Vision-Based Control of a Gough-Stewart Parallel Mechanism using Legs Observation

(Corrected)

Nicolas Andreff and Arnaud Marchadier and Philippe Martinet
LASMEA - CNRS - Université Blaise Pascal/IFMA, 63175 Aubière, France
Email: {andreff | martinet}@lasmea.univ-bpclermont.fr
http://wwwlasmea.univ-bpclermont.fr/Control

Abstract— **This paper presents a novel approach for vision-based control of the end-effector of parallel mechanisms. It is based on the metrological redundancy paradigm, which simplifies their kinematic models by introducing additional proprioceptive sensors. By observing the mechanism legs, vision replaces advantageously these sensors by delivering, in a Cartesian frame, an exteroceptive measurement of the internal state of the mechanism. Formally, the latter is expressed by an original concept of vision-based kinematics for parallel mechanisms. Based on it, control is derived that visually servoes the direction of the legs, rather than the end-effector pose. The method is illustrated and validated on a Gough-Stewart platform simulation.**

I. Introduction

Parallel mechanism are such that there exist several kinematic chains (or legs) between their base and their end-effector. Therefore, they may exhibit a better repeatability [1] than serial mechanisms but not a better accuracy [2], because of the large number of links and passive joints. There can be two ways to compensate for the low accuracy. The first way is to perform a kinematic calibration of the mechanism and the second one is to use a control law which is robust to calibration errors.

There exists a large amount of work on the control of parallel mechanisms (see [3] for a long list of references). In the focus of attention, Cartesian control is naturally achieved through the use of the inverse Jacobian which transforms Cartesian velocities into joint velocities. It is noticeable that the inverse Jacobian of parallel mechanisms does not only depend on the joint configuration (as for serial mechanisms) but also on the end-effector pose.

Consequently, one needs to be able to estimate or measure the latter. As far as we know, all the effort has been put on the estimation of the end-effector pose through the forward kinematic model and the joint measurements. However, this yields much trouble, related to the fact that there is usually no analytic formulation of the forward kinematic model of a parallel mechanism. Hence, one numerically inverts the inverse kinematic model, which is analytically defined for most of the parallel mechanisms. However, it is known [4], [5] that this numerical inversion requires high order polynomial root determination, with several possible solutions (up to 24 real solutions for a Gough-Stewart platform). Much of the work is thus devoted to solving this problem accurately and in real-time (see for instance [6]), or to designing parallel mechanisms with analytical forward kinematic model [7], [8]. One of the promising paths lies in the use of the so-called metrological redundancy [9], which simplifies the kinematic models by introducing additional sensors into the mechanism and thus yields easier control [10].

Computer vision being an efficient way of estimating the end-effector pose [11], [12], it is a good alternative to use it for Cartesian control of parallel mechanisms. It can be done in three ways.

a) Vision as a sensor: The first one consists in computing the end-effector poses by vision, then in translating them into joint configurations, through the inverse kinematic model, and finally servoing in the joint space. This scheme is rather easy to implement for serial mechanisms provided that inverting the forward kinematic model can be done satisfactorily. The latter is straightforward for parallel mechanisms since they usually have an analytical inverse kinematic model. Similarly, one can consider computer vision as a contact-less redundant sensor, as already stated in the context of parallel mechanism calibration [13], and use the simplified models based on the redundant metrology paradigm.

However, such schemes should be used carefully for parallel mechanisms, since joint control does not take into account the kinematic closures and may therefore yield high internal forces [14].

b) Visual servoing: Second, vision can be additionally used to perform visual servoing [15]. Indeed, instead of measuring the end-effector pose and convert it into joint values, one could think of using this measure directly for control. Recall that there exist many visual servoing techniques ranging from position-based visual servoing (PBVS) [16] (when the pose measurement is explicit) to image-based visual servoing (IBVS) [15] (when it is made implicit by using only image measurements). Most applications embed the vision system onto the end-effector to position the latter with respect to a rigid object whose accurate position is unknown, but one can also find applications with a fixed camera observing the end-effector [17]. The interested reader is referred to [18] for a thorough and up-to-date state-of-the-art.

Visual servoing techniques are very effective since they close the control loop on the vision sensor. This yields a high robustness to perturbations and calibration errors. Thus, we highly recommend to use them also for parallel mechanism control.

Essentially, these techniques generate a Cartesian desired velocity which is converted to joint actuation by the inverse Jacobian. Hence, one can translate such techniques to parallel mechanisms. It is even rather easier than in the serial case, since the inverse Jacobian of a parallel mechanism is usually analytical. The only difficulty comes from its dependency to the Cartesian pose, which would need be estimated, but, as stated above, vision can also do that ! Notice that this point pleads for PBVS, which is effectively the choice made in [19], [20], [21] for parallel robots with a reduced number of DOF.

c) A novel approach: However, these previous two ways consist solely in a simple adaptation of now classical control schemes, which, although probably very efficient, are not very innovative. Therefore, we propose a novel third way to use vision, which gathers the advantages of redundant metrology and of visual servoing and avoids most of their drawbacks.

