



HAL
open science

Minimal Representation for the Control of Parallel Robots via Leg Observation Considering a Hidden Robot Model

Sébastien Briot, Victor Rosenzweig, Philippe Martinet, Erol Ozgür, Nicolas Bouton

► **To cite this version:**

Sébastien Briot, Victor Rosenzweig, Philippe Martinet, Erol Ozgür, Nicolas Bouton. Minimal Representation for the Control of Parallel Robots via Leg Observation Considering a Hidden Robot Model. Mechanism and Machine Theory, 2016, 106. hal-01357509

HAL Id: hal-01357509

<https://hal.science/hal-01357509>

Submitted on 25 Jun 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Minimal Representation for the Control of Parallel Robots via Leg Observation Considering a Hidden Robot Model

Sébastien Briot^{a,*}, Victor Rosenzweig^a, Philippe Martinet^{a,b}, Erol Özgür^c, Nicolas Bouton^c

^a*Institut de Recherches en Communications et Cybernétique de Nantes (IRCCyN)
UMR CNRS 6597 – 1 rue de la Noë, 44321 Nantes, France*

^b*École Centrale de Nantes, 1 rue de la Noë, 44321 Nantes, France*

^c*Institut Pascal (UMR 6602) – Sigma Clermont – Campus des Cèzeaux, 27 Rue Roche Genès, 63178 Aubière, France*

Abstract

Previous works on the visual servoing of parallel robots using the observation of their leg directions validated the feasibility of the approach but they have enlighten two main surprising results for which no answer was given: (i) the observed robot which is composed of n legs could be controlled in most cases using the observation of only m leg directions ($m < n$), and that (ii) in some cases, the robot did not converge to the desired end-effector pose, even if the observed leg directions did (i.e. there was not a global diffeomorphism between the observation space and the robot space).

Recently, it was shown that the visual servoing of the leg directions of the Gough-Stewart platform and the Adept Quattro with 3 translational degrees of freedom was equivalent to controlling other virtual robots that have assembly modes and singular configurations different from those of the real ones. These hidden robot models are tangible visualizations of the mapping between the observation space and the real robots Cartesian space. Thanks to this concept, all the aforementioned points were answered for the mentioned robots.

In this paper, the concept of hidden robot model is generalized for any type of parallel robots controlled using visual servos based on the observation of the leg directions. It is shown that the concept of hidden robot model is a powerful tool that gives useful insights about the visual servoing of robots using leg direction observation. With the concept of hidden robot model, the singularity problem of the controller can be addressed and the convergence issues of the controller can be explained, understood and solved.

All these results are validated in simulations and through experiments on a Quattro robot.

Keywords: Parallel robots, visual servoing, controllability, kinematics, singularity.

1. Introduction

Parallel robots are mechanical architectures whose end-effector is linked to the fixed base by means of at least two kinematic chains [1]. Compared to serial robots, such robots are stiffer and can reach higher speeds and accelerations [2]. However, their control is troublesome because of the complex mechanical structure, highly coupled joint motions and many other factors (e.g. clearances, assembly errors, etc.) which degrade stability and accuracy.

Many research papers focus on the control of parallel mechanisms [2]. 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, in a general manner, 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.

Past research works have proven that the robot end-effector pose can be effectively estimated by vision [3, 4, 5]. The most common approach consists of the direct observation of the end-effector pose [6,

*Corresponding author

Email addresses: Sebastien.Briot@irccyn.ec-nantes.fr (Sébastien Briot),
Victor.Rosenzweig@irccyn.ec-nantes.fr (Victor Rosenzweig), Philippe.Martinet@irccyn.ec-nantes.fr
(Philippe Martinet), erol.ozgur@sigma-clermont.fr (Erol Özgür), nicolas.bouton@sigma-clermont.fr (Nicolas Bouton)

14 7, 8]. However, some applications prevent the observation of the end-effector of a parallel mechanism by
15 vision [9, 10, 11, 12]. For instance, it is not wise to imagine observing the end-effector of a machine-tool
16 while it is generally not a problem to observe its legs that are most often designed with slim and rectilinear
17 rods [2].

18 A first step in this direction was made in [13] where vision was used to derive a visual servoing scheme
19 based on the observation of a Gough-Stewart (GS) parallel robot [14]. In that method, the leg directions were
20 chosen as visual primitives and control was derived based on their reconstruction from the image. By stacking
21 the observation matrices corresponding to the observation of several legs, a control scheme was derived and
22 it was then shown that such an approach allowed the control of the observed robot. After these preliminary
23 works, the approach was extended to the control of the robot directly in the image space by the observation of
24 the leg edges (from which the leg direction can be extracted), which has proven to exhibit better performances
25 in terms of accuracy than the previous approach [15]. The approach was applied to several types of robots,
26 such as the Adept Quattro and other robots of the same family [16, 17].

27 The proposed control scheme was not usual in visual servoing techniques, in the sense that in the con-
28 troller, both robot kinematics and observation models linking the Cartesian space to the leg direction space
29 are involved. As a result, some surprising results were obtained:

- 30 1. the observed robot which is composed of n legs could be controlled in most cases using the observation
31 of only m leg directions ($m < n$), knowing the fact that the minimal number of observed legs should be,
32 for 3D unit vectors characterizing the leg directions, an integer greater than $n/2$
- 33 2. in some cases, the robot does not converge to the desired end-effector pose (even if the observed leg
34 directions did)

35 without finding some concrete explanations to these points. Especially, the last point showed that it may be
36 possible that a global diffeomorphism between the Cartesian space and the leg direction space does not exist,
37 but no formal proof was given.

38 In parallel, some important questions were never answered, such as:

- 39 3. How can we be sure that the stacking of the observation matrices cannot lead to local minima (for
40 which the error in the observation space is non zero while the robot platform cannot move [18]) in the
41 Cartesian space?
- 42 4. Are we sure that there is no singularity in the mapping between the leg direction space and the Cartesian
43 space?

44 All these points were never answered because of the *lack of existing tools* able to analyze the intrinsic
45 properties of the controller. Additionally, we would like to point out that the understanding of the singularity
46 cases of the mapping used in the controller is of the utmost because these singularities leads to the loss
47 of controllability of the robot [19], and thus the define the boundaries of the reachable workspace for the
48 controller. As a result, the accessible workspace for the robot controlled by leg observation is the intersection
49 of two workspaces: (1) the singularity-free workspace of the robot and (2) the workspace free of singularities
50 linked to the mapping between the leg direction space and the Cartesian space.

51 Recently, two of the authors of the present paper have demonstrated in [20] that these points could be
52 explained by considering that the visual servoing of the leg directions of the GS platform was equivalent to
53 controlling another robot “hidden” within the controller, the 3-UPS¹ that has assembly modes and singular
54 configurations different from those of the GS platform.

55 In both cases, considering this hidden robot model allowed the finding of a minimal representation for the
56 leg-observation-based control of the studied robots that is linked to a virtual hidden robot which is a tangible
57 visualization of the mapping between the observation space and the real robot Cartesian space. The hidden
58 robot model:

- 59 1. can be used to explain why the observed robot which is composed of n legs can be controlled using the
60 observation of only m leg directions ($m < n$),

¹In the following of the paper, R, P, U, S, Π will stand for passive revolute, prismatic, universal, spherical and planar parallelogram joint [21], respectively. If the letter is underlined, the joint is considered active.

- 61 2. can be used to prove that there does not always exist a global diffeomorphism between the Cartesian
62 space and the leg direction space, but can also bring solutions for avoiding to converge to a non desired
63 pose,
64 3. simplifies the singularity analysis of the mapping between the leg direction space and the Cartesian
65 space by reducing the problem to the singularity analysis of a new robot.
66 4. can be used to certify that the robot will not converge to local minima, through the application of tools
67 developed for the singularity analysis of robots.

68 Thus, the concept of hidden robot model, associated with mathematical tools developed by the mechanical
69 design community, is a powerful tool able to analyze the intrinsic properties of some controllers developed
70 by the visual servoing community. Moreover, this concept shows that in some visual servoing approaches,
71 stacking several interaction matrices to derive a control scheme without doing a deep analysis of the intrinsic
72 properties of the controller is clearly not enough. Further investigations are required.

73 Therefore, in this paper, the generalization of the concept of hidden robot model is presented and a general
74 way to find the hidden robots corresponding to any kind of robot architecture is explained. It will be shown
75 that the concept of hidden robot model is a powerful tool that gives useful insights about the visual servoing
76 of robots using leg direction observation. With the concept of hidden robot model, the singularity problem
77 of the mapping between the space of the observed robot links and the Cartesian space can be addressed, and
78 above all, it is possible to give and certify information about the controllability of the observed robots using
79 the proposed controller. Therefore, with the hidden robot concept, we are able to understand and find the
80 minimal representation for the control of parallel robots via leg observation, i.e. to find what are the necessary
81 (minimal) information to use in the controller in order to allow the full control of the robot.

82 The paper is decomposed as follows. Section 2 makes some brief recalls on the visual servoing of parallel
83 robots using leg observations. Then, Section 3 presents the concept of hidden robot model and generalizes
84 the approach for any type of parallel robots. In Section 4, some examples of typical classes of parallel robots
85 are studied: the planar parallel robots, the n -Pod family (i.e. GS platform-like robots), and the Delta-like
86 robots. Simulations and experimental validations on the Adept Quattro are presented in Section 5. Finally,
87 our conclusions are written in Section 6.

88 2. Background on visual servoing of parallel robots using leg observations

89 2.1. Line modeling

90 A line \mathcal{L} in space, expressed in the camera frame, is defined by its Bi-normalized Plücker coordinates [22]:
91

$$92 \quad \mathcal{L} \equiv (\underline{c}\mathbf{u}, \underline{c}\mathbf{n}, c_n) \tag{1}$$

93 where $\underline{c}\mathbf{u}$ is the unit vector giving the spatial orientation of the line², $\underline{c}\mathbf{n}$ is the unit vector defining the so-called
94 interpretation plane of line \mathcal{L} and c_n is a non-negative scalar. The latter are defined by $c_n \underline{c}\mathbf{n} = {}^c\mathbf{P} \times \underline{c}\mathbf{u}$ where
95 ${}^c\mathbf{P}$ is the position of any point P on the line, expressed in the camera frame. Notice that, using this notation,
96 the well-known (normalized) Plücker coordinates [23, 2] are the couple $(\underline{c}\mathbf{u}, c_n \underline{c}\mathbf{n})$.

97 The projection of such a line in the image plane, expressed in the camera frame, has for characteristic
98 equation [22]:

$$99 \quad \underline{c}\mathbf{n}^T \underline{c}\mathbf{p} = 0 \tag{2}$$

100 where $\underline{c}\mathbf{p}$ are the coordinates in the camera frame of a point P in the image plane, lying on the line.

²In the following of the paper, the superscript before the vector denotes the frame in which the vector is expressed (“ b ” for the base frame, “ c ” for the camera frame and “ p ” for the pixel frame). If there is no superscript, the vector can be written in any frame.

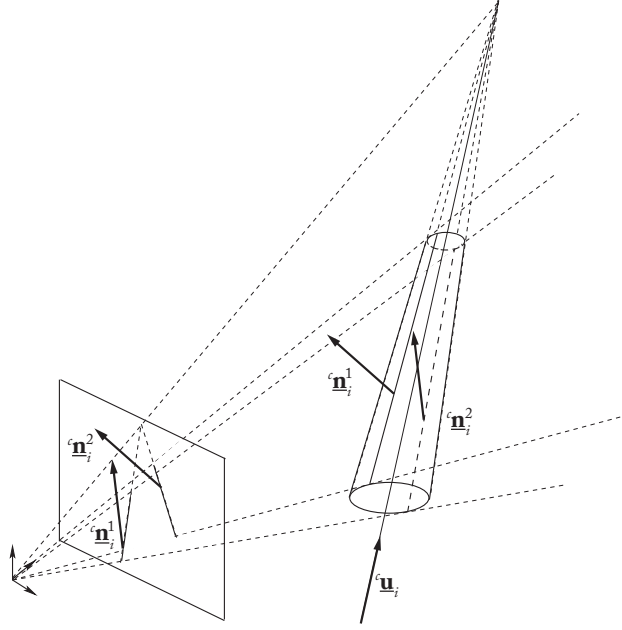


Figure 1: Projection of a cylinder in the image

101 2.2. Cylindrical leg observation

102 The legs of parallel robots have usually cylindrical cross-sections [2]. The edges of the i -th cylindrical leg
 103 are given, in the camera frame, by [15] (Fig 1):

$$104 \quad {}^c \underline{\mathbf{n}}_i^1 = -\cos \theta_i {}^c \underline{\mathbf{h}}_i - \sin \theta_i {}^c \underline{\mathbf{u}}_i \times {}^c \underline{\mathbf{h}}_i \quad (3)$$

$$105 \quad {}^c \underline{\mathbf{n}}_i^2 = +\cos \theta_i {}^c \underline{\mathbf{h}}_i - \sin \theta_i {}^c \underline{\mathbf{u}}_i \times {}^c \underline{\mathbf{h}}_i \quad (4)$$

106 where $\cos \theta_i = \sqrt{c h_i^2 - R_i^2} / c h_i$, $\sin \theta_i = R_i / c h_i$ and $({}^c \underline{\mathbf{u}}_i, {}^c \underline{\mathbf{h}}_i, c h_i)$ are the Bi-normalized Plücker coordinates of
 107 the cylinder axis and R_i is the cylinder radius.

