{pmatrix} \tag{18} $$

In the differential kinematic model, one needs to compute: ${}^c\underline{\mathbf{x}}_e = {}^c\underline{\mathbf{y}}_e \times {}^c\underline{\mathbf{z}}_e$, ${}^c\underline{\mathbf{y}}_e = \frac{{}^c\overrightarrow{\mathbf{C}_4\mathbf{C}_1}}{h}$ and ${}^c\underline{\mathbf{z}}_e = {}^c\underline{\mathbf{z}}_b$. with

$$ {}^c\mathbf{C}_4 = {}^c\mathbf{B}_4 - d_4 {}^c\underline{\mathbf{x}}_c + h_4 {}^c\underline{\mathbf{y}}_c \tag{19} $$

$$ {}^c\mathbf{C}_1 = {}^c\mathbf{B}_1 - d_1 {}^c\underline{\mathbf{x}}_c - h_1 {}^c\underline{\mathbf{y}}_c \tag{20} $$

where (d_4, h_4) and (d_1, h_1) are constant parameters. The effective computation of $\mathbf{C}_i$, and hence of $\mathbf{B}_i$, depends on the positions of $\mathbf{B}_{i1}$ which are computed using the articular positions vectors ($l {}^c\underline{\mathbf{x}}_{pi}$) and the directions ${}^c\underline{\mathbf{u}}_i$:

$$ {}^c\mathbf{B}_{i1} = {}^c\mathbf{A}_{i1} + L {}^c\underline{\mathbf{u}}_i = {}^c\mathbf{P}_i + {}^c\overrightarrow{\mathbf{P}_i\mathbf{A}_i} + {}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}} + L {}^c\underline{\mathbf{u}}_i \tag{21} $$

$$ {}^c\mathbf{B}_{i1} = {}^c\mathbf{P}_i + l {}^c\underline{\mathbf{x}}_{pi} + {}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}} + L {}^c\underline{\mathbf{u}}_i \tag{22} $$

IV. Visual servoing of Par4

A. Edge Interaction matrix

We propose to servo some error between the current ${}^p\underline{\mathbf{n}}_i^j$ and desired cylinder edges ${}^p\underline{\mathbf{n}}_i^{j^*}$. This control makes use of the detected edges in the image (Fig. 6). It should be more robust than a control based on the cylinder orientations in the camera frame [13].

To derive such a control, we need to express the interaction matrix ${}^p\mathbf{L}_i^j$ relating the Cartesian velocity ${}^c\mathcal{V}_e$ to the time derivative of the cylinder edges ${}^p\dot{\underline{\mathbf{n}}}_i^j$, expressed in the image frame:

$$ {}^p\dot{\underline{\mathbf{n}}}_i^j = {}^p\mathbf{L}_i^j {}^c\mathcal{V}_e \tag{23} $$

It can be computed as the product of two matrices:

$$ {}^p\mathbf{L}_i^j = {}^p\mathbf{J}_c {}^c\mathbf{L}_i^j \tag{24} $$

where ${}^p\mathbf{J}_c$ is associated to the camera-to-pixel change of frame:

$$ {}^p\dot{\underline{\mathbf{n}}}_i^j = {}^p\mathbf{J}_c {}^c\dot{\underline{\mathbf{n}}}_i^j \tag{25} $$

and the second matrix ${}^c\mathbf{L}_i^j$ relates the time derivative of a cylinder edge ${}^c\dot{\underline{\mathbf{n}}}_i^j$, in the camera frame, to ${}^c\mathcal{V}_e$.

$$ {}^c\dot{\underline{\mathbf{n}}}_i^j = {}^c\mathbf{L}_i^j {}^c\mathcal{V}_e \tag{26} $$

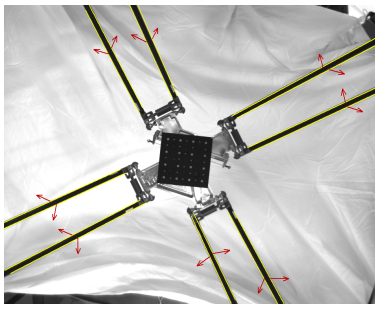

Fig. 6. Detected edges in the image, here in the case of the I4R.

