2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

Minimal Representation for the Control of Gough-Stewart Platforms via Leg Observation Considering a Hidden Robot Model

Sébastien Briot[1] and Philippe Martinet[2]

Abstract— **This paper presents new insights about the sensor-based control of Gough-Stewart (GS) platforms. Previous works have shown that it was possible to control the GS platform by observing its legs directions instead of using the encoders values or the measurement of the platform pose. It was demonstrated that observing only three legs directions was enough for the control but no physical explanations were given. Moreover, sometimes, the GS platform was not converging to the desired pose and the reasons of these divergences were not disclosed.**

This paper aims at answering to this two opened problems. It is shown that observing three leg directions involves controlling the displacement of a hidden robot whose models differs from those of the usual GS platform. This robot has assembly modes and singular configurations different from those of the GS platform. This involves that the legs to observe should be chosen carefully in order to avoid inaccuracy problems. In this sense, the accuracy analysis of the new robot is performed to show the importance of the leg selection. All these results are validated on a GS platform simulator created using ADAMS/Controls and interfaced with Matlab/Simulink.

I. INTRODUCTION

Parallel robots are claimed to have superior skills than serial robots: they can reach high-speeds, show high-dynamic performances and achieve good repeatability [1]. However, their control is troublesome because of the complex mechanical structure, highly coupled joint motions due to the closed-loop kinematic chains and many factors such as clearances in passive joints, assembly errors, etc., which degrade stability and accuracy.

There exists a large amount of work on the control of parallel mechanisms (see [2] for a long list of references). In the focus of attention, Cartesian control is naturally achieved through the use of the inverse differential kinematic model which transforms Cartesian velocities into joint velocities. It is noticeable that the inverse differential kinematic model 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.

In this sense, recent research works have proved that vision is an effective way to estimate the end-effector pose. The most common approach consists of the direct observation of the end-effector pose [3], [4], [5]. However, 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 machine-tool. On the opposite, it is generally not a problem to observe the legs of the mechanism which are most often designed with slim and rectilinear legs [2].

A first step in this direction was made in [6] were vision was used to derive a visual servoing scheme based on the observation of a Gough-Stewart (GS) parallel robot [7]. In that method, the legs direction were chosen as visual primitives and control was derived based on their reconstruction from the image. In this work, it has also been shown that the GS platform can be controlled using the observation of only three leg directions (arbitrarily chosen among its six legs). However, in some cases, the GS platform do not converge to the desired end-effector pose (even if the observed leg directions did).

Since these first works, it has been proven that the leg direction observation can be used for calibration [8] and computed torque control schemes [9]. But, two questions are still not answered:

1) why is it possible to control the GS platform using the observation of only three leg directions?
2) why does the end-effector not systematically converge to the desired pose, even if the observed leg directions do?

This paper aims at answering to these two fundamental questions. In this sense, it is divided as follows: the next section makes some brief recalls on the servoing using leg observation for parallel robots. Section III introduces a new robot model that intrinsically arise from this way of controlling the GS platform. The robot architecture is described as well as its singularities that are different from those of the GS platform. In section IV, the accuracy analysis of this new robot is performed. It is shown that the accuracy directly depends on the legs chosen for the control. Section V presents simulation results and, finally, in section VI, conclusions are drawn.

II. RECALLS ON CONTROL USING LEG OBSERVATION

This part aims at presenting brief recalls on the servoing of the GS platform using leg observation.

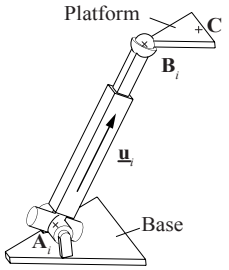
A. Kinematics of the GS platform

Consider the GS platform in Fig. 1. It has six *UPS* legs of varying length $q_i, i \in 1...6$, (in the remainder of the paper, *U* (*U*, resp.), *P* (*P*, resp.) and *S* will stand for passive (active, resp.) universal, prismatic and spherical joints, respectively) attached to the base by *U* joints located in points $\mathbf{A}_i$ and

[1]S. Briot is with the French CNRS and the Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN), 44321 Nantes France (Sebastien.Briot@irccyn.ec-nantes.fr)

[2]P. Martinet is with the LUNAM, Ecole Centrale de Nantes and the IRCCyN, 44321 Nantes France (Philippe.Martinet@irccyn.ec-nantes.fr)

(a) picture of the robot



(b) schematics of leg i

Fig. 1. A GS platform from DeltaLab.

to the moving platform (end-effector) by S joints located in points $\mathbf{B}_i$. The inverse kinematic model is given by:

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

expressing that q_i is the length of vector $\overrightarrow{\mathbf{A}_i\mathbf{B}_i}$, i.e.

$$\overrightarrow{\mathbf{A}_i\mathbf{B}_i} = q_i\underline{\mathbf{u}}_i \text{ , or also } \mathbf{A}_i + q_i\underline{\mathbf{u}}_i = \mathbf{B}_i \tag{2}$$

where $\underline{\mathbf{u}}_i$ is the unit vector of the line passing through points $\mathbf{A}_i$ and $\mathbf{B}_i$.