108 It was also shown in [15] that the leg orientation, expressed in the camera frame, is given by

$$109 \quad {}^c \underline{\mathbf{u}}_i = \frac{{}^c \underline{\mathbf{n}}_i^1 \times {}^c \underline{\mathbf{n}}_i^2}{\|{}^c \underline{\mathbf{n}}_i^1 \times {}^c \underline{\mathbf{n}}_i^2\|} \quad (5)$$

110 Let us remark now that each cylinder edge is a line in space, with Bi-normalized Plücker expressed in the
 111 camera frame $({}^c \underline{\mathbf{u}}_i, {}^c \underline{\mathbf{n}}_i^j, c n_i^j)$ (Fig 1).

112 2.3. Leg direction based visual servoing

113 The proposed control approach was to servo the leg directions ${}^c \underline{\mathbf{u}}_i$ [13]. Some brief recalls on this type of
 114 controller are done below.

115 2.3.1. Interaction matrix

116 Visual servoing is based on the so-called interaction matrix \mathbf{L}^T [24] which relates the instantaneous relative
 117 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
 118 primitives that are used through:

$$119 \quad \dot{s} = \mathbf{L}_{(s)}^T T_c \quad (6)$$

120 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.
 121 the camera frame.

122 In the case where we want to directly control the leg directions ${}^c \underline{\mathbf{u}}_i$, and if the camera is fixed, (6) becomes:

$$123 \quad {}^c \dot{\underline{\mathbf{u}}}_i = \mathbf{M}_i^T {}^c \tau_c \quad (7)$$

124 where \mathbf{M}_i^T is the interaction matrix for the leg i .

126 2.3.2. Control

127 For the visual servoing of a robot, one achieves exponential decay of an error $e(s, s_d)$ between the current
 128 primitive vector s and the desired one s_d using a proportional linearizing and decoupling control scheme of
 129 the form:

$$130 \quad T_c = \lambda \hat{\mathbf{L}}_{(s)}^{T+} e(s, s_d) \quad (8)$$

131 where T_c is used as a pseudo-control variable and the superscript “+” corresponds to the matrix pseudo-
 132 inverse.

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

$$135 \quad \mathbf{e}_i = {}^c \underline{\mathbf{u}}_i \times {}^c \underline{\mathbf{u}}_{di} \quad (9)$$

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

137 It can be proven that, for spatial parallel robots, matrices \mathbf{M}_i are in general of rank 2 [13] (for planar
 138 parallel robots, they are of rank 1). As a result, for spatial robots with more than 2 *dof*, the observation of
 139 several independent legs is necessary to control the end-effector pose. An interaction matrix \mathbf{M}^T can then
 140 obtained by stacking k matrices \mathbf{M}_i^T of k legs.

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

$$143 \quad \dot{\mathbf{e}} = -\lambda \mathbf{e} \quad (10)$$

144 Then, introducing $\mathbf{L}_i^T = -[{}^c \underline{\mathbf{u}}_{di}]_{\times} \mathbf{M}_i^T$, where $[{}^c \underline{\mathbf{u}}_{di}]_{\times}$ is the cross product matrix associated with the vector
 145 ${}^c \underline{\mathbf{u}}_{di}$, the combination of (9), (7) and (10) gives

$$146 \quad {}^c \boldsymbol{\tau}_c = -\lambda \mathbf{L}^{T+} \mathbf{e} \quad (11)$$

147 where \mathbf{L}^T can be obtained by stacking the matrices \mathbf{L}_i^T of k legs. The conditions for the rank deficiency of
 148 matrix \mathbf{L}^T , as well as the conditions that lead to local minima [18] of the Eq. (11) are discussed in Section 3.

149 This expression can be transformed into the control joint velocities:

$$150 \quad \dot{\mathbf{q}} = -\lambda {}^c \mathbf{J}^{inv} \mathbf{L}^{T+} \mathbf{e} \quad (12)$$

151 where ${}^c \mathbf{J}^{inv}$ is the inverse Jacobian matrix of the robot relating the end-effector twist to the actuator velocities,
 152 i.e. ${}^c \mathbf{J}^{inv} {}^c \boldsymbol{\tau}_c = \dot{\mathbf{q}}$.

153 In the next Section, it is shown that the equations used in the controller, characterizing the mapping
 154 between the observation (leg-direction) space and the Cartesian space, can indeed be related to a virtual
 155 architecture hidden within the controller. Understanding and using the models of these hidden robots is
 156 mandatory for being able to analyze the controllability of parallel robots using the proposed visual servoing
 157 approach.

158 3. The concept of hidden robot model

159 The concept of hidden robot model has been first introduced in [20] for the visual servoing of the GS
 160 platform. In this paper, it has been demonstrated that the leg direction based visual servoing of such robots
 161 intrinsically involves the appearance of a hidden robot model, which has assembly modes and singularities
 162 different from the real robot. It was shown that the concept of hidden robot model fully explains the possible
 163 non convergence of the observed robot to the desired final pose and that it considerably simplifies the singu-
 164 larity analysis of the mapping involved in the controller. The aim of this Section is to generalize and extend
 165 the concept to any class of robots.

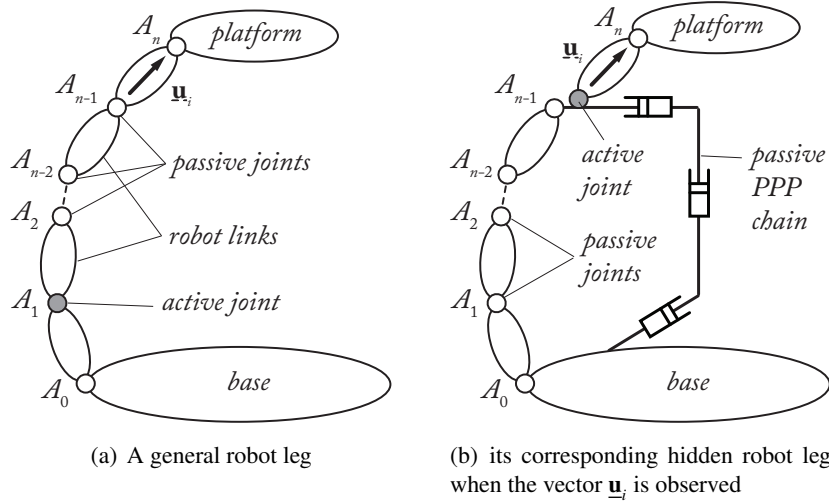


Figure 2: A general robot leg and its corresponding hidden robot leg when the vector $\underline{\mathbf{u}}_i$ is observed

166 **3.1. Statement of the problem**

167 The concept of hidden robot model comes from the following observation: in the classical control ap-
 168 proach, the encoders measure the motion of the actuator; in the previously described control approach (Sec-
 169 tion 2), the leg directions or leg edges are observed. So, in a reciprocal manner, one could wonder to what
 170 kind of virtual actuators such observations correspond, and as a result, what could be the virtual architecture
 171 hidden inside the controller of a given robot. The main objective of this Section is to give a general answer to
 172 these questions.

173 However, we want to claim that our objective is not to prove the unicity of the virtual robot hidden in
 174 the controller. Indeed, several robot architectures could be found for the same controller, but all of them
 175 will be equivalent in terms of geometry and kinematics (same input-output relationships, same singularities).
 176 So, it is not necessary to find an architecture solution of an optimal problem, but it is sufficient to find one
 177 architecture having the desired geometric and kinematic properties in order to solve the singularity problem
 178 of the controller. This is what is shown below with a rather simple and intuitive approach.

179 **3.2. How to define the legs of the hidden robots**

180 Let us consider a general leg for a parallel robot in which the direction $\underline{\mathbf{u}}_i$ of a segment is observed
 181 (Fig. 2(a) – in this figure, the last segment is considered observed, but the following explanations can be
 182 generalized to any segment located in the leg chain). In what follows, we only consider that we observe the
 183 leg direction $\underline{\mathbf{u}}_i$, and not the leg edges in the image space, as the leg edges are only used as a measure of $\underline{\mathbf{u}}_i$.
 184 So *the problem is the same*, except in the fact that we must consider the singularity of the mapping between
 185 the edges and $\underline{\mathbf{u}}_i$, but this problem is well handled: these singularities appear when $\underline{\mathbf{n}}_i^1$ and $\underline{\mathbf{n}}_i^2$ are collinear, i.e.
 186 the cylinders are at infinity [15].

187 In the general case, the unit vector $\underline{\mathbf{u}}_i$ can obviously be parameterized by two independent coordinates,
 188 that can be two angles, for example the angles α and β of Fig. 3 defined such that $\cos \alpha = \underline{\mathbf{x}} \cdot \underline{\mathbf{v}} = \underline{\mathbf{y}} \cdot \underline{\mathbf{w}}$ (where
 189 $\underline{\mathbf{v}}$ and $\underline{\mathbf{w}}$ are defined such that $\underline{\mathbf{z}} \cdot \underline{\mathbf{v}} = \underline{\mathbf{z}} \cdot \underline{\mathbf{w}} = 0$) and $\cos \beta = \underline{\mathbf{u}} \cdot \underline{\mathbf{x}}$. Thus α is the angle of the first rotation of the
 190 link direction $\underline{\mathbf{u}}_i$ around $\underline{\mathbf{z}}$ and β is the angle of the second rotation around $\underline{\mathbf{v}}$.

191 It is well known that a U joint is able to orientate a link around two orthogonal axes of rotation, such as $\underline{\mathbf{z}}$
 192 and $\underline{\mathbf{v}}$. Thus U joints can be the virtual actuators with generalized coordinates α and β we are looking for. Of
 193 course, other solutions can exist, but U joints are the simplest ones.

194 If a U joint is the virtual actuator that makes the vector $\underline{\mathbf{u}}_i$ move, it is obvious that:

- 195 • if the value of $\underline{\mathbf{u}}_i$ is fixed, the U joint coordinates α and β must be constant, i.e. *the actuator must be*
 196 *blocked*,
- 197 • if the value of $\underline{\mathbf{u}}_i$ is changing, the U joint coordinates α and β must also vary.

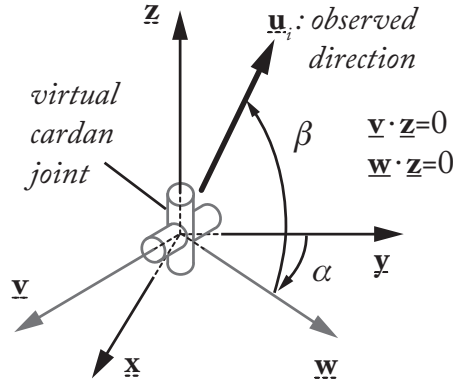


Figure 3: Parameterization of a unit vector \underline{u}_i with respect to a given frame \underline{x} , \underline{y} and \underline{z}

198 As a result, to ensure the aforementioned properties for α and β if \underline{u}_i is expressed in the base or camera
 199 frame (but the problem is identical as the camera is considered fixed on the ground), vectors \underline{x} , \underline{y} and \underline{z} of
 200 Fig. 3 must be the vectors defining the base or camera frame. Thus, in terms of properties for the virtual
 201 actuator, this implies that the first \underline{U} joint axis must be constant w.r.t. the base frame, i.e. the \underline{U} joint must be
 202 attached to a link *performing a translation w.r.t. the base frame*³.

203 However, in most of the cases, the real leg architecture is not composed of \underline{U} joints attached on links
 204 performing a translation w.r.t. the base frame. Thus, the architecture of the hidden robot leg must be modified
 205 w.r.t. the real leg such as depicted in Fig. 2(b). The \underline{U} joint must be mounted on a passive kinematic chain
 206 composed of at most 3 orthogonal passive \underline{P} joints that ensures that the link on which is it attached performs
 207 a translation w.r.t. the base frame. This passive chain is also linked to the segments before the observed links
 208 so that they do not change their kinematic properties in terms of motion. Note that:

- 209 • it is necessary to fix the \underline{PPP} chain on the preceding leg links because the information given by the
 210 vectors \underline{u}_i is not enough for rebuilding the full platform position and orientation: it is also necessary to
 211 get information on the location of the anchor point A_{n-1} of the observed segment [15]. This information
 212 is kept through the use of the \underline{PPP} chain fixed on the first segments;
- 213 • 3 \underline{P} joints are only necessary if and only if the point A_{n-1} describes a motion in the 3D space; if not, the
 214 number of \underline{P} joints can be decreased: for example, in the case of the GS platform presented in [20], the
 215 \underline{U} joint of the leg to control was located on the base, i.e. there was no need to add passive \underline{P} joints to
 216 keep the orientation of its first axis constant;
- 217 • when the vector \underline{u}_i is constrained to move in a plane such as for planar legs, the virtual actuator be-
 218 comes an \underline{R} joint which must be mounted on the passive \underline{PPP} chain (for the same reasons as mentioned
 219 previously).

220 For example, let us have a look at the \underline{RU} leg with one actuated \underline{R} joint followed by a \underline{U} joint of Fig. 4(a).
 221 Using the previous approach, its virtual equivalent leg should be an $\{\underline{R}-\underline{PP}\}-\underline{U}$ leg (Fig. 4(b)), i.e. the \underline{U} joint
 222 able to orientate the vector \underline{u}_i is mounted on the top of a $\underline{R}-\underline{PP}$ chain that can guarantee that:

- 223 1. the link on which the \underline{U} joint is attached performs a translation w.r.t. the base frame,
- 224 2. the point C (i.e. the centre of the \underline{U} joint) evolves on a circle of radius l_{AB} , like the real leg.

225 It should be noticed that, in several cases for robots with a lower mobility (i.e. spatial robots with a number
 226 of *dof* less than 6, or planar robots with a number of *dof* less than 3), the last joint that links the leg to the
 227 platform should be changed so that, if the number of observed legs is inferior to the number of real legs, the
 228 hidden robot keeps the same number of controlled *dof* (see Section 4.2.2).