Indeed, adding redundant sensors is not always technically feasible (think of a spherical joint) and always requires either that the sensors are foreseen at design stage or that the mechanism is physically modified to install them after its building. Anyhow, there are then additional calibration parameters in the kinematic model and one needs to estimate them in order to convert redundant joint readings into a unit vector expressed in the appropriate reference frame. Moreover, observing the end-effector of a parallel mechanism by vision may be incompatible with its application. For instance, it is not wise to imagine observing the end-effector of a machining tool. On the opposite, it should not be a problem to observe the legs of the mechanism, even in such extreme cases. Thereby one would turn vision from an exteroceptive sensor to a somewhat more proprioceptive sensor. This brings us back to the redundant metrology paradigm.

Consequently, the contribution of this paper is to present an original vision-based control of parallel mechanisms by observing their legs with a camera fixed with respect to the base. It is introduced in the case of parallel mechanisms of the hexapod type, with illustration on a Gough-Stewart platform [22], [23] (Figure 1).

The remainder of the paper is the following. Section II is devoted to vision-based kinematics of the hexapod and models the leg observation. Then, section III addresses the differential geometry aspect of the leg observation and the control derived from it. Finally, simulation results and conclusion are given respectively in section IV and section V.

II. Vision-based kinematics

A. Preliminaries

Parallel mechanisms are most often designed with slim and rectilinear legs. Thus, one is inclined to consider them as straight lines as it was done for kinematic analysis [1] or kinematic calibration [24].

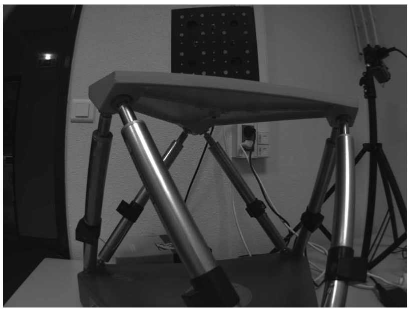

Fig. 1. A Gough-Stewart platform observed by a camera with short focal length.

Hence, we need a representation for lines suited to control. Among the work on visual servoing from lines [15], [25], [26], [27], [28], we prefer the so-called *Binormalized Plücker coordinates* representation in [25] which turns out to be coherent with kinematic modeling of parallel mechanisms.

In such a representation, a straight line in the oriented 3D space [29] is modeled by the triplet ($\underline{\mathbf{u}}$, $\underline{\mathbf{h}}$, h) where:

- $\underline{\mathbf{u}}$ is the unit vector, giving the orientation of the line ;
- $\underline{\mathbf{h}}$ is also a unit vector and h is a non-negative scalar. They are defined by $h\underline{\mathbf{h}} = \mathbf{P} \times \underline{\mathbf{u}}$ where $\mathbf{P}$ is any point on the line.

Notice that, using this notation, the well-known (normalized) Plücker coordinates [30], [31] are the couple $(\underline{\mathbf{u}}, h\underline{\mathbf{h}})$.

An interesting property of this representation, concerning computer vision, is that $\underline{\mathbf{h}} = (h_x, h_y, h_z)^T$ represents the image projection of the line, *i.e.* the equation of the image line verifies

$$h_x x + h_y y + h_z = 0 \tag{1}$$

where x and y are the coordinates of a point in the image. The interpretation of the scalar h is the orthogonal distance of the line to the center of projection.

B. Kinematics of an hexapod

Consider the hexapod in Figure 1. It has 6 legs of varying length $q_i, i \in 1..6$, attached to the base by spherical joints located in points $\mathbf{A}_i$ and to the moving platform (end-effector) by spherical joints located in points $\mathbf{B}_i$. The inverse kinematic model of such an hexapod is given by

$$\forall i \in 1..6, \quad q_i^2 = \overrightarrow{\mathbf{A}_i\mathbf{B}_i}^T \overrightarrow{\mathbf{A}_i\mathbf{B}_i} \tag{2}$$

expressing that q_i is the length of vector $\overrightarrow{\mathbf{A}_i\mathbf{B}_i}$. This model can be expressed in any Euclidean reference frame. Hence, it can be expressed in the base frame $\mathcal{R}_b$, in the end-effector frame $\mathcal{R}_e$ or in the camera frame $\mathcal{R}_c$. In the remainder and when needed, the reference frame used will be made explicit by a left upper-script.

Let us consider $(\underline{\mathbf{u}}_i, \underline{\mathbf{h}}_i, h_i)$, the Binormalized Plücker coordinates of the line passing through $\mathbf{A}_i$ and $\mathbf{B}_i$, oriented from $\mathbf{A}_i$ to $\mathbf{B}_i$. Then, we trivially have

$$\overrightarrow{\mathbf{A}_i\mathbf{B}_i} = q_i \underline{\mathbf{u}}_i \tag{3}$$

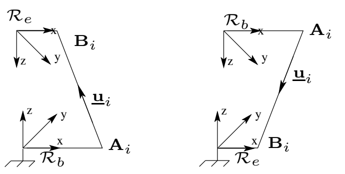

Fig. 2. Duality between the mobile end-effector mode and the fixed end-effector mode.