From [1], the inverse Jacobian of the GS platform, relating the end-effector twist τ_e to the joint velocities is given by

$$\dot{\mathbf{q}} = \mathbf{J}_e^{inv}\tau_e, \text{ with } \mathbf{J}_e^{inv} = \begin{bmatrix} \underline{\mathbf{u}}_1^T & \left(\overrightarrow{\mathbf{CB}_1} \times \underline{\mathbf{u}}_1\right)^T \\ \vdots & \vdots \\ \underline{\mathbf{u}}_6^T & \left(\overrightarrow{\mathbf{CB}_6} \times \underline{\mathbf{u}}_6\right)^T \end{bmatrix} \tag{3}$$

where $\mathbf{C}$ is the center of the end-effector reference frame $\mathcal{R}_e$.

B. Kinematics of the GS platform using leg observation

The approach presented in [6] proposed to estimate the vectors $\underline{\mathbf{u}}_i$ using a camera. The estimation of $\underline{\mathbf{u}}_i$ is not disclosed in this paper. For more information, the reader should refer to the previously cited work. If this camera is fixed on the ground, then the reference frame associated to it is, without loss of generality, the base frame $\mathcal{R}_b$. As a result, the kinematics of the GS platform do not express as simply as in the end-effector embedded camera case. Indeed, expressed in the base frame, (3) becomes:

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

where $\overrightarrow{{}^b\mathbf{C}{}^b\mathbf{B}_i} = {}^b\mathbf{R}_e\,{}^e\mathbf{B}_i$, with ${}^b\mathbf{R}_e$ the rotation matrix between the base and end-effector frames and ${}^e\mathbf{B}_i$ the position of point $\mathbf{B}_i$ in the end-effector frame. Hence, it is necessary to estimate the end-effector orientation with respect to the base frame which is uncommon.

An alternate formulation was proposed in [6], which is well suited to visual servoing using leg observation. It consists in considering the mechanism in its dual operating mode: the end-effector is fixed and the base moves with respect to it (Fig. 2). Thus, we are interested in the inverse Jacobian relating the base twist ${}^b\tau_b$ expressed in the base frame to the joint velocities.

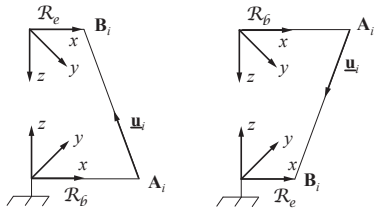


Fig. 2. Duality between the mobile end-effector mode and the fixed end-effector mode.

By analogy with (4), i.e. by permutation of the roles of $\mathbf{B}_i$ and $\mathbf{A}_i$ and of $\mathcal{R}_e$ and $\mathcal{R}_b$ (Fig. 2), one obtains the vision-based kinematics of the GS platform expressed in the base frame [6]:

$$\dot{\mathbf{q}} = {}^b\mathbf{J}_b^{invb}\tau_b, \text{ with } {}^b\mathbf{J}_b^{inv} = -\begin{bmatrix} {}^b\underline{\mathbf{u}}_1^T & {}^bh_1\,{}^b\underline{\mathbf{h}}_1^T \\ \vdots & \vdots \\ {}^b\underline{\mathbf{u}}_6^T & {}^bh_6\,{}^b\underline{\mathbf{h}}_6^T \end{bmatrix} \tag{5}$$

where ${}^bh_i\,{}^b\underline{\mathbf{h}}_i = {}^b\mathbf{A}_i \times {}^b\underline{\mathbf{u}}_i = {}^b\mathbf{B}_i \times {}^b\underline{\mathbf{u}}_i$.

C. Control scheme and interaction matrix

Visual servoing is based on the so-called interaction matrix $\mathbf{L}^T$ [10] 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:

$$\dot{s} = \mathbf{L}_{(s)}^T T_c \tag{6}$$

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

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:

$$T_c = \lambda\hat{\mathbf{L}}_{(s)}^{T+}e(s, s_d) \tag{7}$$

where T_c is used as a pseudo-control variable and the upperscript $+$ corresponds to the matrix pseudo-inverse.