³In the case where the camera is not mounted on the frame but on a moving link, the virtual \underline{U} joint must be attached on a link performing a translation w.r.t. the considered moving link.

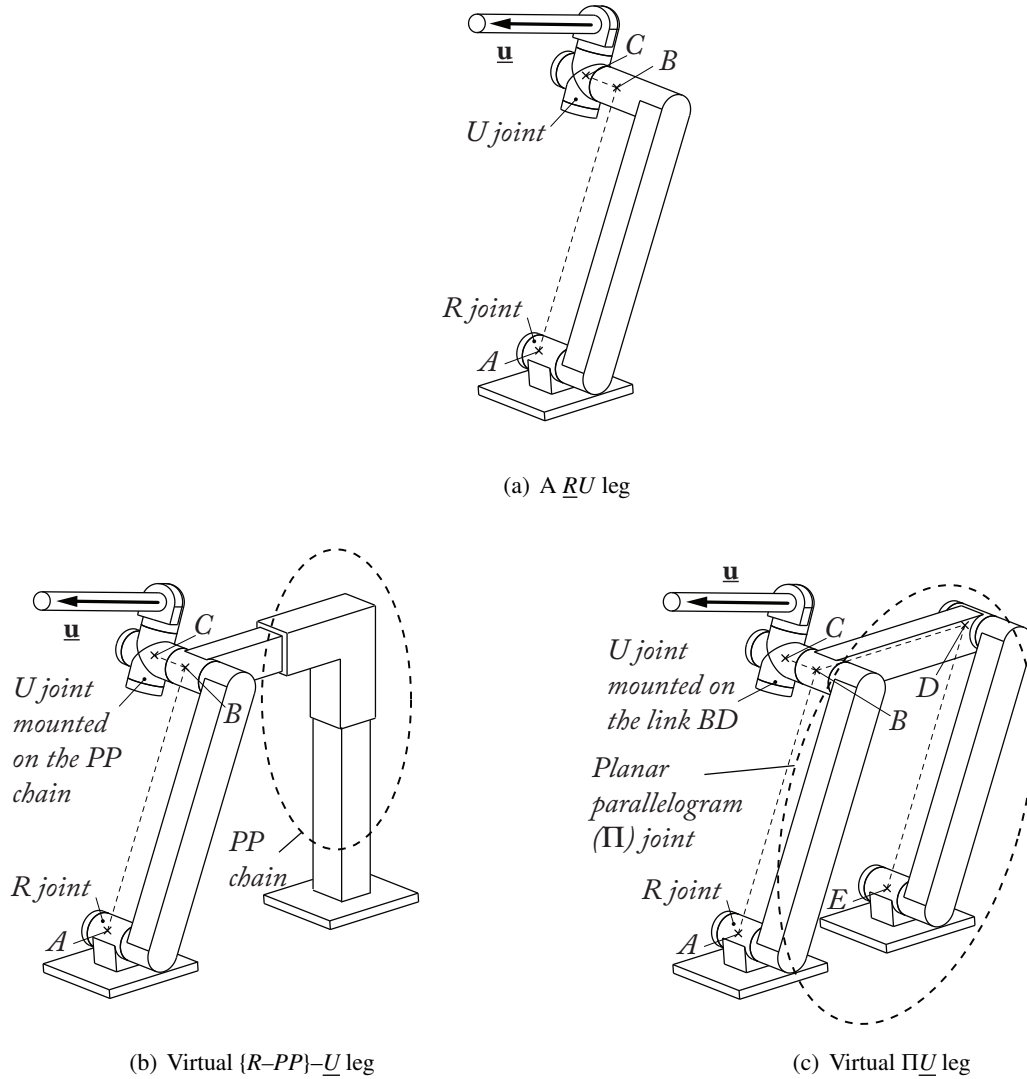


Figure 4: A \underline{RU} leg and two equivalent solutions for its hidden leg

229 It should also be mentioned that we have presented above the most general methodology that is possible
 230 to propose, but it is not the most elegant way to proceed. In many cases, a hidden robot leg architecture can
 231 be obtained such that less modifications w.r.t the real leg are achieved. For example, the $R-PP$ chain of the
 232 hidden robot leg $\{R-PP\}-\underline{U}$ (Fig. 4(b)) could be equivalently replaced by a planar parallelogram (Π) joint
 233 without changing the aforementioned properties of the \underline{U} virtual actuator (Fig. 4(c)), i.e. only one additional
 234 joint is added for obtaining the hidden robot leg (note that we consider that a Π joint, even if composed of
 235 several pairs, can be seen as one single joint, as in [21]).

236 Anyway, as mentioned above, the objective of this part is not to find an architecture solution of an optimal
 237 problem (optimal with respect to design complexity [21], for instance) but it was to find one architecture hav-
 238 ing the desired geometric and kinematic properties in order to solve the singularity problem of the controller.
 239 This is what was done below with a rather simple and intuitive approach.

240 It should be noted that the same results could be also demonstrated with more rigorous Type synthesis
 241 approaches [25, 26, 27]. However, adding such methodologies in the present paper would have not provided
 242 any further explanations of the problem and would have make it longer and more unclear.

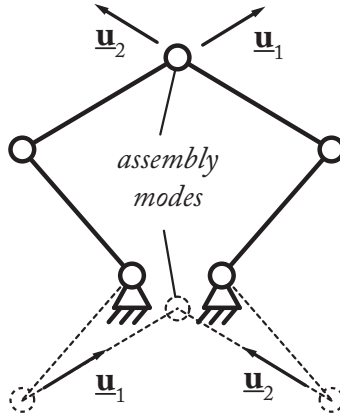


Figure 5: Two configurations of a five bar mechanism for which the directions \underline{u}_i are identical (for $i = 1, 2$)

243 3.3. How to use the hidden robot models for analyzing the controllability of the servoed robots

244 The aim of this Section is to show how to use the hidden robots for answering points 1 to 4 enumerated in
 245 the introduction of the paper.

246

247 *Point 1: the hidden robot model can be used to explain why the observed robot which is composed of n legs
 248 could be controlled in most cases using the observation of only m leg directions ($m < n$), knowing the fact that
 249 the minimal number of observed legs should be, for 3D unit vectors, an integer greater than $n/2$.*

250 To answer this point, let us consider a general parallel robot composed of 6 legs (one actuator per leg) and
 251 having six *dof*. Using the approach proposed in Section 3.2, each observed leg will lead to a modified virtual
 252 leg with at least one actuated \underline{U} joint that has two degrees of actuation. For controlling 6 *dof*, only 6 degrees
 253 of actuation are necessary, i.e. three actuated \underline{U} are enough (as long as the motions of the \underline{U} joints are not
 254 correlated, i.e. the robot is fully actuated). Thus, in a general case, only three legs have to be observed to fully
 255 control the platform *dof*.

256

257 *Point 2: the hidden robot model can be used to prove that there does not always exist a global diffeomorphism
 258 between the Cartesian space and the leg direction space, but can also bring solutions for avoiding to converge
 259 to a non desired pose.*

260 Here, the answer comes directly from the fact that the real controlled robot may have a hidden robot
 261 model with different geometric and kinematics properties. This means that the hidden robot may have as-
 262 sembly modes and singular configurations different from those of the real robot. If the initial and final robot
 263 configurations are not included in the same aspect (i.e. a workspace area that is singularity-free and bounded
 264 by singularities [2]), the robot won't be able to converge to the desired pose, but to a pose that corresponds to
 265 another assembly mode that has the same leg directions as the desired final pose (see Fig. 5).

266 Solutions for avoiding to converge to a non-desired pose are given in Appendix Appendix A.

267

268 *Point 3: the hidden robot model simplifies the singularity analysis of the mapping between the leg direction
 269 space and the Cartesian space by reducing the problem to the singularity analysis of a new robot.*

270 The interaction matrix \mathbf{M}^T involved in the controller gives the value of ${}^c\dot{\mathbf{u}}$ as a function of ${}^c\boldsymbol{\tau}_c$. Thus,
 271 \mathbf{M}^T is the inverse Jacobian matrix of the hidden robot (and, consequently, \mathbf{M}^{T+} is the hidden robot Jacobian
 272 matrix). Except in the case of decoupled robots [28, 29, 30], the Jacobian matrices of parallel robots are not
 273 free of singularities.

274 Considering the input/output relations of a robot, three different kinds of singularity can be observed [31]⁴:

- 275 • the *Type 1 singularities* that appear when the robot Jacobian matrix is rank-deficient; in such configura-
276 tions, any motion of the actuator that belongs to the kernel of the Jacobian matrix is not able to produce
277 a motion of the platform,
- 278 • the *Type 2 singularities* that occur when the robot inverse Jacobian matrix is rank-deficient; in such
279 configurations, any motion of the platform that belongs to the kernel of the inverse Jacobian matrix
280 is not able to produce a motion of the actuator. And, reciprocally, near these configurations, a small
281 motion of the actuators lead to large platform displacements, i.e. the accuracy of the robot becomes
282 very poor,
- 283 • the *Type 3 singularities* that appear when both the robot Jacobian and inverse Jacobian matrices are
284 rank-deficient.

285 Thus,

- 286 • finding the condition for the rank-deficiency of \mathbf{M}^T is equivalent to find the Type 2 singularities of the
287 hidden robot,
- 288 • finding the condition for the rank-deficiency of \mathbf{M}^{T+} is equivalent to find the Type 1 singularities of the
289 hidden robot.

290 Since a couple of decades ago, many tools have been developed by the mechanical design community for
291 finding the singular configurations of robots. The interested reader could refer to [33, 2, 34, 35] and many
292 other works on the Grassmann Geometry and Grassmann-Cayley Algebra for studying the singular configu-
293 rations problem. In what follows in the paper, these tools are used but only the final results concerning the
294 singular configuration conditions are given.

295
296 *Point 4: the hidden robot model can be used to certify that the robot will not converge to local minima.*

297 The robot could converge to local minima if the matrix \mathbf{L}^{T+} of (11) is rank deficient. A necessary and
298 sufficient condition for the rank deficiency of this matrix is that the \mathbf{M}^{T+} is rank deficient, i.e. the hidden
299 robot model encounters a Type 1 singularity. As mentioned above, many tools have been developed by the
300 mechanical design community for finding the singular configurations of robots and solutions can be provided
301 to ensure that the hidden robot model does not meet any Type 1 singularity (see also the Appendix Appendix
302 A).

303 4. Hidden robot models of some known parallel robot families

304 Let us now present the hidden robot models of some well known families of parallel robots and deal with
305 their forward kinematics and singular configurations.

306 It should be noted that, in the singularity analysis of the considered robots, only the Type 2 singular
307 configurations are detailed as Type 1 singularity always appear when at least one leg is fully stretched or
308 folded (workspace boundary). Therefore, these conditions are not recalled.

309 4.1. Application to planar parallel robots

310 4.1.1. The hidden robot legs of planar parallel robots

311 The usual planar parallel manipulators (*ppm*) are composed of planar serial chains with at most three
312 1-*dof* joints, respectively, among which one is actuated. As mentioned in [2], using the different possible
313 combinations of *R* and *P* joints, only 10 different serial chains, that lead to robots that can be actuated, can

⁴There exist other types of singularities, such as the constraint singularities [32], but they are due to passive constraint degeneracy only, and are not involved in the mapping between the leg directions space and the robot controlled Cartesian coordinate space.

314 be obtained. These chains are represented in Table 1 (in this table and the following pictures, the gray pairs
315 denote the actuated joints).

316 Now, using the approach presented in Section 3.2, and considering that the direction $\underline{\mathbf{u}}_i$ of the last segment
317 of each leg is observed, one can find the hidden robot leg corresponding to this observation (Table 1).

Table 1: The 10 possible architectures for the legs of *ppm* and their equivalent hidden robot leg for the visual servoing using leg directions

| Real leg architecture | Hidden robot leg architecture | |
|---|---|---|
| <p>for \underline{RRR}</p> | <p>and \underline{RRR} legs \Rightarrow</p> | <p>a $\Pi\underline{RR}$ leg</p> |
| <p>for \underline{RPR}</p> | <p>and \underline{RPR} legs \Rightarrow</p> | <p>a \underline{RPR} leg</p> |
| <p>for \underline{PRR}</p> | <p>and \underline{PRR} legs \Rightarrow</p> | <p>a \underline{PRR} leg</p> |
| <p>for \underline{PPR}</p> | <p>and \underline{PPR} legs \Rightarrow</p> | <p>no equivalent hidden robot leg</p> |
| <p>for \underline{PRP} legs \Rightarrow</p> | <p>and \underline{PRP} legs \Rightarrow</p> | <p>a \underline{PRP} leg</p> |

Table 1: The 10 possible architectures for the legs of ppm and their equivalent hidden robot leg for the visual servoing using leg directions

| Real leg architecture | Hidden robot leg architecture |
|---|--|
| <p>for \underline{RRP} legs \Rightarrow</p> | <p>a $\Pi\underline{RRP}$ leg</p> |

318 From Table 1, the following information can be extracted:

- 319
- for \underline{RPR} and \underline{PRR} legs, the hidden robot legs are the same as the real ones;
 - \underline{PRP} legs lead to \underline{PRP} hidden robot legs; and robots made of \underline{PRP} legs are well known not to be controllable [2]. A similar result appears for \underline{RRP} legs that lead to $\underline{\Pi RR P}$ hidden robot legs.
 - the last element of \underline{PPR} and \underline{PPR} having a constant direction \underline{u}_i , robots made of such legs cannot be controlled using leg direction observation. As a result, they don't have an equivalent hidden robot model.