B. Image line velocity in pixel coordinates

Using (2) for each cylinder, we can write:

$$ {}^p\underline{\mathbf{n}}_i^j = \frac{\mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j}{\parallel \mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j \parallel} = \frac{\mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j}{((\mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j)^T \mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j)^{\frac{1}{2}}} \tag{27} $$

which yields:

$$ {}^p\dot{\underline{\mathbf{n}}}_i^j = \frac{1}{\parallel \mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j \parallel}(\mathbf{I}_3 - \frac{\mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j (\mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j)^T}{\parallel \mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j \parallel^2}) \mathbf{K}^{-T} {}^c\dot{\underline{\mathbf{n}}}_i^j \tag{28} $$

From (2) again one can compute:

$$ \mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j = {}^p\underline{\mathbf{n}}_i^j \parallel \mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j \parallel \tag{29} $$

and

$$ \parallel \mathbf{K}^{-T} {}^c\underline{\mathbf{n}}_i^j \parallel = \frac{1}{\parallel \mathbf{K}^T {}^p\underline{\mathbf{n}}_i^j \parallel} \tag{30} $$

Hence, inserting (29) and (30) into (28), one gets the following expression for ${}^p\mathbf{J}_c$:

$$ {}^p\mathbf{J}_c = \parallel \mathbf{K}^T {}^p\underline{\mathbf{n}}_i^j \parallel (\mathbf{I}_3 - {}^p\underline{\mathbf{n}}_i^j {}^p\underline{\mathbf{n}}_i^{jT}) \mathbf{K}^{-T} \tag{31} $$

C. Edge velocity in the camera frame

Note that $({}^c\underline{\mathbf{u}}_i, {}^c\underline{\mathbf{n}}_i^j, {}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)$ form an orthonormal basis. Consequently, one can write:

$$ {}^c\dot{\underline{\mathbf{n}}}_i^j = \gamma_1 {}^c\underline{\mathbf{u}}_i + \gamma_2 {}^c\underline{\mathbf{n}}_i^j + \gamma_3 {}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j \tag{32} $$

One can compute γ_1, γ_2 et γ_3 by time differentiating the constraints (4), which gives:

$$ {}^c\dot{\underline{\mathbf{n}}}_i^{jT} {}^c\underline{\mathbf{u}}_i + {}^c\dot{\underline{\mathbf{u}}}_i^T {}^c\underline{\mathbf{n}}_i^j = 0 \tag{33} $$

$$ {}^c\dot{\underline{\mathbf{n}}}_i^{jT} {}^c\underline{\mathbf{n}}_i^j = 0 \tag{34} $$

$$ {}^c\dot{\mathbf{A}}_{i1}^T {}^c\underline{\mathbf{n}}_i^j + {}^c\mathbf{A}_{i1}{}^T {}^c\dot{\underline{\mathbf{n}}}_i^j = 0 \tag{35} $$

From (5), we have:

$$ {}^c\dot{\mathbf{A}}_{i1} = l\dot{q}_i {}^c\underline{\mathbf{y}}_{pi} \tag{36} $$

Inserting the later into (35) yields:

$$ {}^c\mathbf{A}_{i1}{}^T {}^c\dot{\underline{\mathbf{n}}}_i^j = -l {}^c\underline{\mathbf{y}}_{pi}^T {}^c\underline{\mathbf{n}}_i^j \dot{q}_i \tag{37} $$

Then, we obtain from (32) and (33):

$$ \gamma_1 = -{}^c\dot{\underline{\mathbf{u}}}_i{}^T {}^c\underline{\mathbf{n}}_i^j = -{}^c\underline{\mathbf{n}}_i^{jT} {}^c\dot{\underline{\mathbf{u}}}_i \tag{38} $$

and from (32) and (34):

$$\gamma_2 = 0 \tag{39}$$

Finally, using (32) and (37), we get:

$$\gamma_3 = -\frac{l\,{}^c\underline{\mathbf{y}}_{pi}^T\,{}^c\underline{\mathbf{n}}_i^j}{{}^c\mathbf{A}_{i1}^T({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)}\dot{q}_i + \frac{{}^c\mathbf{A}_{i1}^T\,{}^c\underline{\mathbf{u}}_i}{{}^c\mathbf{A}_{i1}^T({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)}\,{}^c\underline{\mathbf{n}}_i^{jT}\,{}^c\dot{\underline{\mathbf{u}}}_i \tag{40}$$