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:

$$\mathbf{e}_i = {}^b\underline{\mathbf{u}}_i \times {}^b\underline{\mathbf{u}}_{di} \tag{8}$$

where ${}^b\underline{\mathbf{u}}_{di}$ is the desired value of ${}^b\underline{\mathbf{u}}_i$.

It is then necessary to relate the base twist ${}^b\tau_b$ to the derivative of ${}^b\underline{\mathbf{u}}_i$ with respect to time. From [6], it comes that:

$${}^b\dot{\underline{\mathbf{u}}}_i = \mathbf{M}_i^{Tb}\tau_b \tag{9}$$

$$\mathbf{M}_i^T = -\frac{1}{q_i}\left(\mathbf{I}_3 - {}^b\underline{\mathbf{u}}_i\,{}^b\underline{\mathbf{u}}_i^T\right)\left[\mathbf{I}_3 \quad -\left[{}^b\mathbf{A}_i + q_i\,{}^b\underline{\mathbf{u}}_i\right]_\times\right] \quad (10)$$

where $[...]_\times$ is the antisymetric matrix associated to a 3D vector [5].

It can be proven that matrix $\mathbf{M}_i$ is of dimension 2 [6]. As a result, a minimum of three independent legs is necessary to control the end-effector pose. An interaction matrix $\mathbf{M}^T$ can then obtained by stacking the matrices $\mathbf{M}_i^T$ of k legs ($k = 3...6$).

Finally, a control is chosen a control such that E, the vector stacking the errors $\mathbf{e}_i$ associated to of k legs ($k = 3...6$), decreases exponentially, i.e. such that

$$\dot{E} = -\lambda E \quad (11)$$

Then, introducing $\mathbf{N}_i^T = -\left[{}^b\underline{\mathbf{u}}_{di}\right]_\times \mathbf{M}_i^T$, the combination of (8), (9) and (11) gives

$${}^b\tau_b = -\lambda \mathbf{N}^{T+} E \quad (12)$$

where $\mathbf{N}^T$ can be obtained by obtained by stacking the matrices $\mathbf{N}_i^T$ of k legs ($k = 3...6$).

This expression can be transformed into the control joint velocities using (5):

$$\dot{\mathbf{q}} = -\lambda\,{}^b\mathbf{J}_b^{inv}\mathbf{N}^{T+} E \quad (13)$$

In the next section, it is shown that observing the displacement of the leg directions ${}^b\underline{\mathbf{u}}_i$ is intrinsically equivalent to controlling another robot, different from the GS platform.

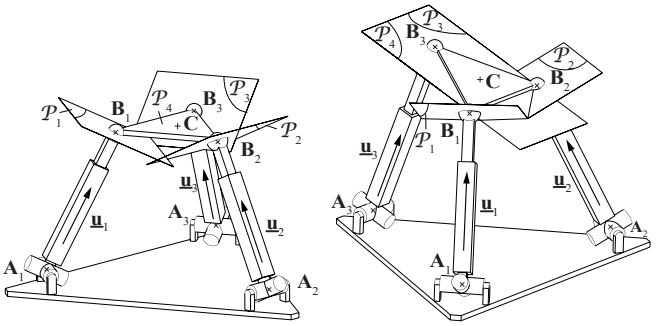
III. DESCRIPTION OF THE HIDDEN ROBOT MODEL

A. Description of the hidden robot architecture

In the classical control approach, the encoders measure the motion of the actuator. In the previously described control approach [5], the leg directions are observed. So, in a reciprocal manner, one could wonder to what kind of virtual actuators this observation corresponds.

For answering to this question, let us analyse the leg i (Fig. 1(b)). Its unit vector ${}^b\underline{\mathbf{u}}_i$ can obviously be parameterized by two independent coordinates that can be the angles defined by the *U* joint rotations. Thus, ${}^b\underline{\mathbf{u}}_i$ is a measure of the *U* joint displacements. As a result, the *U* joint is the virtual actuator we were looking for. Observing the directions of the leg remains not to control the displacement of a *UPS* leg but of a virtual *UPS* leg with the same geometric properties as the real leg.