From [1], we know that the inverse Jacobian of the hexapod, relating the end-effector Cartesian velocity $\tau_e = \left(\begin{smallmatrix} V_e \\ \Omega_e \end{smallmatrix}\right)$ to the joint velocities is

$$\mathbf{J}_e^{inv} = \begin{bmatrix} \underline{\mathbf{u}}_1^T & (\overrightarrow{\mathbf{CB}_1}\times\underline{\mathbf{u}}_1)^T \\ \vdots & \vdots \\ \underline{\mathbf{u}}_6^T & (\overrightarrow{\mathbf{CB}_6}\times\underline{\mathbf{u}}_6)^T \end{bmatrix} \tag{4}$$

where $\mathbf{C}$ is the center of the end-effector reference frame. Notice that we write the inverse Jacobian $\mathbf{J}_e^{inv}$ rather than $\mathbf{J}^{-1}$ to clearly state that it has an analytical expression, contrary to the inverse Jacobian of a serial mechanism.

C. Vision-based kinematics of an hexapod

It has been noticed [1] that the lines of the inverse Jacobian are the Plücker coordinates of the legs. However, this is only true if the reference frame where these coordinates are expressed is centered on $\mathbf{C}$. In such a case, $\overrightarrow{\mathbf{CB}_i}, i = 1..6$ are constant and $\mathbf{J}_e^{inv}$ only depends on $\underline{\mathbf{u}}_i, i = 1..6$. Consequently, if one can measure or estimate $\underline{\mathbf{u}}_i, i = 1..6$ in the end-effector frame, one can easily convert ${}^e\tau_e$, the end-effector Cartesian velocity *expressed in the end-effector frame* into joint velocities.

This measure can be done with a camera embedded onto the end-effector (i.e. $\mathcal{R}_c = \mathcal{R}_e$) and observing the legs (see II-D). In this case, the vision-based kinematics of the hexapod expressed in the end-effector frame are very simple:

$$q_i {}^e\underline{\mathbf{u}}_i = {}^e\mathbf{B}_i - {}^e\mathbf{R}_b {}^b\mathbf{A}_i - {}^e\mathbf{t}_b \tag{5}$$

$$\dot{\mathbf{q}} = {}^e\mathbf{J}_e^{inv}\, {}^e\tau_e \tag{6}$$

$$\text{with} \quad {}^e\mathbf{J}_e^{inv} = \begin{bmatrix} {}^e\underline{\mathbf{u}}_1^T & {}^e h_1 {}^e\underline{\mathbf{h}}_1^T \\ \vdots & \vdots \\ {}^e\underline{\mathbf{u}}_6^T & {}^e h_6 {}^e\underline{\mathbf{h}}_6^T \end{bmatrix} \tag{7}$$

$${}^e h_i {}^e\underline{\mathbf{h}}_i = {}^e\mathbf{B}_i \times {}^e\underline{\mathbf{u}}_i, i = 1..6. \tag{8}$$

This formulation can apply to a classical visual servoing scheme with embedded camera. Indeed, such a scheme generates, without loss of generality, a desired ${}^e\tau_e$ from the images of an externally fixed target. In practice, it may be awkward since the camera should observe both the external target and all the legs. Alternately, several cameras could be used, but need be synchronized and calibrated with respect to each other.

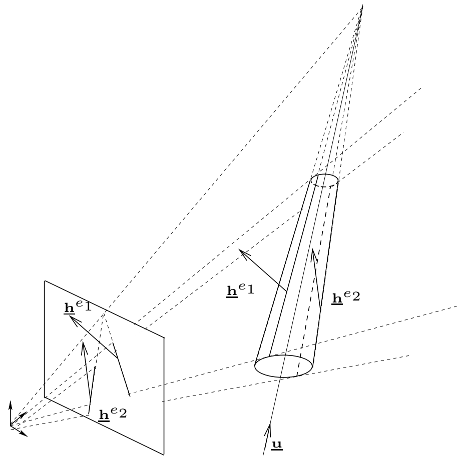
$\underline{\mathbf{h}}^{e1}$
$\underline{\mathbf{h}}^{e2}$
$\underline{\mathbf{h}}^{e1}$
$\underline{\mathbf{h}}^{e2}$
$\underline{\mathbf{u}}$

Fig. 3. Projection of a cylinder in the image.

In practice, it may thus be more convenient if the camera observing the legs is fixed to the base. Then, the reference frame associated to it is, without loss of generality, the base frame and the kinematics of the hexapod do not express as simply as in the end-effector embedded camera case. Indeed, expressed in the base frame, (4) becomes

$${}^b\mathbf{J}_b^{inv} = \begin{bmatrix} {}^b\underline{\mathbf{u}}_1^T & (\overrightarrow{{}^b\mathbf{C}{}^b\mathbf{B}_1}\times{}^b\underline{\mathbf{u}}_1)^T \\ \vdots & \vdots \\ {}^b\underline{\mathbf{u}}_6^T & (\overrightarrow{{}^b\mathbf{C}{}^b\mathbf{B}_6}\times{}^b\underline{\mathbf{u}}_6)^T \end{bmatrix} \tag{9}$$

where $\overrightarrow{{}^b\mathbf{C}{}^b\mathbf{B}_i} = {}^b\mathbf{R}_e {}^e\mathbf{B}_i \ \forall i = 1..6$. Hence, it is necessary with this expression to estimate the end-effector orientation with respect to the base frame.