Hence, ${}^c\dot{\underline{\mathbf{n}}}_i^j$ can be written as:

$${}^c\dot{\underline{\mathbf{n}}}_i^j = (\mathbf{Q}_i^j\,{}^c\mathbf{D}_{ei}^{inv} + \mathbf{R}_i^j\mathbf{M}_i)\,{}^c\mathcal{V}_e = {}^c\mathbf{L}_i^j\,{}^c\mathcal{V}_e \tag{41}$$

with

$$\begin{cases} \mathbf{Q}_i^j = -\frac{l({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)\,{}^c\underline{\mathbf{y}}_{pi}^T\,{}^c\underline{\mathbf{n}}_i^j}{{}^c\mathbf{A}_{i1}{}^T\,{}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j} \\ \mathbf{R}_i^j = -(\mathbf{I}_3 - \frac{({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)\,{}^c\mathbf{A}_{i1}{}^T}{{}^c\mathbf{A}_{i1}{}^T({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)})\,{}^c\underline{\mathbf{u}}_i\,{}^c\underline{\mathbf{n}}_i^{jT} \\ {}^c\mathbf{L}_i^j = \mathbf{Q}_i^j\,{}^c\mathbf{D}_{ei}^{inv} + \mathbf{R}_i^j\mathbf{M}_i \end{cases} \tag{42}$$

D. Control

Since we want to drive the unit vectors associated to the leg edges to their desired values, we choose to servo the geodesic errors:

$$e_{i,j} = {}^p\underline{\mathbf{n}}_i^j \times {}^p\underline{\mathbf{n}}_i^{j^*} \tag{43}$$

whose time derivatives are

$$\dot{e}_{i,j} = {}^p\dot{\underline{\mathbf{n}}}_i^j \times {}^p\underline{\mathbf{n}}_i^{j^*} = -[{}^p\underline{\mathbf{n}}_i^{j^*}]_\times\,{}^p\dot{\underline{\mathbf{n}}}_i^j \tag{44}$$

Using (23), one can verify that:

$$\dot{e}_{i,j} = \mathbf{N}_i^j\,{}^c\mathcal{V}_e \tag{45}$$

with

$$\mathbf{N}_i^j = -[{}^p\underline{\mathbf{n}}_i^{j^*}]_\times\,{}^p\mathbf{L}_i^j \tag{46}$$

This yields the following pseudo-control vector ${}^c\mathcal{V}_e$:

$${}^c\mathcal{V}_e = -\lambda\mathbf{N}^+ e \tag{47}$$

with $\mathbf{N}$ is the compound matrix from the associated individual interaction matrices $\mathbf{N}_i^j$.

Inserting (47) into (14) delivers the final control law:

$$\dot{q} = -\lambda\,{}^c\widehat{\mathbf{D}_e^{inv}}\,\widehat{\mathbf{N}^+}e \tag{48}$$

V. Results

In all the simulations presented below, we choose the same initial position obtained by choosing ${}^b\mathbf{C}_4 = (0, 0, -900mm)^T$ as the end-effector position and $\theta = 0°$.

In a first simulation, the desired configuration (Fig. 8) is obtained from the initial one only by a pure rotation of 20° around the $\underline{\mathbf{z}}_e$ axis.

Fig. 7 shows that the errors on legs 1 ($e_{1,1}$ and $e_{1,2}$) and 2 ($e_{2,1}$ and $e_{2,2}$) converge to 0. Moreover, errors on legs 3 ($e_{3,1}$ and $e_{3,2}$) and 4 ($e_{4,1}$ and $e_{4,2}$) are initially equal to 0 and stay equal to 0, which is coherent with the expected pure

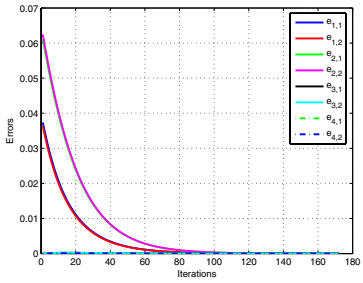

Fig. 7. Evolution of the (unit-less) error $e_{i,j}^T e_{i,j}$ on each edge, case of a pure rotation.