325 Thus, using the concept of hidden robot leg and hidden robot model, the problem of the robot controllability can be directly addressed without any mathematical derivations.

327 The next Section presents the hidden robot models of the 2 and 3- dof controllable robot with a symmetric leg arrangement and made of the legs presented in Table 1.

329 4.1.2. The hidden robot models of planar parallel robots

330 Using the results of the previous Section, 12 ppm with symmetric leg arrangement (i.e. with identical leg architectures) and that can be controlled using the leg direction observation can be found:

- 332
- for manipulators with 2 dof : \underline{RRRRR} , \underline{RRRRR} , \underline{RPRPR} , \underline{RPRPR} , \underline{PRRRP} , \underline{PRRRP} robots;
 - for manipulators with 3 dof : 3- \underline{RRR} , 3- \underline{RRR} , 3- \underline{RPR} , 3- \underline{RPR} , 3- \underline{PRR} , 3- \underline{PRR} robots.

334 Their architectures are well-known and, as they can also be easily deduced from the leg arrangement given in Tab. 1, their schematics are not depicted again.

336 However, for illustrating this Section, let us present the forward kinematic problem (fkp) and singularity analysis of the hidden robot model of the 3- \underline{RRR} robot, when controlled using leg direction observation (Fig. 6(a)). Using the results of Table 1, it can be found that its equivalent hidden robot model is a 3- $\underline{\Pi RR}$ robot (Fig. 6(b)). Each of its legs is composed of a passive planar parallelogram (Π joint) which is able to maintain constant the orientation of the links B_iD_i w.r.t. the base and of an \underline{RR} chain which is mounted on the link B_iD_i .

343 **Forward kinematics and assembly modes.** Using the usual methodology [2], all the solutions to the fkp are at the intersections of the coupler curve (which represents the displacement loci of one platform extremity when one of the leg is disassembled, the actuators of the two other being fixed (see Fig. 7(a))) with the vertex space of the disassembled leg (that represents the passive displacement of the leg tip when the actuator is fixed (see Table 1)). For the studied 3- $\underline{\Pi RR}$ robot, as the leg vertex spaces are circles (Table 1), the coupler curve is a sextic curve [2], i.e. an algebraic curve of degree 6 (in the case where the vertex spaces had been lines,

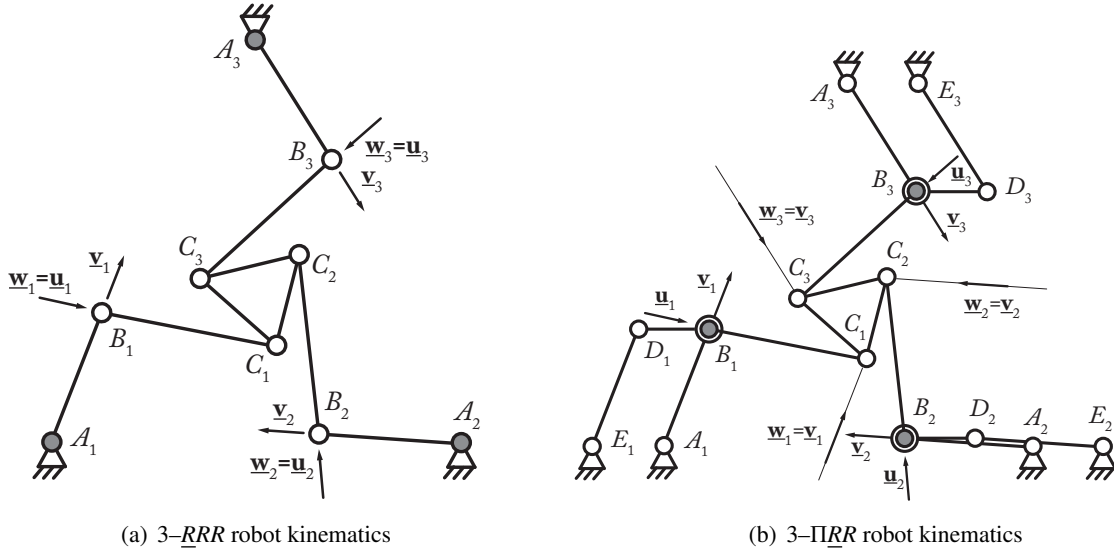


Figure 6: The 3-RRR robot and its hidden robot model

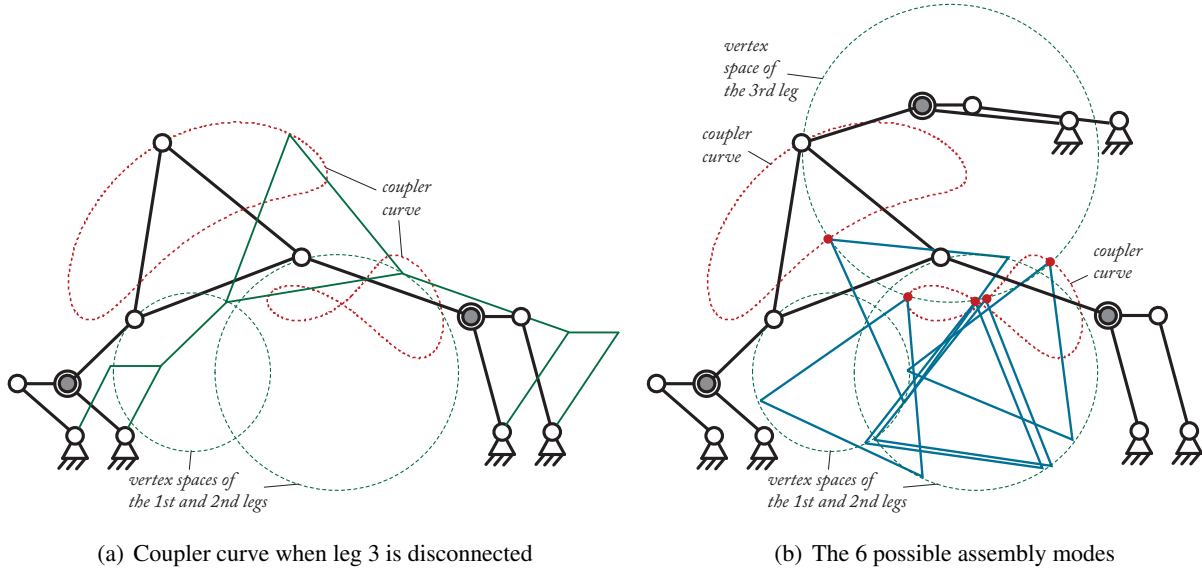


Figure 7: Solutions of the *fkp* for a 3-PIRR robot

349 the coupler curve would have been an ellipse [36]). Thus, the solutions of the *fkp* are at the intersection points
 350 between the aforementioned circle and sextic curve. And it is shown in [2] that if a circle intersects a sextic
 351 curve, there are at most 6 intersection points. An example of possible assembly modes for the 3-PIRR robot
 352 are presented in Fig. 7(b).

353 It should be mentioned that, even if for the 3-RRR (and as a consequence, for the 3-RRR), the hidden robot
 354 has 6 assembly modes, for the other 10 robots cited at the beginning of Section 4.1.2, the maximal number of
 355 assembly modes is 2 (see Appendix Appendix A), i.e. the control approach based on the observation of the
 356 leg direction allows most of the time the decrease of complexity for the *fkp*.

357

358 **Singular configurations.** The Type 2 singular configurations of *ppm* have been deeply studied in the past and
 359 are well-known. For 3 *dof ppm* (moving in the xOy plane), the singularities appear when $\mathbf{s}_1 \cdot (\mathbf{s}_2 \times \mathbf{s}_3) = 0$,
 360 where $\mathbf{s}_i^T = [w_j^x \ w_j^y \ m_j^z]$ in which w_j^x and w_j^y are the x and y components of \mathbf{w}_j (\mathbf{w}_j corresponds to the direction
 361 of the effort applied by the actuated leg on the platform [33] – see Table 1 and Fig. 6) and m_j^z is the moment

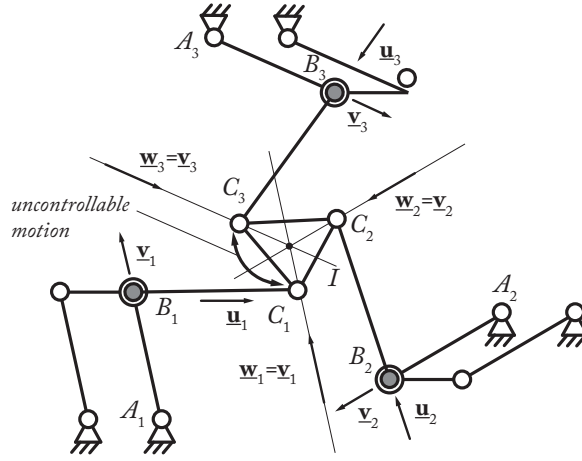


Figure 8: Example of Type 2 singularities for the 3-IIR robot

362 of $[w_j^x \ w_j^y]^T$ (note that \underline{w}_j is always applied at point C_j) [33]. Such a condition means that the lines of Plücker
 363 coordinates \underline{s}_j intersect in one single point I that corresponds to the instantaneous centre of rotation of the
 364 platform in the uncontrolled *dof* (Fig. 8). This point can be at infinity: in this case, all vectors \underline{w}_j are parallel
 365 and the robot gets one uncontrolled translational motion in the direction orthogonal to \underline{w}_j .

366 Obtained results show that singular configurations of the 3-IIR robot are different from those singular
 367 configurations of the real 3-RRR robot, for which they appear when all lines passing through C_i of direction
 368 \underline{u}_i intersects in one point [33].

369 Note that, near the hidden robot singular configurations, the real robot accuracy will be lower. Insights
 370 about this item are given in Appendix Appendix A.

371 Let us now apply the concept of hidden robot models to some particular classes of spatial parallel robots.

372 4.2. Application to spatial parallel robots

373 For spatial parallel robots, due to the existence of hundreds of possible different architectures and due to
 374 the difficulty of classifying the robots by families, it is not possible to present all the hidden robot models.
 375 Thus, it is decided in this Section to show the hidden robot models of two of the best known families of parallel
 376 robots:

- 377 • the n -Pod family (i.e. robots such as Hexapods (or GS platforms) [14], the Tsai mechanism [37], etc.)
- 378 • the Delta-like robot family (i.e. robots such as the Delta [38], the Quattro [39], the Orthoglide [40], etc.)

379 For other types of robot architectures, the interested reader is referred to Section 3.2 for finding a possible
 380 hidden robot model and to [2, 34, 35] and many other works on the Grassmann Geometry and Grassmann-
 381 Cayley Algebra for studying the singular configuration problem.

382 4.2.1. The n -Pod robot family

383 The n -Pod robot family regroups the robots made of n UPS legs (Fig. 9(a)), or some of their variations
 384 such as UPU [37] or even RPS legs [41], i.e. the legs are composed of one passive U or R joint located on the
 385 ground, followed by an active P joint and then by one passive S or U joint.

386 Probably the most known robots of this family are the GS platform [14] (Fig. 9(b)), the 3-UPU robots
 387 (e.g. see [37, 42]) and the 3-RPS robot [41].

388 For such types of legs, the prismatic joint direction can be observed [15, 20]. From Section 3.2 and also
 389 from [20], it can be shown that the virtual equivalent legs are:

- 390 • for UPS legs: a UPS leg;
- 391 • for UPU legs: a UPU leg;

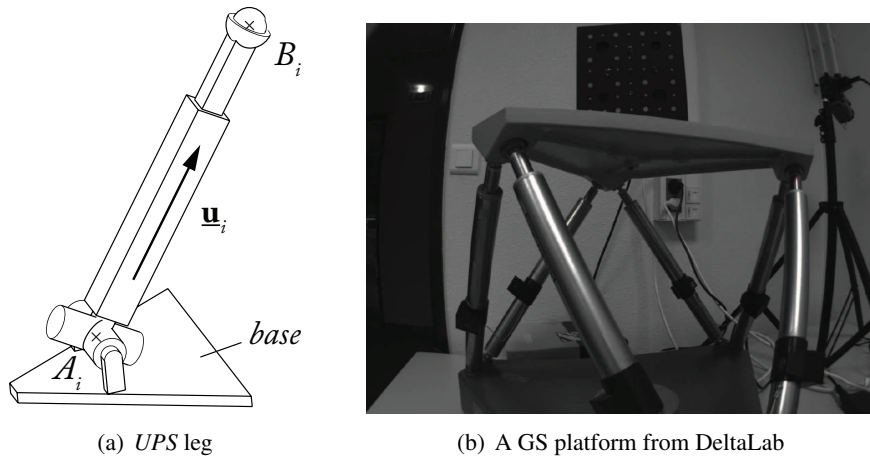


Figure 9: Typical legs and robots of the n -Pod family

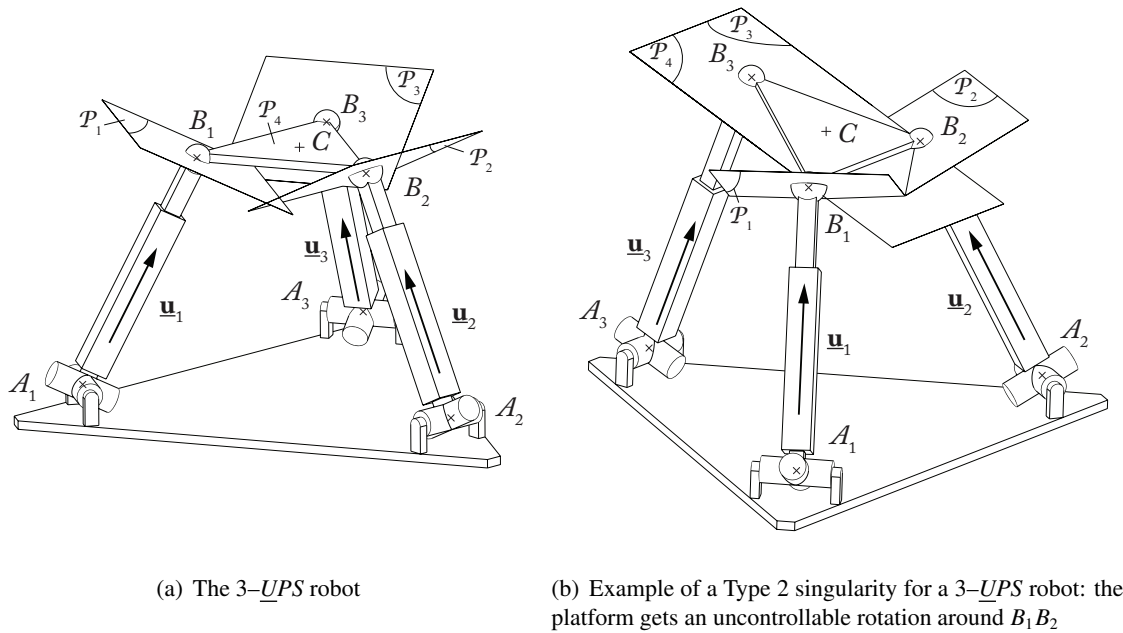


Figure 10: Hidden robot model of the GS platform when 3 legs are observed

392 • for RPS legs: a RPS leg;

393 i.e. the joint fixed on the ground becomes actuated, while the prismatic joint becomes passive. Then, using
 394 the usual methodology, the fkp and singularity analysis can be carried out.