It is well known in the parallel robot community that a 3-*UPS* robot (Fig. 3(a)) is fully-actuated. Therefore, this is the reason why it is possible to control the GS platform by observing the displacements of three of its six legs. This is equivalent to actuate a virtual 3-*UPS* robot with the same geometric properties as the GS platform (same attachment points, leg length, *U* and *S* joint orientations), but with assembly modes and singular configurations that differ from those of the GS platform. They should be studied in order to avoid control problems.



(a) schematics

(b) example of a Type 2 singularity: the platform get an uncontrollable rotation around $\mathbf{B}_1\mathbf{B}_2$.

Fig. 3. A 3-*UPS* robot.

B. Forward geometric analysis

Cross multiplying the right part of (2) by ${}^b\underline{\mathbf{u}}_i$ leads to:

$${}^b\underline{\mathbf{u}}_i \times \left({}^b\mathbf{A}_i + q_i\,{}^b\underline{\mathbf{u}}_i\right) = {}^b\underline{\mathbf{u}}_i \times {}^b\mathbf{A}_i = {}^b\underline{\mathbf{u}}_i \times {}^b\mathbf{B}_i \quad (14)$$

which are the equations to solve for obtaining the symbolic expressions for the 3-*UPS* robot forward geometric model. However, solving this problem is tedious and, for reasons of paper compactness, will not be developed here. Instead of this, a geometric and qualitative approach is used in order to better understand the forward geometric problem of this robot.

Without loss of generality, let us consider that we analyze the 3-*UPS* robot depicted at Fig. 3(a). If the leg 3 is disassembled at point $\mathbf{B}_3$, as there are only four actuators for controlling the six robot mobilities, the platform gains two degrees-of-freedom. The gained motion is called a *spatial Cardanic motion* [11]. This motion is defined by the fact that the points $\mathbf{B}_1$ and $\mathbf{B}_2$ are constrained to move on the lines of which directions are given by ${}^b\underline{\mathbf{u}}_1$ and ${}^b\underline{\mathbf{u}}_2$, respectively, and the platform is free to rotate around the line $\mathbf{B}_1\mathbf{B}_2$. As demonstrated in [11], the surface described by point $\mathbf{B}_3$ is an octic surface, i.e. an algebraic surface of degree eight.

As $\mathbf{B}_3$ also belongs to leg 3, this point is constrained to move on a line defined by the direction ${}^b\underline{\mathbf{u}}_3$ of the passive prismatic joint. As shown in [11], a line and an octic surface can have up to eight real intersection points. As a result, the 3-*UPS* robot can have up to eight assembly modes. Let us recall here that, in the general case, the GS platform can have up to 40 assembly modes [1] that are different from those of the 3-*UPS* robot.

The existence of these assembly modes explains the second question presented in the introduction, i.e. the non systematic convergence of the end-effector to the desired pose, even if the observed leg directions do. A numerical example of this phenomenon will be presented in the section V.

C. Singularity analysis

Three types of singular configurations may appear for robots with six degrees of freedom [12]:

1) the Type 1 singularities where the robot loses of at least one degree of freedom;
2) the Type 2 singularities where there is the apparition of some uncontrollable motions. These singularities are the worst type parallel robots can meet because, in their neighborhood, the platform accuracy considerably decreases;
3) the Type 3 singularities, where both Type 1 and Type 2 singularities encounter.

The singular configurations of the 3-*UPS*-like robot have been deeply studied in the past [13], [14]. Type 1 singularities appear if one leg length q_i is equal to 0 (this is the same condition for the GS platform). In this case, the leg i can no more produce a motion in the directions normal to ${}^b\underline{\mathbf{u}}_i$. Type 2 singularities appear when the planes $\mathcal{P}_1$, $\mathcal{P}_2$, $\mathcal{P}_3$ (whose normal directions are defined by the vectors ${}^b\underline{\mathbf{u}}_1$, ${}^b\underline{\mathbf{u}}_2$ and ${}^b\underline{\mathbf{u}}_3$, respectively) and the plane $\mathcal{P}_4$ (passing through the points $\mathbf{B}_1$, $\mathbf{B}_2$ and $\mathbf{B}_3$) intersect in one point (that can be at infinity) (Fig. 3(b)).

Obviously, the singularity loci vary depending on the leg chosen for the GS platform control. Therefore, it is extremely important, for having the best performances of the controller, to make an optimal selection of the three legs to observe. This is the topic of the next section.

Finally, it should be mentioned that the singularities of the 3-*UPS* robot are not physical singularities, in the sense that they do not leads to uncontrollable free motions of the platform. However, they are representation singularities due to the mapping from the Cartesian space to the leg direction space [15].