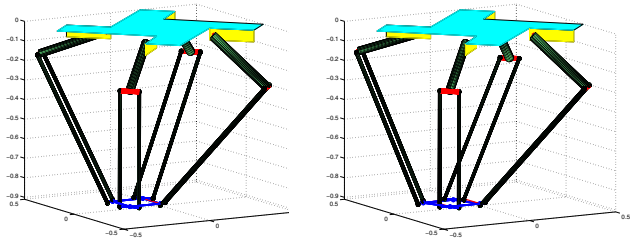

Fig. 8. Desired (right) and initial (left) position, case of a pure rotation.

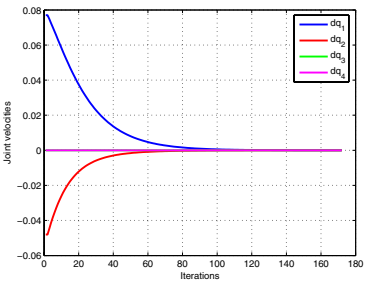

Fig. 9. Evolution of the joint velocities (rad/s) on each legs, case of a pure rotation.

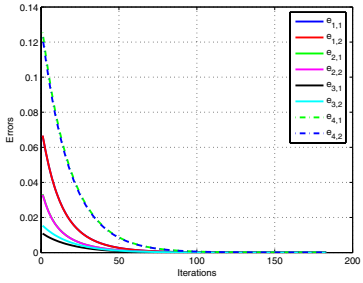

Fig. 10. Evolution of the (unit-less) error $e_{i,j}^T e_{i,j}$ on each edge, case of a general motion.

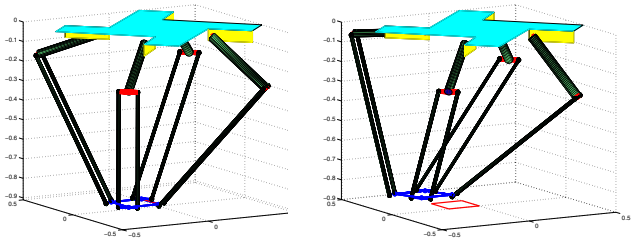

Fig. 11. Desired (right) and initial (left) position, case of a general motion.

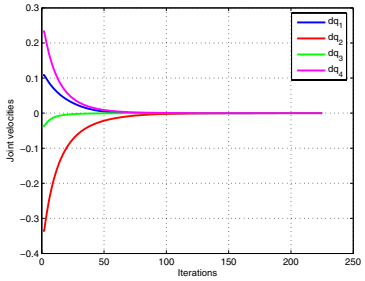

Fig. 12. Evolution of the joint velocities (rad/s) on each legs, case of a general motion.

rotation. This is additionally demonstrated in Fig. 9 where the joint velocities of legs 3 and 4 are also 0.

In a second simulation, the desired position (Fig. 11) is obtained from the reference position by a translation along the 3 axes and a rotation around the $\underline{\mathbf{z}}_e$ axis (${}^b\mathbf{C}_4 = (100mm, -100mm, -850mm)^T$ and $\theta = 20°$). Fig. 10 shows that the errors on each leg converge exponentially to 0, with a perfect decoupling. Fig. 12 shows that the joint velocities on each leg converge also to 0.

VI. Conclusion

In this paper, we proposed a novel approach for vision-based control of a Par4 parallel robot. The presented method was validated on a Par4 platform simulation.

This method differs from standard control in the joint space by the fact that the forward kinematic model is never used in the control. This yields a higher robustness to calibration errors since the control is done directly in the image space.

Moreover, this method differs from standard visual servoing techniques with end-effector observation by the fact that it does not require the use of a visual pattern to calibrate the relative pose of the later with respect to the end-effector. It can even be used when the end-effector is not visible.

The next step in this work be to implement the control on a real Par4 platform. This shall induce the development of a calibration method to estimate the kinematics parameters that are expressed in the camera frame using also the legs observation. It could be followed by robustness studies, as well as the extension of the method to other mechanism.

Acknowledgments

This work was supported by European Community through the IP projet NEXT number 0011815.

References

[1] B. Dasgupta and T.S. Mruthyunjaya. Force redundancy in parallel manipulators: theoretical and practical issues. *Mech. Mach. Theory*, 33(6):727–742, 1998.

[2] J.P. Merlet. *Parallel robots*. Kluwer Academic Publishers, 2000.