395 For illustrating this Section, let us present the fkp and singularity analysis for the hidden robot model of
 396 the GS platform, when controlled using leg direction observation. This analysis was already made in [20] but
 397 the present section provides additional results.

398 The GS platform is made of six UPS legs, thus its equivalent hidden robot will be made of UPS legs. UPS
 399 legs have 2 degree of actuation (the U joint is fully actuated), and this is the reason with only three legs have to
 400 be observed for fully controlling the GS platform using leg direction observation [20]: in this case, the hidden
 401 robot is a 3- UPS robot which is well known to be fully actuated (Fig. 10(a)).

402

403 **Forward kinematics and assembly modes.** Without loss of generality, let us consider that we analyze the
 404 3- UPS robot depicted at Fig. 10(a). If leg 3 is disassembled at point B_3 , as there are only four actuators for
 405 controlling the six robot mobilities, the platform gains two dof . The gained motion is called a *spatial Cardanic*
 406 *motion* [43]. This motion is defined by the fact that points B_1 and B_2 are constrained to move on the lines

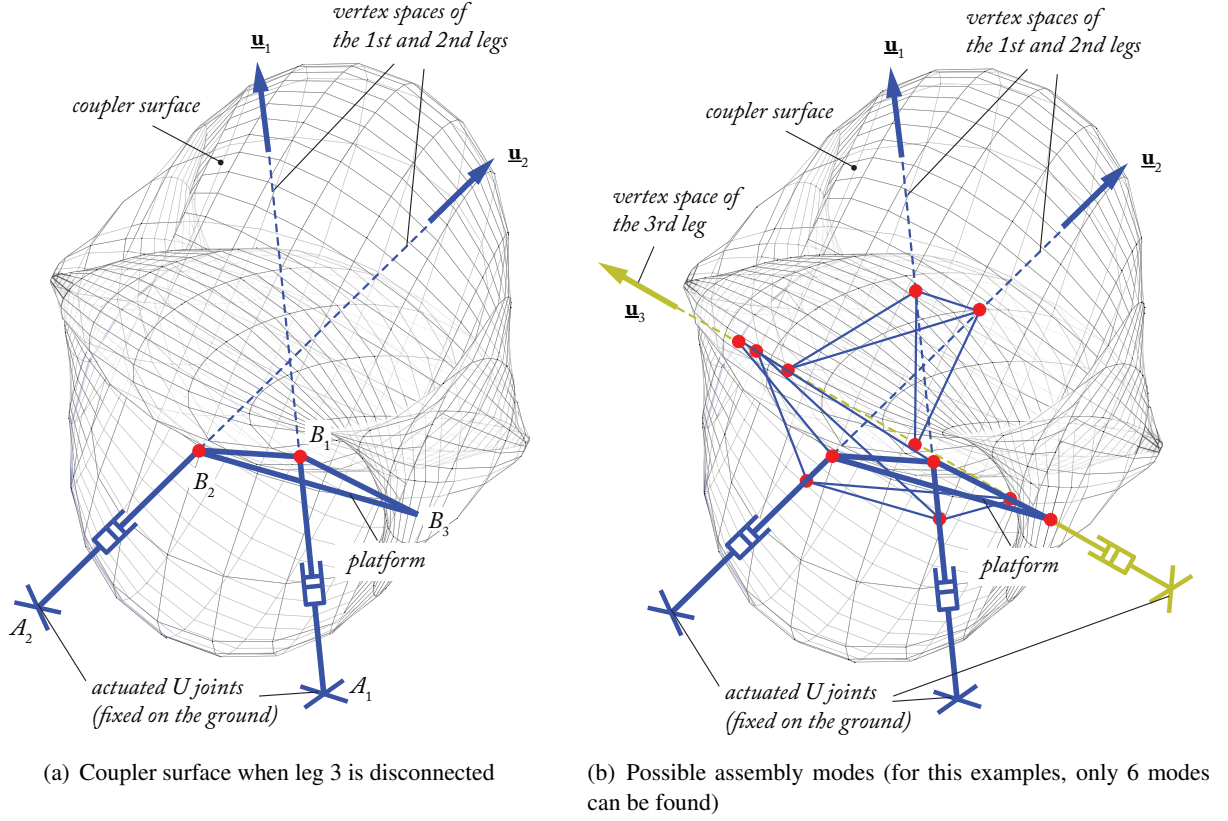


Figure 11: Solutions of the fkp for a 3- \underline{UPS} robot

407 whose directions are given by \mathbf{u}_1 and \mathbf{u}_2 , respectively (these lines represent the vertex spaces of point B_i of
 408 the legs when the U joints are fixed and the P joints are passive), and the platform is free to rotate around the
 409 line B_1B_2 . As demonstrated in [43], the surface described by point B_3 is an octic surface (Fig 11(a)), i.e. an
 410 algebraic surface of degree eight.

411 As B_3 also belongs to leg 3, this point is constrained to move on a line defined by the direction \mathbf{u}_3 of the
 412 passive prismatic joint. As shown in [43], a line and an octic surface can have up to eight real intersection
 413 points. As a result, the 3- \underline{UPS} robot can have up to eight assembly modes. An example of possible assembly
 414 modes of the 3- \underline{UPS} robot is depicted in Fig. 11. Let us recall here that, in the general case, the GS platform
 415 can have up to 40 assembly modes [2] that are different from those of the 3- \underline{UPS} robot.

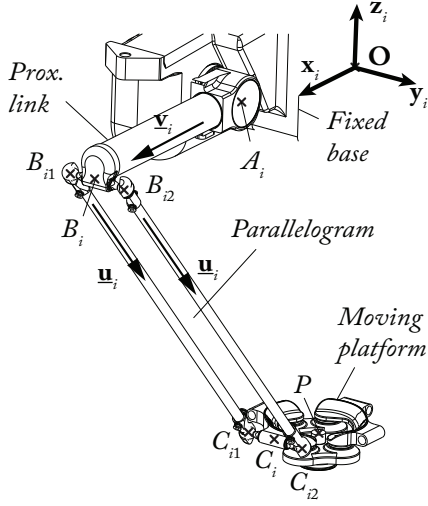
416
 417 **Singular configurations.** The singular configurations of 3- \underline{UPS} -like robots have been deeply studied in the
 418 past [34, 35]. Type 2 singularities appear when the planes $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3$ (whose normal directions are defined by
 419 the vectors $\mathbf{u}_1, \mathbf{u}_2$ and \mathbf{u}_3 , respectively) and the plane \mathcal{P}_4 (passing through the points B_1, B_2 and B_3) intersect
 420 in one point (that can be at infinity) (Fig. 10(b)).

4.2.2. The Delta-like robot family

422 The Delta-like robot family regroups the robots made of $n \underline{A}\{-2-US\}$ legs (where \underline{A} can be either an active
 423 \underline{R} or \underline{P} joint – see Fig. 12(a) for an example of $\underline{R}\{-2-US\}$ leg) [38, 44], or some of their variations such as
 424 $\underline{A}UU$ [45], or even $\underline{A}US$ legs [46, 47], i.e. the leg is composed of one active \underline{R} or \underline{P} joint located on the ground,
 425 followed by either two passive U joints, a spatial parallelogram ($2-US$ loops) or even passive U and S joints.

426 Probably the most known robots of this family are the Delta [38], the Quattro [39] (Fig. 12(b)), the Or-
 427 thoglide [40], but many other types of architectures exist (see for example [44, 48]). For such types of legs,
 428 the distal links direction can be observed (Fig. 12(a)). From Section 3.2, it can be shown that the virtual
 429 equivalent legs are:

- 430 • for $\underline{R}\{-2-US\}$ legs: a $\Pi\{-2-US\}$ or a $\Pi\{-2-UU\}$ leg (Fig. 13(a));



(a) a $\underline{R}\text{-}\{2\text{-}\underline{US}\}$ leg



(b) the Adept Quattro

Figure 12: Example of leg and of robot of the Delta-like family

- 431 • for \underline{RUU} legs: a $\Pi\underline{UU}$ leg;
- 432 • for \underline{RUS} legs: a $\Pi\underline{US}$ leg;
- 433 • for $\underline{P}\text{-}\{2\text{-}\underline{US}\}$ legs: a $\underline{P}\text{-}\{2\text{-}\underline{US}\}$ or a $\underline{P}\text{-}\{2\text{-}\underline{UU}\}$ leg;
- 434 • for \underline{PUU} legs: a \underline{PUU} leg;
- 435 • for \underline{PUS} legs: a \underline{PUS} leg;

436 i.e. the active R joints fixed on the ground are replaced by a passive planar parallelogram joint, while the active
 437 P joints becomes passive and the passive U or $R\Pi$ joints become active. Then, using the usual methodology,
 438 the fkp and singularity analysis can be carried out.

439 For illustrating this Section, let us present the fkp and singularity analysis of the hidden robot model of the
 440 Quattro with 4 dof (that can perform Schoenflies motions), when controlled using leg direction observation.

441 The Quattro is made of 4 $\underline{R}\text{-}\{2\text{-}\underline{US}\}$ legs, thus its equivalent hidden robot will be made of $\Pi\text{-}\{2\text{-}\underline{US}\}$ or
 442 $\Pi\text{-}\{2\text{-}\underline{UU}\}$ legs. As such hidden robot legs have 2 degrees of actuation (the U joint is fully actuated), only
 443 two legs have to be observed for fully controlling the Quattro using leg direction observation. However in this
 444 case, if the hidden robot has a $2\text{-}\Pi\text{-}\{2\text{-}\underline{US}\}$ architecture, the platform will have two uncontrolled dof . This
 445 phenomenon disappear if $\Pi\text{-}\{2\text{-}\underline{UU}\}$ legs are used in the hidden robot model (Fig. 13 – in this picture, the
 446 articulated platform is simplified for a clearer drawing, but has indeed the kinematic architecture presented in
 447 Fig. 14).

448
 449 **Forward kinematics and assembly modes.** Without loss of generality, let us consider that we analyze the
 450 $2\text{-}\Pi\text{-}\{2\text{-}\underline{UU}\}$ robot depicted at Fig. 13(a). Looking at the vertex space of each leg when the active \underline{U} joints
 451 are fixed, the points C_i and D_i are carrying out a circle C_i of radius $l_{A_iB_i}$ centred in S_i (Fig. 13(c)).

452 The Quattro with 4 dof , and consequently its hidden robot model, has a particularity: its platform is
 453 passively articulated (Fig. 14) so that its orientation with respect to the horizontal plan xOy stays constant,
 454 while it can have one degree of rotation around the z axis, i.e. point D_2 can describe a circle C_1 located in
 455 the horizontal plane, centred in D_1 and with a radius $l_{D_1D_2}$. For solving the forward kinematics, it is thus
 456 necessary to virtually cut the platform at point D_2 and to compute the coupler surface of point D_2 when it
 457 belongs to leg 1. This coupler surface is the surface generated by C_1 when it performs a circular translation
 458 along C_1 . Such a surface is depicted in Fig. 15(a) and is called a Bohemian Dome [49].