IV. SELECTION OF THE CONTROLLED LEGS

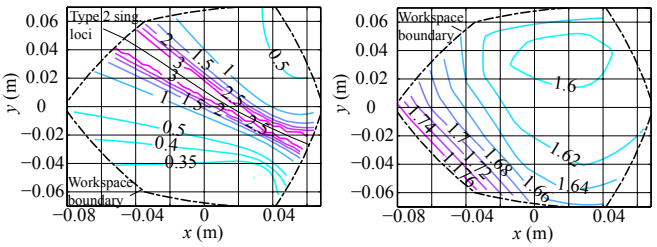
Several indices can be used for characterizing the neighborhood of singularities (e.g. the condition number, the dexterity [16], the pressure angle [17], etc.). Here, as generally, the visual servoing is used for improving the robot accuracy, it is proposed to use it as an index for the characterization of singularity proximity.

A. Accuracy analysis

From 9, and using the first order approximation of the forward geometric model [16], it is possible to write

$$\delta\underline{\mathbf{p}} = \mathbf{M}^{T+}\delta\underline{\mathbf{u}} \tag{15}$$

where $\delta\underline{\mathbf{p}} = [\delta\underline{\mathbf{x}}^T, \delta\underline{\omega}^T]^T$ is the platform pose error composed of the positioning error $\delta\underline{\mathbf{x}}$ and the orientation error $\delta\underline{\omega}$, $\delta\underline{\mathbf{u}}$ is the error on the observation of the leg direction, and $\mathbf{M}^{T+}$ is the pseudo-inverse of the matrix $\mathbf{M}^T$ that can be obtained by stacking the matrices $\mathbf{M}_i^T$ of the three observed legs. Obviously, this matrix is the Jacobian matrix of the equivalent 3-*UPS* robot and, as a result, will degenerate near the singularity configurations presented in section III.C. It should be mentioned here that it is decided to use a simple model for computing the robot accuracy, but any other more complicated models can be used (e.g. models that take into account flexibilities [18], clearances [19], etc.). However, this simple model is enough for our demonstration.



(a) when legs 1,2,5 are observed (b) when legs 1,3,5 are observed

Fig. 4. Maximal pose error (in mm) for z = 0.4 m when the platform is at zero orientation.

In the remainder of the paper, the GS platform of DeltaLab is studied (Fig.1). This robot has the following characteristics:

$${}^b\mathbf{A}_{2k} = R_b \begin{pmatrix} \cos\left(k\frac{\pi}{3} - \alpha\right) \\ \sin\left(k\frac{\pi}{3} - \alpha\right) \\ 0 \end{pmatrix}, {}^b\mathbf{A}_{2k+1} = R_b \begin{pmatrix} \cos\left(k\frac{\pi}{3} + \alpha\right) \\ \sin\left(k\frac{\pi}{3} + \alpha\right) \\ 0 \end{pmatrix}$$

$${}^b\mathbf{B}_{2k} = R_e \begin{pmatrix} \cos\left((2k+1)\frac{\pi}{3} - \beta\right) \\ \sin\left((2k+1)\frac{\pi}{3} - \beta\right) \\ 0 \end{pmatrix}, {}^b\mathbf{B}_{2k+1} = R_e \begin{pmatrix} \cos\left((2k+1)\frac{\pi}{3} + \beta\right) \\ \sin\left((2k+1)\frac{\pi}{3} + \beta\right) \\ 0 \end{pmatrix}$$

where $k \in \{0, 1, 2\}$, R_b = 0.27m, R_e = 0.195m, α=4.25 deg. and β=5.885 deg. Moreover, the legs range are [0.345 m, 0.485 m].

For this mechanism, and for an error $\delta\underline{\mathbf{u}}_i$ defined such that the vector ${}^b\underline{\mathbf{u}}_i$ is contained in a cone of axis ${}^b\underline{\mathbf{u}}_{i0}$ and of half angle ϕ_i (${}^b\underline{\mathbf{u}}_{i0}$ is the nominal value of ${}^b\underline{\mathbf{u}}_i$ and, in what follows, ϕ_i is taken equal to 0.01 deg. for each leg direction), let us compute the maximal positioning error when only three of its six legs are observed. Twenty different combinations are possible. However, the value of the error for only two of them (when legs 1,2,5 and 1,3,5 are observed) is plotted at Fig. 4. On Fig. 4(a), it is possible to note that the maximal error varies very quickly, especially near the singularity loci. On Fig. 4(b), things are different. The variation of the accuracy is very smooth. Thus, it can be concluded that the selection of the legs to observe is crucial for the final pose accuracy.