An alternate formulation is possible, which is somewhat less useful for standard Cartesian control. However, it is well suited to the observation of the legs only and thereby to the control scheme proposed in III. It consists in considering the mechanism in its dual operating mode: the end-effector is fixed and the base moves with respect to it. Thus, we are interested in the inverse Jacobian relating the base Cartesian velocity ${}^b\tau_b = \left(\begin{smallmatrix} {}^bV_b \\ {}^b\Omega_b \end{smallmatrix}\right)$ to the joint velocities *expressed in the base frame*.

By analogy with (5-8), *i.e.* by permutation of the roles of $\mathbf{B}_i$ and $\mathbf{A}_i$ and of $\mathcal{R}_e$ and $\mathcal{R}_b$ (Figure 2), one obtains the vision-based kinematics of the hexapod expressed in the base frame:

$$q_i {}^b\underline{\mathbf{u}}_i = {}^b\mathbf{R}_e {}^e\mathbf{B}_i + {}^b\mathbf{t}_e - {}^b\mathbf{A}_i \tag{10}$$

$$\dot{\mathbf{q}} = {}^b\mathbf{J}_b^{inv}\, {}^b\tau_b \tag{11}$$

$$\text{with} \quad {}^b\mathbf{J}_b^{inv} = -\begin{bmatrix} {}^b\underline{\mathbf{u}}_1^T & {}^b h_1 {}^b\underline{\mathbf{h}}_1^T \\ \vdots & \vdots \\ {}^b\underline{\mathbf{u}}_6^T & {}^b h_6 {}^b\underline{\mathbf{h}}_6^T \end{bmatrix} \tag{12}$$

$${}^b h_i {}^b\underline{\mathbf{h}}_i = {}^b\mathbf{A}_i \times {}^b\underline{\mathbf{u}}_i = {}^b\mathbf{B}_i \times {}^b\underline{\mathbf{u}}_i, i = 1..6. \tag{13}$$

Notice the *minus* signs in (10) and (12), coming from the fact that in the permutation the orientation of the legs has changed. Notice also that now the inverse Jacobian is independent from the relative pose of the end-effector and base.

D. Cylindrical leg observation

As seen above, one needs to estimate the leading vector $\underline{\mathbf{u}}_i$ of each leg. Since the leading vector of a leg is essentially a Cartesian feature, we chose to estimate it by vision. Indeed, vision is an adequate tool for Cartesian sensing, and, following [24], if vision is also chosen for calibration, this does not add extra calibration parameter.

Now the problem is to recover $\underline{\mathbf{u}}_i$ from the leg observation. It may be somehow tedious, although certainly feasible, in the case of an arbitrary shape. Hopefully, for mechanical reasons such as rigidity, most of the parallel mechanisms are not only designed with slim and rectilinear legs, but, even better, with cylindrical shapes.

Consequently, the leading vector $\underline{\mathbf{u}}_i$ of the leg is also the leading vector of the cylinder edges. Thereby, simple intuitive projective geometry states that this common direction is the vanishing point of the two cylinder edges in the image (Figure 3).

The edge of a cylinder being a straight line, it can be represented by its Binormalized Plücker coordinates [32]. Let us note $\underline{\mathbf{h}}^{e_1}$ and $\underline{\mathbf{h}}^{e_2}$ the (adequately oriented) image projections of the two edges of a cylinder. Then, it is easy to show that the leading vector $\underline{\mathbf{u}}$ of the cylinder axis writes

$$\underline{\mathbf{u}} = \frac{\underline{\mathbf{h}}^{e_1} \times \underline{\mathbf{h}}^{e_2}}{\|\underline{\mathbf{h}}^{e_1} \times \underline{\mathbf{h}}^{e_2}\|} \tag{14}$$

III. Control

In this section, we address the control problem: from a given configuration of the hexapod legs observed by a camera attached to the base, how to reach a desired configuration ?

Visual servoing is based on the so-called interaction matrix $\mathbf{L}^T$ [33] which relates the instantaneous relative motion $T_c = {}^c\tau_c - {}^c\tau_s$ between the camera and the scene, to the time derivative of the vector s of all the visual primitives that are used through:

$$\dot{s} = \mathbf{L}_{(s)}^T T_c \tag{15}$$

where ${}^c\tau_c$ and ${}^c\tau_s$ are respectively the kinematic screw of the camera and the scene, both expressed in $\mathcal{R}_c$.

Then, one achieves exponential decay of an error $e(s, s_d)$ between the current primitive vector s and the desired one s_d using a proportional linearizing and decoupling control scheme of the form:

$$T_c = \lambda \hat{\mathbf{L}}_{(s)}^{T+} e(s, s_d) \tag{16}$$

where T_c is used as a pseudo-control variable.

Here also, we will need to define a visual primitive, then form an error between its current value and its desired one, then relate in some way its time derivative to the actuation, and finally find a control relation between the error and the actuation.

A. Visual primitive and error

As foreseen above, we will use the unit vectors $\underline{\mathbf{u}}_i, i = 1..6$ as visual primitives. Since these primitives are expressed in the 3D space, we are close to a PBVS scheme. However, since the reconstruction step (14) is algebraic, it is nevertheless not far away from IBVS.