459 A Bohemian Dome is a quartic surface, i.e. an algebraic surface of degree 4. When it intersects the
 460 vertical plane \mathcal{P}_1 containing the circle C_2 (i.e. vertex space of the second leg), the obtained curve is a quartic

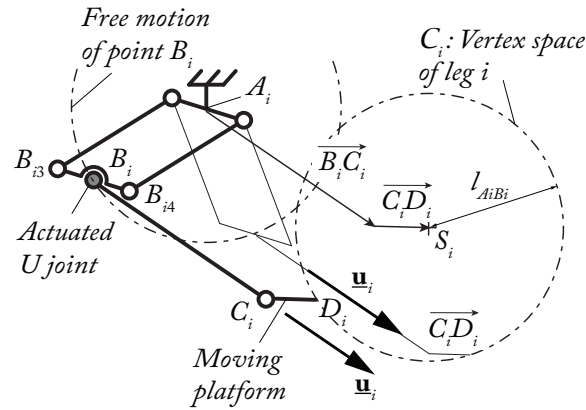
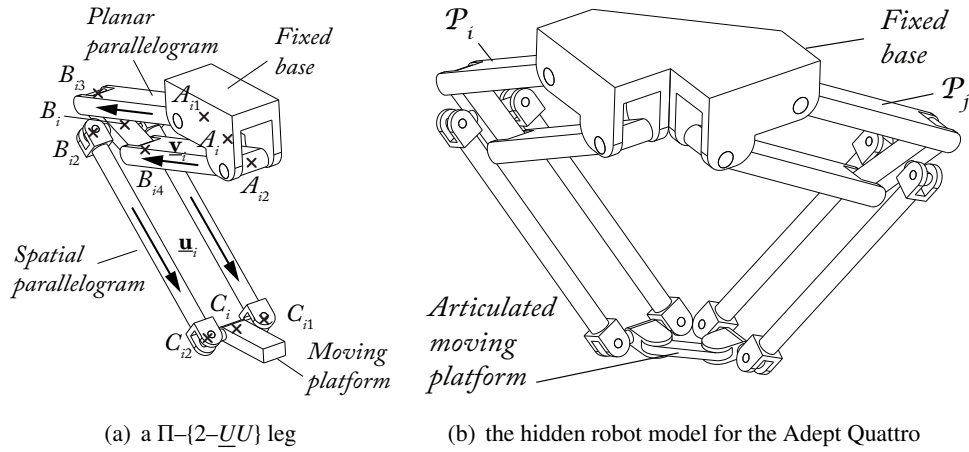


Figure 13: Example of leg and of hidden robot for the Delta-like family

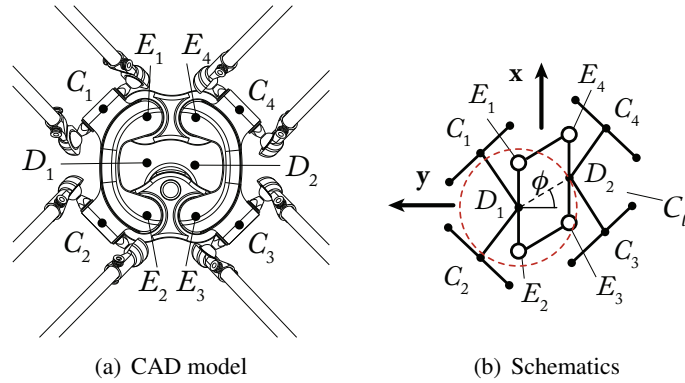
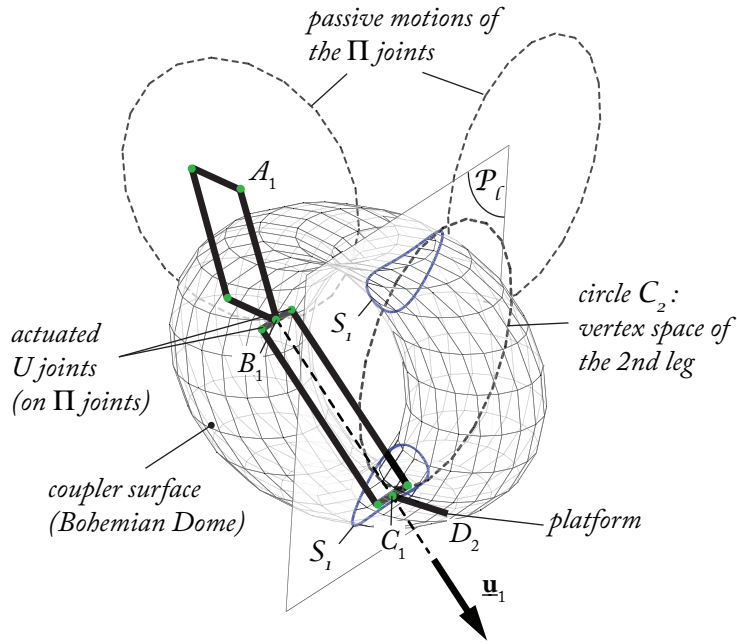


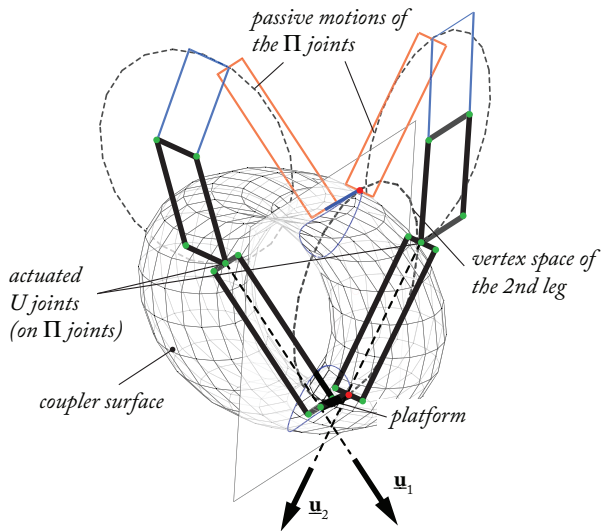
Figure 14: The platform of the Quattro

461 curve (denoted at \mathcal{S}_1 – Fig.15(a)). And using the Bézout theorem [50], it can be proven that, when the circle
 462 corresponding to the vertex space of leg 2 intersects this quartic curve, there can exist at most 8 intersection
 463 points, i.e. 8 assembly modes. Some examples of assembly modes for the $2\text{-}\Pi\text{-}\{2\text{-}\underline{UU}\}$ robot are depicted in
 464 Figs. 15(b) and 15(c).

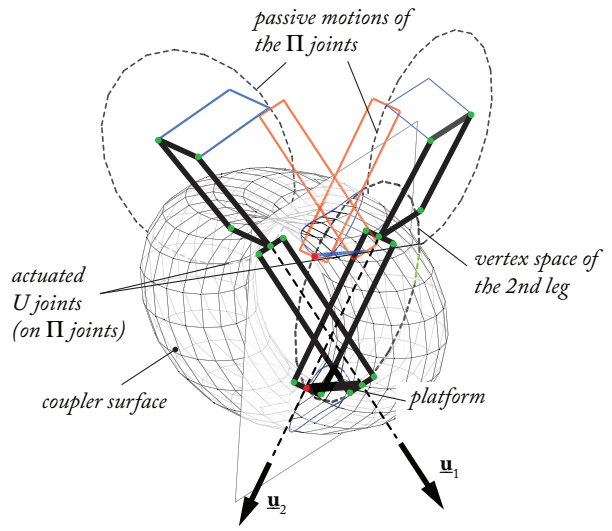
465 It should be noted that, when circles C_1 and C_2 are located in parallel planes, \mathcal{S}_1 degenerates into 1 or 2
 466 circles. In this case, the maximal number of assembly modes decreases to 4. It must be mentioned here that,
 467 in usual controllers when only the encoder data is used, the number of assembly modes of the Quattro is equal
 468 to 8.



(a) Coupler surface when leg 2 is disconnected



(b) First set of possible assembly modes



(c) Second set of possible assembly modes

Figure 15: Solutions of the fkp for a $2-\Pi-\{2-\underline{U}\}$ robot (in this example, only 4 assembly modes exist)

469

470 **Singular configurations.** For the $2-\Pi-\{2-\underline{U}\}$ robot, Type 2 singularities appear when the planes \mathcal{P}_i and \mathcal{P}_j
 471 (whose normal vectors are equal to \underline{v}_i^\perp and \underline{v}_j^\perp , resp.) are parallel. In such cases, the circle C_2 is tangent to
 472 the Bohemian Dome at their intersection point and the robot gains one uncontrollable dof along this tangent
 473 (Fig. 16).

474 4.3. Discussion

475 Thus, the concept of hidden robot model, via the use of tools for the geometric and kinematic analysis
 476 of parallel robots developed by the mechanical design community, can really help the control community to
 477 simplify the verification of the controllability (and above all, to certify the results) of parallel robots controlled
 478 using leg-direction-based visual servos. With such an approach,

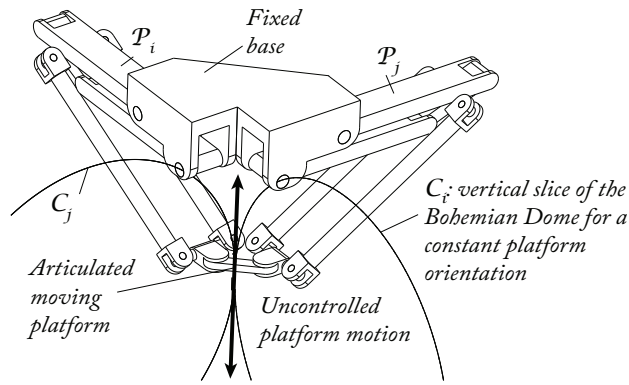


Figure 16: Example of a Type 2 singularity for a 2-Π(2-UU) robot: the platform gets an uncontrollable translation.

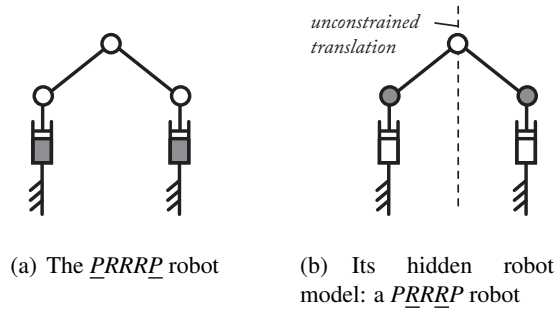


Figure 17: The $\underline{P}\underline{R}\underline{R}\underline{R}\underline{P}$ robot and its hidden robot model

479 • the problem of the correct convergence of the robot to the desired pose can be analyzed (i.e. *it was*
 480 *possible to analyze if a global diffeomorphism between the Cartesian space and the leg direction space*
 481 *exists*),

482 • as well as the problem of the singularity analysis of the mapping between the leg direction space and
 483 the Cartesian space (*including the problem of finding local minima to which the robot could converge*),

484 by reducing the problem to the kinematic analysis of another robot (in all the studied cases, it was possi-
 485 ble to find simple geometric conditions without any mathematical derivations for the *fkp* and the singular
 486 configurations).

487 It is interesting to notice that, for most of the studied robots, the hidden robots models have less assembly
 488 modes, i.e the proposed control approach based on the observation of the leg directions allows the simplifica-
 489 tion of the *fkp* for the studied architectures. This is obvious not a general assertion, but this appears for most
 490 of robots we studied.

491 Considering the singularity analysis of the mapping between the leg direction space and the Cartesian
 492 space, typical examples for which the verification of the controllability is easy to carry out without any mathe-
 493 matical derivations using the concept of hidden robot model are the cases of $\underline{P}\underline{R}\underline{R}\underline{R}\underline{P}$ robots with all \underline{P} parallel
 494 (Fig. 17(a)) and of Delta-like robots actuated via \underline{P} joints for which all \underline{P} are parallel (such as the UraneSX or
 495 the I4L [44, 48]). It was shown in [17] through the analysis of the rank deficiency of the interaction matrix that
 496 it was not possible to control such types of robots using leg direction observation. Considering this problem
 497 with the hidden robot concept is very easy. For example, in the case of the $\underline{P}\underline{R}\underline{R}\underline{R}\underline{P}$ robot with parallel \underline{P} joints,
 498 the hidden robot has a $\underline{P}\underline{R}\underline{R}\underline{R}\underline{P}$ architecture (Fig. 17(b)), where the parallel \underline{P} joints are passive. This robot
 499 is well-known to be architecturally singular as there is no way to control the translation along the axis of the
 500 parallel \underline{P} joints. This result can be easily extended to the cases of the hidden robots of the UraneSX and the
 501 I4L. More developments on the controllability analysis of parallel robots controlled using leg-direction-based
 502 visual servos can be found in [51].

503 It should finally be mentioned that in order to avoid the loss of controllability of such robot architectures,
 504 a way to proceed is to modify the controller so that the second component \underline{h}_i of the Bi-normalised Plücker

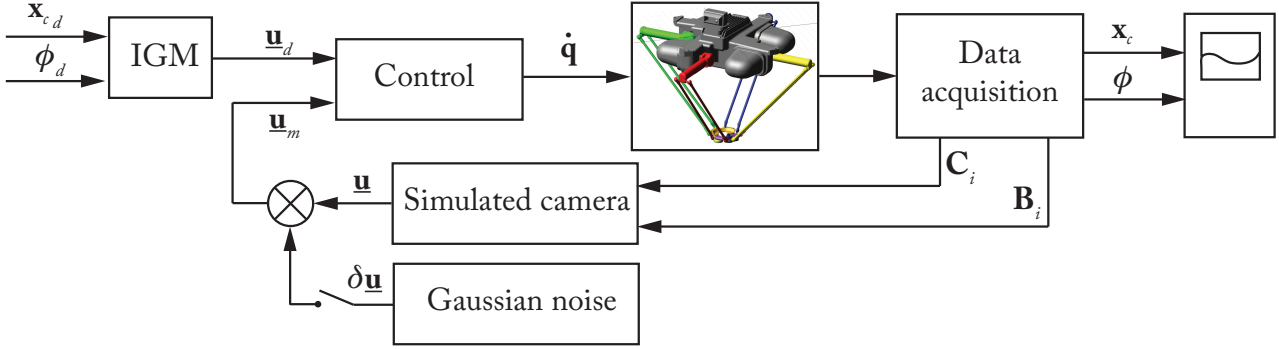


Figure 18: The controller used for the simulations

505 coordinates of the lines is considered. In such a case, other hidden robots should be found, but this is part of
 506 our future work.