B. Discussion

The previous section showed the importance of the legs chosen for the control scheme. Several questions naturally arise here. The first one concerns the number of legs to observe. In terms of accuracy, it is obvious that observing four, five or six legs, i.e. adding measurement redundancy, will improve the pose accuracy of the robot. However, increasing the number of legs to observe leads to an increase of the computational time and may be applied with difficulty when high sampling periods are required. Thus, a compromise must be found between the sampling period and the computational time for any given application.

The second question is about the selection of the legs to observe. With only three legs among six to observe, as mentioned above, twenty different 3-*UPS* robots can be defined. What is thus the best virtual robot model to use?

If the control law proposed in section II is applied, it is first necessary to guaranty that, for the used set of legs:

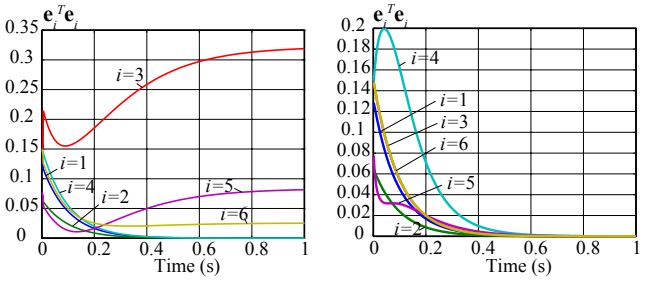
- obviously, the legs must be observable during the whole robot displacement.
- the initial and final robot configurations must be included in the same assembly mode of the virtual 3-*UPS* robot. If not, the controller will not be able to converge to the desired end-effector pose, even if the observed leg directions do. In this last case, the problem can be solved by applying special trajectories that cross Type 2 singularities [20] or encircle a cusp point [21].

Then, if accuracy is needed, the leg selection must guaranty the best final accuracy. To achieve this goal, the following procedure can be used:

1) knowing the six leg orientations at the initial and final GS platform configurations, compute the solutions of the forward geometric model of the twenty 3-*UPS* robots,
2) find, using a procedure similar to the one proposed in [22] for all virtual 3-*UPS* robots, the solutions of the forward geometric model that belong to the same assembly modes; if, for one given virtual robot, initial and final platform configurations do not belong to the same assembly mode, discard it; if it does not exist any 3-*UPS* robot for which initial and final configurations belong to the same assembly mode, the displacement is not feasible, except if special trajectories are planned as mentioned previously,
3) for all remaining virtual 3-*UPS* robots, knowing the observation error $\delta\mathbf{u}$, compute the positionning error using (15); retain the set of legs that guaranty the best accuracy;
4) test the controller (in simulation) with the retained set of legs; if there is no problem of convergence and that the legs are observable during the whole displacement, the problem is solved; if not, discard this set of leg and redo point 3; if it does not exist any 3-*UPS* robot for which initial and final configurations belong to the same assembly mode, the displacement is not feasible, except if special trajectories are planned as mentioned previously.

Obviously, this methodology can be extended when four or five legs are observed. One should also be aware that instead of given the initial and final robot configurations to the controller, it is better to define a trajectory between these two points in order to avoid crossing singularities inadvertently.

Finally, it is considered in this paper that the sensor measurement space is the same as the leg direction space. However, for example using a camera, the legs directions are not directly measured but rebuilt from the observation of the legs limbs projection in the 2D camera space [6]. Thus, for the leg reconstruction, the mapping between the camera space and the real 3D space is involved, and it is not free of singularities (see [23] for an example of mapping singularities). In the neighboorhood of mapping singularities, the robot accuracy will also tend to decrease. As a result, this mapping should be considered in the accuracy computation and in the selection of the legs to observe.



(a) when legs 1,2,4 are observed (b) when legs 2,3,6 are observed

Fig. 5. Error on each leg $\mathbf{e}_i^T\mathbf{e}_i$.

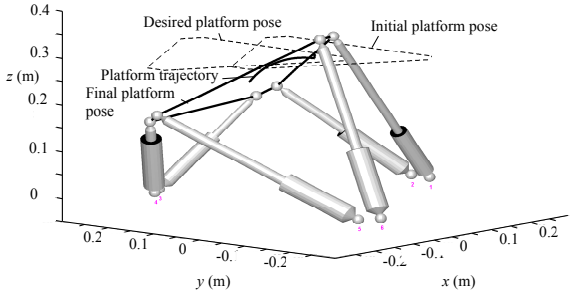


Fig. 6. Trajectory in space with initial, desired and final platform positions.