The visual primitives being unit vectors, it is theoretically more elegant to use the geodesic error rather than the standard vector difference. Consequently, the error grounding the proposed control law will be

$$\mathbf{e}_i = {}^b\underline{\mathbf{u}}_i \times {}^b\underline{\mathbf{u}}_{di} \tag{17}$$

B. Interaction matrix

Here, we relate the time derivative of $\underline{\mathbf{u}}_i$ to the actuation. From (3), we immediately obtain

$${}^b\dot{\underline{\mathbf{u}}}_i = \frac{1}{q_i}\frac{d}{dt}\overrightarrow{{}^b\mathbf{A}_i{}^b\mathbf{B}_i} - \frac{\dot{q}_i}{q_i}{}^b\underline{\mathbf{u}}_i \tag{18}$$

Inserting the interaction matrix associated to a 3D point [16] applied to the moving point ${}^c\mathbf{B}_i$:

$$\frac{d}{dt}\overrightarrow{{}^b\mathbf{A}_i{}^b\mathbf{B}_i} = \begin{bmatrix} -\mathbf{I}_3 & \widetilde{{}^b\mathbf{B}_i} \end{bmatrix} {}^b\tau_b \tag{19}$$

where $\tilde{}$ is the antisymmetric matrix associated to the cross product, into (18) yields

$${}^b\dot{\underline{\mathbf{u}}}_i = -\frac{1}{q_i}\begin{bmatrix} \mathbf{I}_3 & -\widetilde{{}^b\mathbf{B}_i} \end{bmatrix} {}^b\tau_b - \frac{\dot{q}_i}{q_i}{}^b\underline{\mathbf{u}}_i \tag{20}$$

It is interesting to see that both the base Cartesian velocity and the joint velocity vector appear in this expression, while also being linked to each other by the inverse Jacobian in (12). This is certainly due to the existence of closed kinematic chains.

Nevertheless, using precisely the linking inverse Jacobian, one can exhibit a relationship between each ${}^b\dot{\underline{\mathbf{u}}}_i$ and ${}^b\tau_b$ only. Indeed, each line of the inverse Jacobian in (12) rewrites as

$$-\begin{bmatrix} {}^b\underline{\mathbf{u}}_i^T & {}^bh_1{}^b\underline{\mathbf{h}}_i^T \end{bmatrix} = -{}^b\underline{\mathbf{u}}_i^T \begin{bmatrix} \mathbf{I}_3 & -\widetilde{{}^b\mathbf{B}_i} \end{bmatrix} \tag{21}$$

Hence, we get the following relationship

$$\begin{aligned} {}^b\dot{\underline{\mathbf{u}}}_i &= \mathbf{M}_i^T\,{}^b\tau_b && (22)\\ \mathbf{M}_i^T &= -\frac{1}{q_i}\left(\mathbf{I}_3 - {}^b\underline{\mathbf{u}}_i{}^b\underline{\mathbf{u}}_i^T\right)\begin{bmatrix} \mathbf{I}_3 & -\widetilde{{}^b\mathbf{B}_i} \end{bmatrix} && (23) \end{aligned}$$

where $\mathbf{M}_i^T$ is obviously of rank 2.

Since ${}^c\mathbf{B}_i = {}^c\mathbf{A}_i + q_i{}^c\underline{\mathbf{u}}_i$, one can rewrite the above expression, using uniquely constant (${}^c\mathbf{A}_i$) or easily measurable (q_i and ${}^c\underline{\mathbf{u}}_i$) quantities, as

$$\mathbf{M}_i^T = -\frac{1}{q_i}\left(\mathbf{I}_3 - {}^c\underline{\mathbf{u}}_i{}^c\underline{\mathbf{u}}_i^T\right)\begin{bmatrix} \mathbf{I}_3 & -[{}^c\mathbf{A}_i + q_i{}^c\underline{\mathbf{u}}_i]_\times \end{bmatrix} \tag{24}$$

Necessary condition 1: A minimum of 3 independent legs is necessary to control the end-effector pose, provided that there exists a diffeomorphism between the task space and the Cartesian space $se(3)$.

A interaction matrix $\mathbf{M}^T$ can then obtained by stacking $\mathbf{M}_i^T, i = 1..6$. However, it is, in our opinion, an open question whether $\mathbf{M}$ shall or shall not be considered as an interaction matrix. Indeed, in visual servoing the various visual primitives are the image projections of objects in space that are rigidly linked to each other, while, here, each of the legs is in relative motion with respect to the other ones.

Nevertheless, effective control can be derived as shown in the following section.