[3] J. Wang and O. Masory. On the accuracy of a Stewart platform - Part I: The effect of manufacturing tolerances. In *Proc. ICRA93*, pages 114–120, 1993.

[4] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3), June 1992.

[5] M.L. Koreichi, S. Babaci, F. Chaumette, G. Fried, and J. Pontnau. Visual servo control of a parallel manipulator for assembly tasks. In *6th Int. Symposium on Intelligent Robotic Systems, SIRS'98*, pages 109–116, Edimburg, Scotland, July 1998.

[6] H. Kino, C.C. Cheah, S. Yabe, S. Kawamua, and S. Arimoto. A motion control scheme in task oriented coordinates and its robustness for parallel wire driven systems. In *Int. Conf. Advanced Robotics (ICAR'99)*, pages 545–550, Tokyo, Japan, Oct. 25-27 1999.

[7] P. Kallio, Q. Zhou, and H. N. Koivo. Three-dimensional position control of a parallel micromanipulator using visual servoing. In Bradley J. Nelson and Editors Jean-Marc Breguet, editors, *Microrobotics and Microassembly II, Proceedings of SPIE*, volume 4194, pages 103–111, Boston, USA, November 2000.

[8] B. Thuilot, P. Martinet, L. Cordesses and J. Gallice. Position based visual servoing: keeping the object in the field of vision. *Proceedings of the IEEE International Conference on Robotics and Automation* (ICRA'02), pages 1624–1629, May 2002.

[9] L. Tencredi, M. Teillaud, and J.P. Merlet. Forward kinematics of a parallel manipulator with additional rotary sensors measuring the position of platform joints. *Computational Kinematics, J.P. Merlet and B. Ravani, Eds., Dordrecht*, pages 261–270, December 1995.

[10] N. Andreff, A. Marchadier and P. Martinet. Vision-based control of Gough-Stewart parallel mechanism using legs observation. In *Int. Conf. Robotics and Automation (ICRA'05)*, pages 2546-2551, Barcelona, Spain, May 2005.

[11] D. Stewart. A platform with six degrees of freedom. In *Proc. IMechE (London)*, volume 180, pages 371–386, 1965.

[12] V.E. Gough and S.G. Whitehall. Universal tyre test machine. In *Proc. FISITA 9th Int. Technical Congress*, pages 117–137, May 1962.

[13] N. Andreff T. Dallej and P. Martinet. Image-based visual servoing of Gough-Steawart parallel manipulators using legs observation. In *Proceedings of the 8th International IFAC Symposium on Robot Control SYROCO 2006*, Santa Cristina Convent, University of Bologna (Italy) September 6 - 8, 2006, (To appear).

[14] S. Krut. Contribution à l'étude des robots parallèles légers, 3T-1R et 3T-2R, à forts débattements angulaires. *PhdThesis*, Université Montpellier II, 2003.

[15] F. Pierrot and O. Company. H4: A new family of 4-dof parallel robots. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM'99*, pages 508–513, Atlanta, Georgia, USA, September 1999.

[16] S. Krut, O. Company, M. Benoit, H. Ota, and F. Pierrot. I4: A new parallel mechanism for SCARA motions. In *Proc. IEEE/ICRA International Conference on Robotics and Automation, ICRA'03*, pages 1875–1880, Taipei, Taiwan, September 2003.

[17] S. Krut, V. Nabat, O. Company, and F. Pierrot. A high speed robot for SCARA motions. In *Proc. IEEE/ICRA International Conference on Robotics and Automation, ICRA'04*, pages 4109–4115, New Orleans, USA, April 26 - May 1 2004.

[18] V. Nabat, M. O. Rodrigues, O. Company, S. Krut, F. Pierrot. Par4: very high speed parallel robot for pick-and-place. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, pages 1202–1207, Alberta, Canada, August 2-6 2005.

[19] N. Andreff, and P. Martinet. Kinematic modelling of some parallel manipulators for vision-based control purposes. In *Proc. of EuCoMes, the first European Conference on Mechanism Science*, Obergurg, Austria, February 21-26 2006.

[20] J. Plücker. On a new geometry of space. *Philosophical Transactions of the Royal Society of London*, 155:725–791, 1865.

[21] N Andreff, B Espiau, and R Horaud. Visual servoing from lines. *Int. Journal of Robotics Research*, 21(8):679–700, 2002.