V. SIMULATION RESULTS

In this section, simulations are performed on an Adams mockup of the DeltaLab GS platform. This virtual mockup is connected to Matlab/Simulink via the module Adams/Controls. The controller presented in section II is applied with a value of λ assigned to 5. The legs ranges are not considered to show the theoretical behaviour of the robot.

In the first simulations, the initial platform pose is equal to $\{x = 0m, y = 0m, z = 0.3m, q_{r1} = 1, q_{r2} = 0, q_{r3} = 0, q_{r4} = 0\}$ and the final platform pose is set to $\{x = -0.1m, y = 0.1m, z = 0.3m, q_{r1} = 1, q_{r2} = 0, q_{r3} = 0, q_{r4} = 0\}$ where $q_{r1}, q_{r2}, q_{r3}, q_{r4}$ are the quaternions characterizing the platform rotations between $\mathcal{R}_e$ and $\mathcal{R}_b$ [24]. For going from the initial point to the final ones, two sets of observed legs directions are tested: $\{1, 2, 4\}$ and $\{2, 3, 6\}$. The results for the convergence of the legs directions are presented in Fig. 5. It can be shown that when the legs $\{2, 3, 6\}$ are observed, all leg directions converge to 0. For the other case, the non observed legs do not reach their desired pose. Looking at the platform pose computed by ADAMS, the robot reach the configuration $\{x = -0.066m, y = 0.090m, z = 0.239m, q_{r1} = -0.931, q_{r2} = 0.290, q_{r3} = 0.101, q_{r4} = 0.197\}$ (Fig 6). Solving the forward geometric problem using (14) at the final desired robot configuration for legs $\{1, 2, 4\}$, it can be demonstrated that two real assembly modes exist that are $\{x = -0.1m, y = 0.1m, z = 0.3m, q_{r1} = 1, q_{r2} = 0, q_{r3} = 0, q_{r4} = 0\}$ and $\{x = -0.066m, y = 0.090m, z = 0.239m, q_{r1} = -0.931, q_{r2} = 0.290, q_{r3} = 0.101, q_{r4} = 0.197\}$. This validates the theory presented in section III.B.

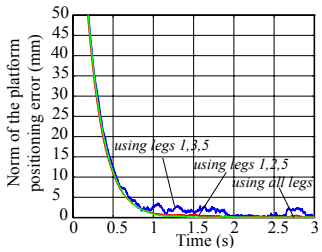


Fig. 7. Simulated platform pose error (in mm).

In the second simulation, the final point is changed to $\{x = 0m, y = -0.06m, z = 0.4m, q_{r1} = 1, q_{r2} = 0, q_{r3} = 0, q_{r4} = 0\}$ and a random noise of 0.01 deg is added on the simulated measure of the leg directions. To show the importance of the leg selection on the robot accuracy, it is decided to control the robot displacement using three different sets of legs: (i) legs $\{1, 2, 5\}$, (ii) legs $\{1, 3, 5\}$ and (iii) all legs. The results (Fig. 7) show that, as presented in Fig. 4, the final platform pose accuracy is better when legs $\{1, 2, 5\}$ are observed (around 0.3 mm) than with legs $\{1, 3, 5\}$ (around 1.7 mm). When all legs are observed, the final pose error is much lower than when only three legs are observed. But, as mentioned in the previous section, the computational time is higher.

VI. CONCLUSIONS

This paper presented new insights about the sensor-based control of GS platforms. It has been shown that observing the leg directions of the GS platform involves controlling the displacement of a hidden robot which is based on *UPS* legs instead of *UPS* legs. If only three legs are observed, the new equivalent robot becomes a 3-*UPS* which has six fully-controlled DOFs. Its has been demonstrated that this robot has up to eight assembly modes that are different from those of the GS platform. Its conditions of singularities have been presented. It has been shown that the legs to observe should be chosen carefully in order to avoid inaccuracy problems. Thus, the accuracy analysis of the new robot has been performed to show the importance of the leg selection.

All these results have been validated on a GS platform simulator created using ADAMS/Controls and interfaced with Matlab/Simulink. The proposed approach can be extended for the control of GS platforms and other types of robots in the image space. This is part of our future works in this field.

ACKNOWLEDGEMENTS

This work was supported by the French ANR ARROW (ANR 2011BS3 006 01). The authors would also like to thank Nicolas Bouton from the IFMA (Institut Pascal) of Clermont-Ferrand for its great help in using Adams/Controls.