C. Control law

Let us choose a control such that $E = (\mathbf{e}_1^T, ..., \mathbf{e}_6^T)^T$ decreases exponentially, i.e. such that

$$\dot{E} = -\lambda E \tag{25}$$

Then, introducing $\mathbf{N}_i^T = -\widetilde{{}^b\underline{\mathbf{u}}_{di}}\mathbf{M}_i^T$ and $\mathbf{N}^T = (\mathbf{N}_1, ..., \mathbf{N}_6)^T$, the combination of (17), (22) and (25) gives

$$\mathbf{N}^T \, {}^b\tau_b \;=\; -\lambda E \tag{26}$$

The Cartesian control velocity is hence

$${}^b\tau_b = -\lambda \mathbf{N}^{T+} E \tag{27}$$

and can be transformed into the control joint velocities using (11)

$$\dot{\mathbf{q}} = -\lambda {}^b\mathbf{J}_b^{inv}\mathbf{N}^{T+} E \tag{28}$$

Notice that since the joint velocities are obtained through the inverse Jacobian, they are admissible and do not generate any internal force, provided that the inverse Jacobian is accurate and the time sampling high enough.

IV. Results

A. Simulating a parallel mechanism

Due to the kinematic closure constraints, it is not that easy to simulate a parallel mechanism for control. Indeed, to simulate the end-effector pose ${}^b\mathbf{T}_e$, from which all the other information on the mechanism can be obtained, one has two options: either one integrates in the joint space and solves for the forward kinematic problem at each time period, or one integrates directly the end-effector velocity with respect to the base ${}^b\tau_e$ in the Cartesian space.

The first option is troublesome because of the non analytic forward kinematic problem. Thus, we prefer the second option, which *only* requires to be very cautious (especially with the integration step), since one integrates on a curved spaced in place of a vector space.

Once the integration step is validated, then the simulation is easy (Figure 4). Indeed, from (7), one has ${}^e\tau_e$ from the control joint velocities $\dot{\mathbf{q}}$. Using

$${}^b\tau_e = \begin{pmatrix} {}^b\mathbf{R}_e & \mathbf{0} \\ \mathbf{0} & {}^b\mathbf{R}_e \end{pmatrix} {}^e\tau_e, \tag{29}$$

and noting ${}^b\mathbf{J}_e^{inv} = \begin{pmatrix} {}^b\mathbf{R}_e & \mathbf{0} \\ \mathbf{0} & {}^b\mathbf{R}_e \end{pmatrix} {}^e\mathbf{J}_e^{inv}$, one finally has

$${}^b\tau_e = {}^b\mathbf{J}_e^{inv^{-1}}\dot{\mathbf{q}} \tag{30}$$

which is integrated as above.

Consequently, the vision-based inverse kinematics of the hexapod *expressed in the end-effector frame* are used for simulation, while *expressed in the base frame* they are used for control.

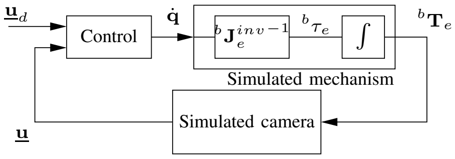

Fig. 4. Simulation scheme

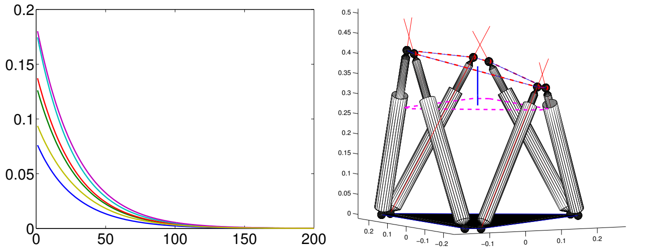

Fig. 5. **Left:** Errors on each leg $\mathbf{e}_i^T\mathbf{e}_i$, **Right:** Trajectory in space with initial (magenta, dashed) and desired (red, dash-dotted) position of the platform

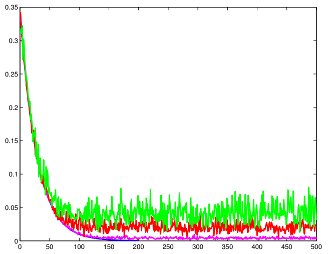

Fig. 6. Robustness to noise : sum of squares of the errors E^TE *vs* time, with a noise amplitude of 0.01 deg (dashed), 0.05 deg (dashed) and 0.1 deg (dash-dotted).

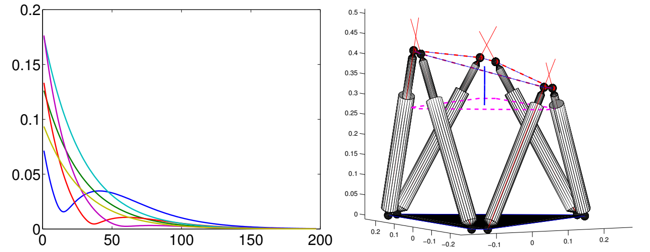

Fig. 7. Lucky convergence in the case where legs 2, 4 and 6 only are used for control.

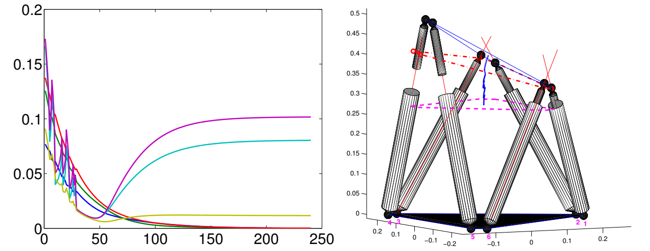

Fig. 8. Non-convergence in the case where legs 1, 2 and 3 only are used for control. The direction of the legs are superimposed (red) on the cylinders. A leg has converged to its desired orientation if its direction crosses the end-effector in the desired pose at the joint location. Notice that this happens only for the 3 controlled legs.