507 5. Case Study

508 In this Section, simulations and experiments are performed on the Adept Quattro presented in Sec-
 509 tion 4.2.2.

510 5.1. Simulation results

511 The geometric parameters of the Quattro are given in Appendix Appendix B, as well as the equations
 512 necessary to perform the following simulations.

513 5.1.1. Description of the simulator

514 In this section, simulations are performed on an ADAMS mockup of the Adept Quattro (Fig. 18) with the
 515 same kinematic properties as the real robot by Adept. This virtual mockup is connected to Matlab/Simulink
 516 via the module ADAMS/Controls. The controller presented in Section 2.3 is applied (with $\lambda = 0.8$ – Fig. 18)
 517 in which:

- 518 • the observation of the leg is simulated by extracting in the ADAMS model the positions of the anchor
 519 points of each parallelograms,
- 520 • the actuated joint velocities are given as inputs of the ADAMS mockup.

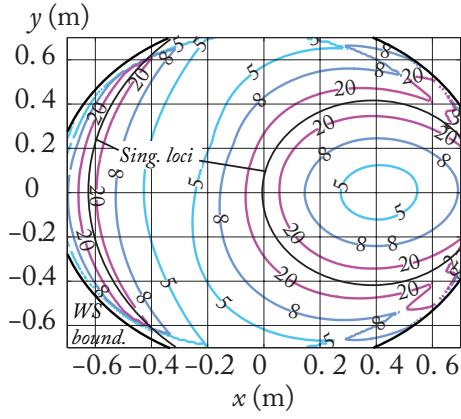
521 On the used control scheme, we can also switch on or off simulated measurement noise (whose amplitude
 522 can be parameterized) on the observation of the leg directions.

523 5.1.2. Accuracy analysis of the Quattro using leg observation

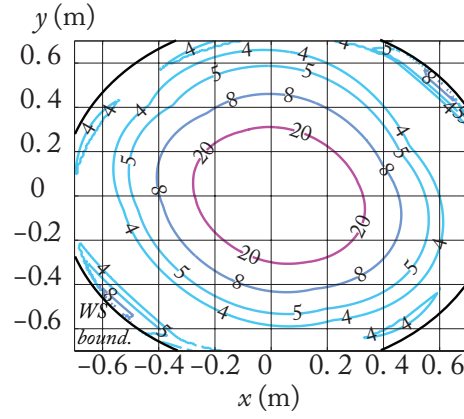
524 First, let us make the analysis of the accuracy of the Quattro using leg observation based on the accuracy
 525 model depicted in Appendix Appendix A.1. For this mechanism, in the case of a leg direction based visual
 526 servoing and for an error ${}^b\delta\mathbf{u}_i$ defined such that the vector ${}^b\mathbf{u}_i$ is contained in a cone of axis ${}^b\mathbf{u}_{i0}$ and of half
 527 angle ψ_i (${}^b\mathbf{u}_{i0}$ is the nominal value of ${}^b\mathbf{u}_i$ and, in what follows, ψ_i is taken equal to 0.1 deg for each leg
 528 direction), let us first compute the maximal positioning and orientation error when only two of its four legs
 529 are observed. Six different combinations are possible. However, the value of the error for only two of them
 530 (when legs {2, 3} and {2, 4} are observed) is plotted at Figs. 19 and 20.

531 In Figs. 19(a) and 20(a), it is possible to note that the maximal error varies very quickly, especially near
 532 singularities of the hidden robot. In Figs. 19(b) and 20(b), things are different. The variation of the accuracy
 533 is smoother for the orientation error, and the position accuracy decrease in the middle of the workspace only.
 534 Thus, it can be concluded that the selection of the legs to observe is crucial for the final pose accuracy.

535 Let us now compute the maximal positioning error when the four legs are observed. It can be observed
 536 that the position error is larger near $\{x = 0m, y = 0m, z = -0.61m, \phi = 0deg\}$. This can be explained by the

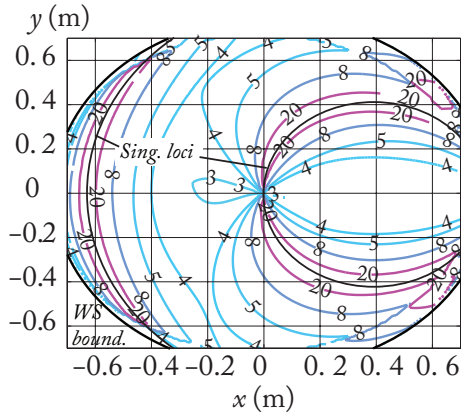


(a) legs {2, 3} are observed

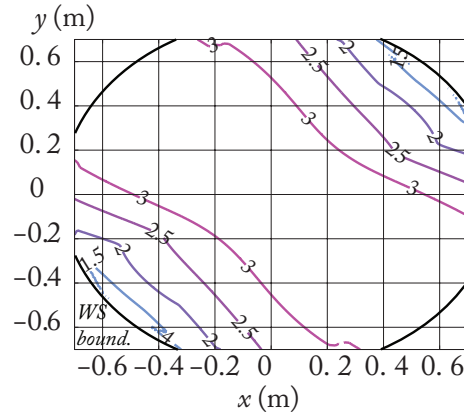


(b) legs {2, 4} are observed

Figure 19: Maximal position error (in mm) for $z = -0.7$ m and $\phi = 0$ deg. Singularity loci are singularity cases of the hidden robot associated with the leg-direction-based visual servoing of the Quattro.



(a) legs {2, 3} are observed



(b) legs {2, 4} are observed

Figure 20: Maximal orientation error (in deg) for $z = -0.7$ m and $\phi = 0$ deg. Singularity loci are singularity cases of the hidden robot associated with the leg-direction-based visual servoing of the Quattro.

537 fact that this configuration is a singularity of the model for which all the planes \mathcal{P}_i ($i = 1, 2, 3, 4$) are parallel.
 538 Thus, even if all the legs are observed, singular configurations may appear near which the accuracy is poor.
 539 Such a phenomenon shows the importance of the study of the intrinsic properties of the controller via the
 540 hidden robot concept.

541 These results will be compared with the results obtained in the following numerical simulations.

542 5.1.3. Numerical validations

543 Testing the convergence of the robot to the desired pose

544 In the first simulation, no noise is added on the simulated values of the leg directions. The initial platform
 545 pose is equal to $\{x = 0m, y = 0m, z = -0.75m, \phi = 0deg\}$ and the final desired platform pose is set to
 546 $\{x = -0.2m, y = 0m, z = -0.56m, \phi = 0deg\}$. For going from the initial point to the final ones, two sets of
 547 observed leg directions are tested: $\{1, 4\}$ and $\{2, 3\}$. For those two set of legs, solving the *fkp* of the hidden robot
 548 model of the Quattro presented in Section 4.2.2 at the desired final configuration of the robot, the following
 549 assembly modes can be obtained:

- 550 • for legs $\{1, 4\}$:
 - 551 – solution 1: $\{x = -0.2m, y = 0m, z = -0.56m, \phi = 0deg\}$

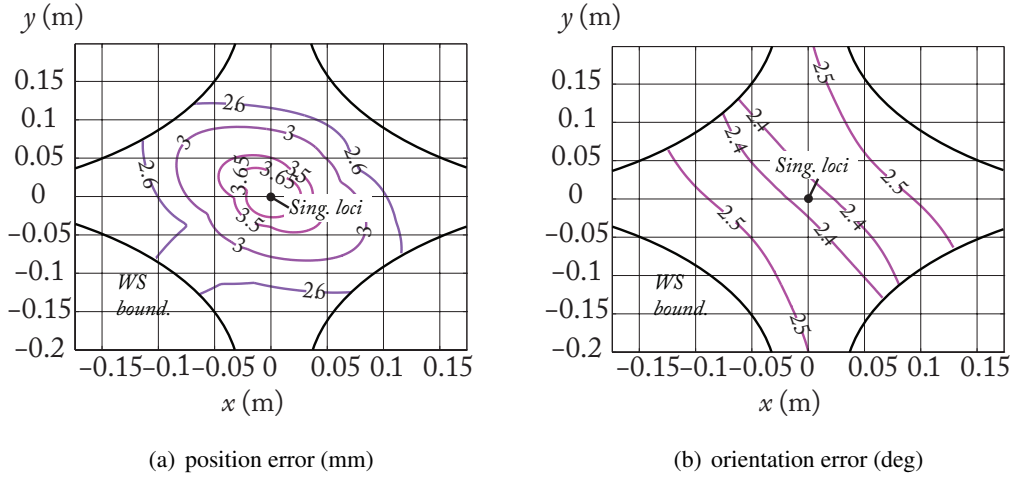


Figure 21: Maximal position and orientation error for $z = -0.61$ m and $\phi = 0$ deg when all legs are observed. Singularity loci are singularity cases of the hidden robot associated with the leg-direction-based visual servoing of the Quattro.

- 552 – solution 2: $\{x = -0.2m, y = 0m, z = -0.909m, \phi = 0deg\}$
- 553 – solution 3: $\{x = -0.138m, y = 0.062m, z = -1.019m, \phi = 0deg\}$
- 554 – solution 4: $\{x = -0.138m, y = 0.062m, z = -0.45m, \phi = 0deg\}$
- 555 • for legs {2, 3}:
 - 556 – solution 1: $\{x = -0.2m, y = 0m, z = -0.56m, \phi = 0deg\}$
 - 557 – solution 2: $\{x = -0.2m, y = 0m, z = -0.296m, \phi = 0deg\}$
 - 558 – solution 3: $\{x = -0.262m, y = 0.062m, z = -0.694m, \phi = 0deg\}$
 - 559 – solution 4: $\{x = -0.262m, y = 0.062m, z = -0.161m, \phi = 0deg\}$

560 The results for the convergence of the leg directions are presented in Fig. 22. It can be shown that when
 561 the legs {2, 3} are observed, all leg directions converge to 0. This is not true for the second case. Looking at the
 562 platform pose computed by ADAMS, the robot reach the configuration $\{x = -0.2m, y = m, z = -0.909m, \phi =$
 563 $0deg\}$, i.e. the second solution for the *fkp* of the hidden robot model of the Quattro (Fig. 23).

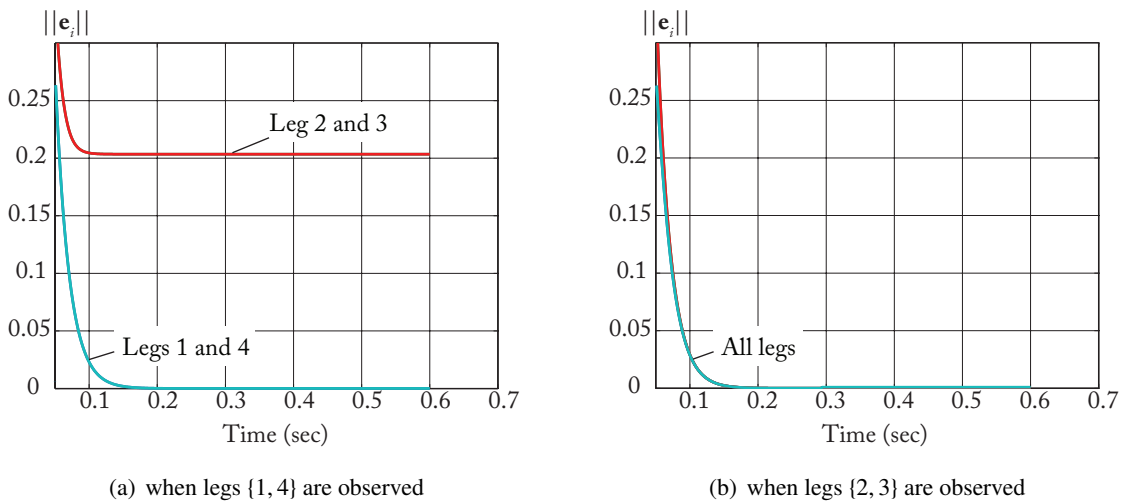


Figure 22: Error norm on each leg $\|e_i\|$.

564 A second simulation is performed in which all legs are observed. The initial platform pose is equal to
 565 $\{x = 0.05m, y = 0.05m, z = -0.8m, \phi = 0deg\}$ and the final desired platform pose is set to $\{x = 0.03m, y =$

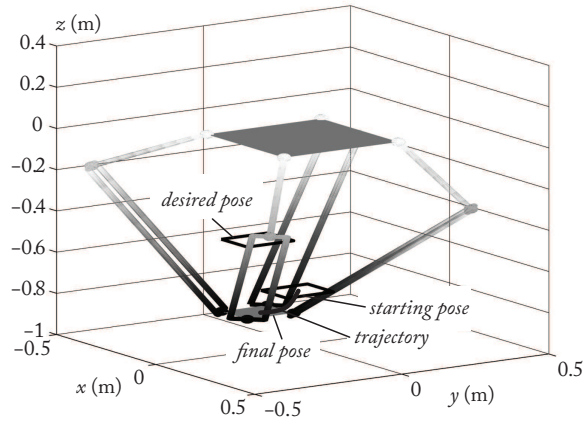


Figure 23: Initial, desired and final position when legs {1, 4} are observed

566 $0.03m, z = -0.59m, \phi = 0deg$. Solving the *fkp* of the hidden robot model of the Quattro when all legs are
 567 observed at the desired final configuration of the robot, it can be proven that there still exist two assembly
 568 modes, separated by singularities (Fig. 21) which are:

- 569 • solution 1: $\{x = 0.03m, y = 0.03m, z = -0.59m, \phi = 0deg\}$
- 570 • solution 2: $\{x = 0.03m, y = 0.03m, z = -0.65m, \phi = 0deg\}$

571 Looking at the platform pose computed by ADAMS, even if all errors on the legs vanish (Fig. 24), the
 572 robot reaches the configuration $\{x = 0.03m, y = 0.03m, z = -0.65m, \phi = 0deg\}$, i.e. the second solution for
 573 the *fkp* (Fig. 25).