REFERENCES

[1] J. Merlet, *Parallel Robots*. Springer, 2nd ed., 2006.
[2] J. Merlet, "http://www-sop.inria.fr/members/jean-pierre.merlet/merlet.html," 2012.
[3] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, 1992.
[4] 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.
[5] P. Martinet, J. Gallice, and D. Khadraoui, "Vision based control law using 3d visual features," in *Proceedings of the World Automation Congress, WAC96, Robotics and Manufacturing Systems*, vol. 3, (Montpellier, France), pp. 497–502, May 1996.
[6] N. Andreff, A. Marchadier, and P. Martinet, "Vision-based control of a gough-stewart parallel mechanism using legs observation," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'05*, (Barcelona, Spain), pp. 2546–2551, April 18-22 2005.
[7] V. Gough and S. Whitehall, "Universal tyre test machine," in *Proceedings of the FISITA 9th International Technical Congress*, pp. 117–317, May 1962.
[8] N. Andreff and P. Martinet, "Visually servoing a gough-stewart parallel robot allows for reduced and linear kinematic calibration," in *Proceedings of the 2nd International Conference on Informatics in Control, Automation and Robotics, ICINCO'05*, vol. 3, (Barcelona, Spain), pp. 119–124, September 14-17 2005.
[9] E. Ozgur, N. Andreff, and P. Martinet, "Dynamic control of the quattro robot by the leg edgels," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA11*, (Shanghai, China), May 9-13 2011.
[10] F. Chaumette, *La commande des robots manipulateurs*. Hermes, 2002.
[11] C. Tischler, K. Hunt, and A. Samuel, "A spatial extension of cardanic movement: its geometry and some derived mechanisms," *Mechanism and Machine Theory*, vol. 33, pp. 1249–1276, 1998.
[12] C. Gosselin and J. Angeles, "Singularity analysis of closed-loop kinematic chains," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 281–290, 1990.
[13] P. Ben-Horin and M. Shoham, "Singularity analysis of a class of parallel robots based on grassmanncayley algebra," *Mechanism and Machine Theory*, vol. 41, pp. 958–970, August 2006.
[14] S. Caro, G. Moroz, T. Gayral, D. Chablat, and C. Chen, "Singularity analysis of a six-dof parallel manipulator using grassmann-cayley algebra and grobner bases," in *Proceedings of the Symposium on Brain, Body and Machine*, (Montreal, QC, Canada), November 10-12 2010.
[15] O. Ma and J. Angeles, "Architecture singularities of parallel manipulators," *The International Journal of Robotics and Automation*, vol. 7, no. 1, pp. 23–29, 1992.
[16] J. Merlet, "Jacobian, manipulability, condition number, and accuracy of parallel robots," *ASME Transactions Journal of Mechanical Design*, vol. 128, no. 1, p. 199206, 2006.
[17] V. Arakelian, S. Briot, and V. Glazunov, "Increase of singularity-free zones in the workspace of parallel manipulators using mechanisms of variable structure," *Mechanism and Machine Theory*, vol. 43, no. 9, pp. 1129–1140, 2008.
[18] A. Pashkevich, D. Chablat, and P. Wenger, "Stiffness analysis of overconstrained parallel manipulators," *Mechanism and Machine Theory*, vol. 44, no. 5, pp. 966 – 982, 2009.
[19] N. Binaud, P. Cardou, S. Caro, and P. Wenger, "The kinematic sensitivity of robotic manipulators to joint clearances," in *Proceedings of ASME Design Engineering Technical Conferences*, (Montreal, QC, Canada), August 15-18 2010.
[20] S. Briot and V. Arakelian, "Optimal force generation of parallel manipulators for passing through the singular positions," *International Journal of Robotics Research*, vol. 27, no. 8, pp. 967–983, 2008.
[21] M. Zein, P. Wenger, and D. Chablat, "Non-singular assembly-mode changing motions for 3-rpr parallel manipulators," *Mechanism and Machine Theory*, vol. 43, no. 4, pp. 480–490, 2008.
[22] I. A. Bonev, D. Chablat, and P. Wenger, "Working and assembly modes of the agile eye," in *Proceedings of the IEEE International Conference On Robotics And Automation (ICRA 1996)*, (Orlando, Florida, USA), 1996.
[23] H. Michel and P. Rives, "Singularities in the determination of the situation of a robot effector from the perspective view of 3 points," tech. rep., INRIA, 1993.
[24] W. Khalil and E. Dombre, *Modeling, Identification and Control of Robots*. Hermes Penton London, 2002.