B. Control simulation

We simulated a commercial DeltaLab hexapod, such that ${}^{b}\mathbf{A}_{2k} = R_b \begin{pmatrix} \cos(k\frac{\pi}{3}+\alpha) \\ \sin(k\frac{\pi}{3}+\alpha) \\ 0 \end{pmatrix}$, ${}^{b}\mathbf{A}_{2k+1} = R_b \begin{pmatrix} \cos(k\frac{\pi}{3}-\alpha) \\ \sin(k\frac{\pi}{3}-\alpha) \\ 0 \end{pmatrix}$, ${}^{e}\mathbf{B}_{2k} = R_e \begin{pmatrix} \cos(k\frac{\pi}{3}+\beta) \\ \sin(k\frac{\pi}{3}+\beta) \\ 0 \end{pmatrix}$, ${}^{e}\mathbf{B}_{2k+1} = R_e \begin{pmatrix} \cos(k\frac{\pi}{3}-\beta) \\ \sin(k\frac{\pi}{3}-\beta) \\ 0 \end{pmatrix}$, $k \in \{0, 1, 2\}$ with $R_b = 270mm$, $\alpha = 4.25°$, $R_e = 195mm$, $\beta = 5.885°$ and the legs range are $[345mm, 485mm]$.

In all the simulations presented here, the initial configuration of the platform is the reference configuration where all the legs have minimal length. The goal configuration is obtained from this reference configuration by a translation by $10cm$ along the z axis of the platform (upward vertical) and a rotation of $15°$ around the x axis, thus reaching the workspace limit.

In a first simulation, all the legs are used for control. Figure 5 shows that the errors on each leg converge exponentially to 0 and that the desired end-effector pose is reached.

We also added some noise to the simulation (Figure 6). To do so, since it is hard to characterize the noise in the image, we applied, as a first guess, a random rotation on the unit vectors $\underline{\mathbf{u}}_i$ with maximal amplitude of 0.01, 0.05 and 0.1°. Results show a potentially good robustness.

In a second simulation, only the legs 2, 4 and 6 were used for control, to show that 3 legs *may* be enough to reach the desired configuration. Figure 7 shows that convergence is ensured even if the error on legs 1, 3 and is not controlled.

In a third simulation, only the legs 1, 2 and 3 were used. Figure 8 shows that the Necessary condition 1 is not a sufficient condition: the controlled legs converge towards their desired orientation, while neither the non-controlled legs nor the end-effector reach their goal. Notice that we deliberately removed in the simulation any joint limit to show the theoretical behaviour of the control with only 3 legs.

In both the second and third simulations, the interaction matrix is of full rank and does not yield degenerate control. However, in both case, defining the orientation of 3 legs is not enough to uniquely determine the end-effector pose. The second simulation shall be considered as a lucky trial, which can be explained by the fact that the motion of the 3 controlled legs between the initial and desired configuration pulls the other 3 legs towards their desired direction. Nevertheless, this result is interesting since it suggests that self-occlusions of the legs, when using a single fixed camera, should only have a local impact on the occluded leg.

V. Conclusion

We proposed a novel method for controlling a parallel mechanism using vision as a redundant sensor, adding a proprioceptive nature to the usual exteroceptive nature of vision. It was validated and illustrated on a Gough-Stewart platform simulation, showing a probably large convergence domain and potentially good robustness properties.

However, this paper is only the seed of a vast research domain. Indeed, there are several points to be addressed before a safe and satisfying implementation can be made on a real platform. First, this control does not take into account joint limit avoidance. This point is fundamental since these limits can be easily be reached and their avoidance may not be as trivial as for serial mechanisms. Second, this control assumes a detection of cylinder edges, which is known to be delicate in vision. Third, since the control is essentially based on the direction of each leg, one may think of extracting it from the image of a generally shaped leg. Fourth, the convergence seems potentially global but has not been proven, nor robustness and controllability have been thoroughly studied. Finally, one would like to apply this control to any parallel mechanism.

Acknowledgment

This study was jointly funded by CPER Auvergne 2003-2005 program and by the CNRS-ROBEA MP2 project.

References

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

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

[3] Http://www-sop.inria.fr/coprin/equipe/merlet.

[4] J.-P. Merlet, "An algorithm for the forward kinematics of general 6 d.o.f. parallel manipulators," INRIA, Tech. Rep. 1331, Nov. 1990.

[5] M. Husty, "An algorithm for solving the direct kinematics of general Gough-Stewart platforms," *Mech. Mach. Theory*, vol. 31, no. 4, pp. 365–380, 1996.

[6] X. Zhao and S. Peng, "Direct displacement analysis of parallel manipulators," *Journal of Robotics Systems*, vol. 17, no. 6, pp. 341–345, 2000.

[7] H. Kim and L.-W. Tsai, "Evaluation of a Cartesian parallel manipulator," in *Advances in Robot Kinematics: Theory and Applications*, J. Lenarčič and F. Thomas, Eds. Kluwer Academic Publishers, June 2002.

[8] G. Gogu, "Fully-isotropic T3R1-type parallel manipulator," in *On Advances in Robot Kinematics*, J. Lenarčič and C. Galletti, Eds., 2004, vol. Kluwer Academic Publishers, pp. 265–272.

[9] L. Baron and J. Angeles, "The on-line direct kinematics of parallel manipulators under joint-sensor redundancy," in *Advances in Robot Kinematics*, 1998, pp. 126–137.

[10] F. Marquet, "Contribution à l'étude de l'apport de la redondance en robotique parallèle," Ph.D. dissertation, LIRMM - Univ. Montpellier II, Oct. 2002.

[11] D. DeMenthon and L. Davis, "Model-based object pose in 25 lines of code," *Lecture Notes in Computer Science*, pp. 335–343, 1992.

[12] J. Lavest, M. Viala, and M. Dhome, "Do we really need an accurate calibration pattern to achieve a reliable camera calibration," in *Proceedings of ECCV98*, Freiburg, June 1998, pp. 158–174.

[13] N. Andreff, P. Renaud, P. Martinet, and F. Pierrot, "Vision-based kinematic calibration of an H4 parallel mechanism: practical accuracies," *Industrial Robot: An international journal*, vol. 31, no. 3, pp. 273–283, May 2004.

[14] B. Dasgupta and T. Mruthyunjaya, "Force redundancy in parallel manipulators: theoretical and practical issues," *Mech. Mach. Theory*, vol. 33, no. 6, pp. 727–742, 1998.

[15] B. Espiau, F. Chaumette, and P. Rives, " A New Approach To Visual Servoing in Robotics," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 3, June 1992.

[16] P. Martinet, J. Gallice, and D. Khadraoui, "Vision based control law using 3d visual features," in *Proc. World Automation Congress, WAC'96, Robotics and Manufacturing Systems*, vol. 3, May 1996, pp. 497–502.

[17] R. Horaud, F. Dornaika, and B. Espiau, "Visually Guided Object Grasping," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 4, pp. 525–532, 1998.

[18] H. Christensen and P. Corke, Eds., *Int. Journal of Robotics Research – Special Issue on Visual Servoing*, vol. 22, no. 10/11, Oct. 2003.

[19] M. 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*, Edimburg, Scotland, July 1998, pp. 109–116.

[20] H. Kino, 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)*, Tokyo, Japan, Oct. 25-27 1999, pp. 545–550.

[21] P. Kallio, Q. Zhou, and H. N. Koivo, "Three-dimensional position control of a parallel micromanipulator using visual servoing," in *Microrobotics and Microassembly II, Proceedings of SPIE*, B. J. Nelson and E. Jean-Marc Breguet, Eds., vol. 4194, Boston, USA, Nov. 2000, pp. 103–111.

[22] V. Gough and S. Whitehall, "Universal tyre test machine," in *Proc. FISITA 9th Int. Technical Congress*, May 1962, pp. 117–137.

[23] D. Stewart, "A platform with six degrees of freedom," in *Proc. IMechE (London)*, vol. 180, no. 15, 1965, pp. 371–386.

[24] P. Renaud, N. Andreff, G. Gogu, and P. Martinet, "On vision-based kinematic calibration of a Stewart-Gough platform," in *11th World Congress in Mechanism and Machine Science (IFToMM2003)*, Tianjin, China, April 1-4 2004, pp. 1906–1911.

[25] N. Andreff, B. Espiau, and R. Horaud, "Visual servoing from lines," *Int. Journal of Robotics Research*, vol. 21, no. 8, pp. 679–700, Aug. 2002.

[26] R. Mahony and T. Hamel, "Visual servoing using linear features for under-actuated rigid-body dynamics," in *IEEE/RSJ Int. Conf. on Intelligent Robots Systems*, Hawaii, USA, 2001.

[27] E. Malis, J. Borrelly, and P. Rives, "Intrinsics-free visual servoing with respect to straight lines," in *IEEE/RSJ Int. Conf. on Intelligent Robots Systems*, Lausanne, CH, Oct. 2002.

[28] D. Khadraoui, R. Rouveure, C. Debain, P. Martinet, P. Bonton, and J. Gallice, "Vision based control in driving assistance of agricultural vehicles," *Int. Journal of Robotics Research*, vol. 17, no. 10, Oct. 1998.

[29] J. Stolfi, *Oriented Projective Geometry*. Academic Press, 1991.

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

[31] H. Pottmann, M. Peternell, and B. Ravani, "Approximation in line space – applications in robot kinematics and surface reconstruction," in *Advances in Robot Kinematics: Analysis and Control*, J. Lenarčič and M. L. Husty, Eds. Kluwer Academic Publishers, 1998, pp. 403–412.

[32] A. Marchadier, "Commande cinématique référencée vision d'un robot parallèle de type plateforme de Gough," Master's thesis, Univ. B. Pascal, Clermont-Ferrand, 2004.

[33] F. Chaumette, *La commande des robots manipulateurs, Traité IC2*. Hermès, 2002, ch. Asservissement visuel, pp. 105–150.