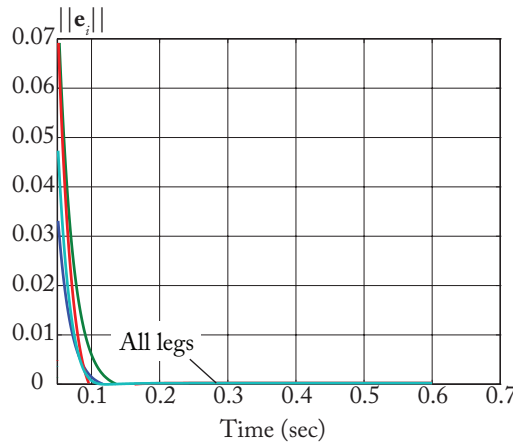


Figure 24: Error norm on each leg $\|e_i\|$ when all the legs are observed.

574 All these numerical results confirm the presence of the virtual robot hidden within the controller that must
 575 be studied in order to avoid convergence problems due to inadequate stacking of interaction matrices.

576

577 Testing the presence of local minima

578 For that simulations, all legs are observed. The initial platform pose is equal to $\{x = 0.028m, y = 0m, z =$
 579 $-0.617m, \phi = 0deg\}$ and the final desired platform pose is set to $\{x = -0.6m, y = 0m, z = -0.8m, \phi = 0deg\}$.
 580 No noise is added on the simulated values of the leg directions. After about 0.3 s of simulations, the robot
 581 stops in the configuration $\{x = -0.588m, y = 0m, z = -0.847m, \phi = 0deg\}$ while the error on the leg direction
 582 is far from zero (Fig. 26). Thus we are in the presence of a local minimum.

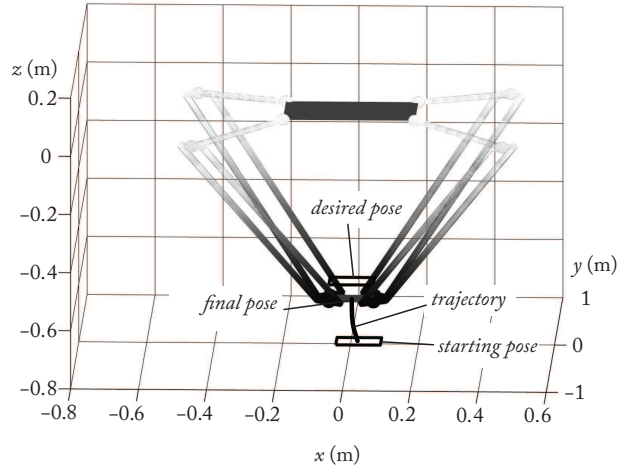
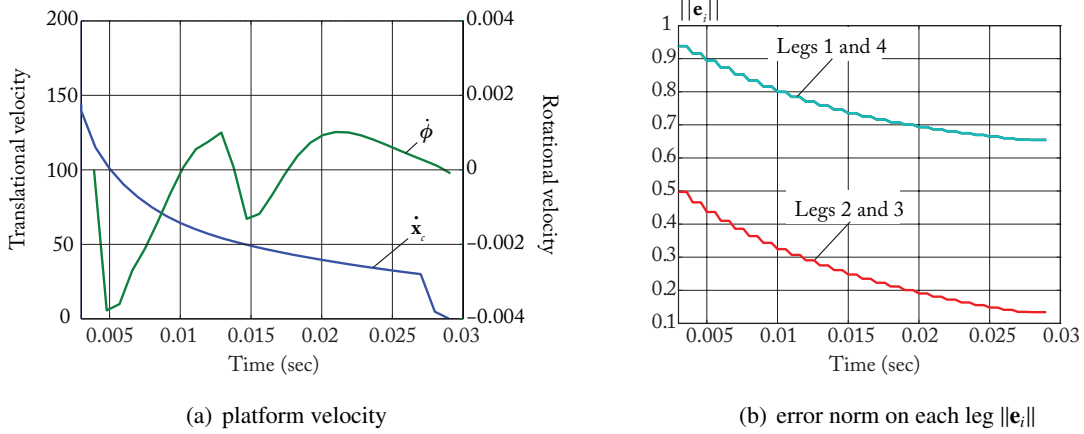


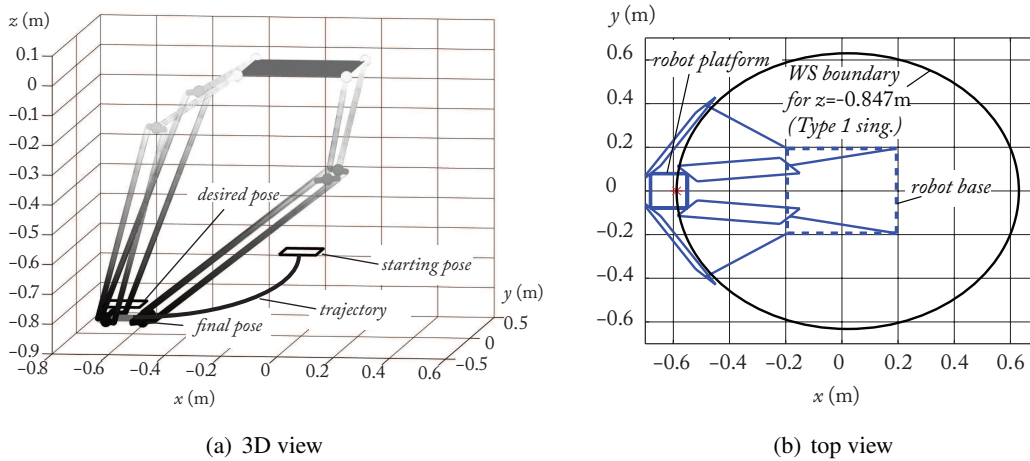
Figure 25: Initial, desired and final position when all legs are observed



(a) platform velocity

(b) error norm on each leg $\|e_i\|$

Figure 26: Platform velocity and error norm on each leg when the robot meets a local minimum.



(a) 3D view

(b) top view

Figure 27: Robot configuration when it meets a local minimum.

583 Looking at the configuration in which the robot is blocked, it appears that, as forecasted, it is a Type 1
 584 singularity (boundary of the workspace (Fig. 27)). This confirms the fact that the local minima appear in the
 585 Type 1 singularities of the hidden robot model, as mentioned in Section 3.3.

586

587 **Testing the importance of the selection of the observed legs on the robot accuracy**

588 In the first simulation, the initial platform pose is equal to $\{x = 0.02m, y = 0.1m, z = -0.7m, \phi = 0deg\}$ and
 589 the final desired platform pose is set to $\{x = -0.2m, y = 0.01m, z = -0.7m, \phi = 0deg\}$. A random noise of 0.1
 590 deg is added to the simulated measure of the leg directions. To show the importance of the leg selection on the
 591 robot accuracy, it is decided to control the robot displacement using two different sets of legs: (i) legs {2, 3}
 592 and (ii) legs {2, 4}. The results (Fig. 28) show that, as presented in Fig. 19, the final platform pose accuracy is
 593 better when legs {2, 3} are observed (around 3 mm and 0.05 rad) than with legs {2, 4} (around 7 mm and 0.07
 594 rad).

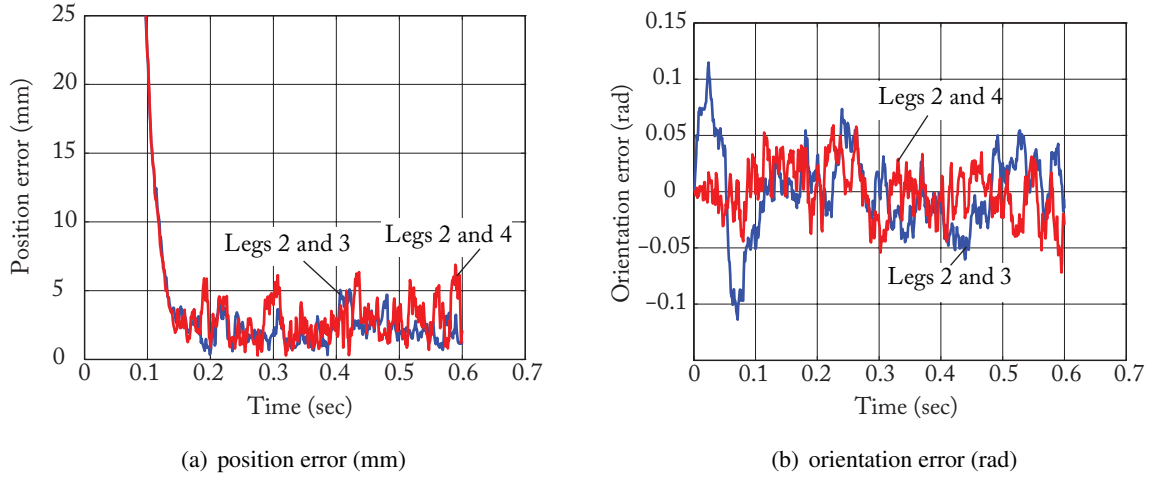


Figure 28: Orientation and position error when legs {2, 3} and legs {2, 4} are observed.

595 In the second simulation, the initial platform pose is equal to $\{x = 0.05m, y = 0.05m, z = -0.8m, \phi = 0deg\}$
 596 and the final desired platform pose is set to $\{x = 0.03m, y = 0.03m, z = -0.65m, \phi = 0deg\}$. It is decided to
 597 control the robot displacement using three different sets of legs: (i) legs {1, 4}, (ii) legs {1, 3, 4} and (iii) all legs.
 598 The results (Fig. 29) show that the final platform pose accuracy is better when all legs are observed, while
 599 the accuracy is quite the same when two or three legs are observed. However, this result must not hide the
 600 fact that, even if four legs can lead to better accuracy, some convergence problems can still appear, as shown
 601 previously.

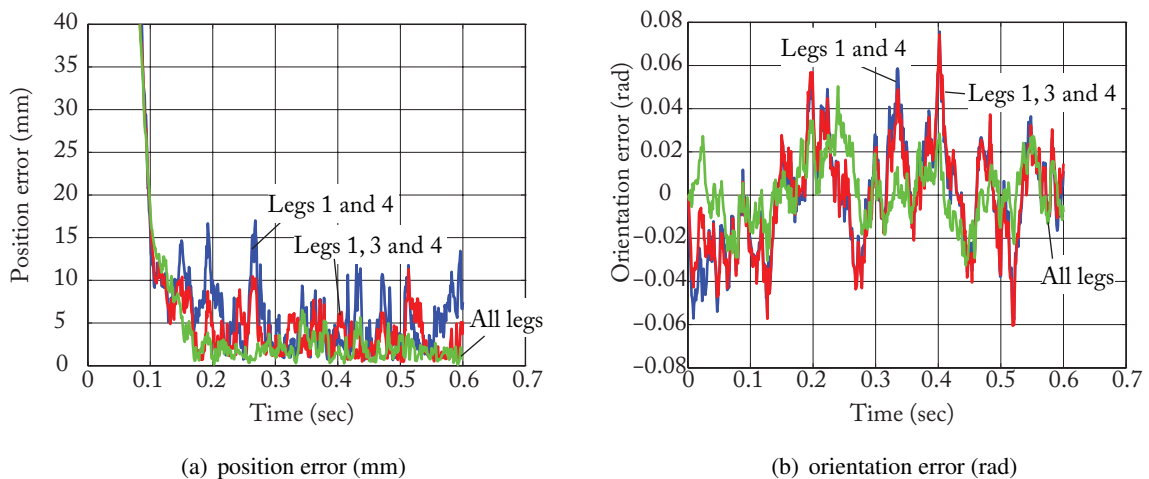


Figure 29: Orientation and position error when legs {1, 4}, {1, 3, 4} and all legs are observed.

602 All these simulations validate the theory presented in Section 3.

603 5.2. Experimental results

604 5.2.1. Description of the benchmark

605 In this section, experiments are now performed on a real Adept Quattro. The benchmark is composed of
606 (Fig. 30):

- 607 • an Adept Quattro robot bought by the Institut Pascal of Clermont-Ferrand (France),
- 608 • a camera AVT Marlin F131B firewire IEEE1394 (lens: 3.6mm 1:1.6 1/2 inch for CCD camera), which
609 is mounted at the centre of the robot base so that all the legs can be observed without any problems of
610 occlusion and whose intrinsic and extrinsic parameters have been calibrated,
- 611 • a lighting system that provides an homogenous lighting to the scene,
- 612 • a computer that extracts the data coming from the camera, computes the value of the leg directions \underline{u} ,
613 then calculates the robot actuator velocity \underline{q} using the controller of Section 2.3.2 and send the infor-
614 mation to the robot controller. Note that, in experiments, the value of λ in the controller is fixed to
615 0.2.

616 Moreover, the robot is covered by a cloth that prevents the lighting variations and guarantees the contrast
617 quality required for observing the black legs of the robot (Fig. 31).

618 Finally, it must be mentioned we have deliberately decided to use the minimal camera resolution and to
619 not undistort the image captured. The measurement noise on the leg direction is thus of about 0.1 rad, but:

- 620 • such a high noise is interesting to show the controller robustness to leg direction prediction errors,
- 621 • the noise is so high that, for analyzing the robot accuracy and measuring the distance between the
622 real and nominal robot configurations, we can directly record and use the value of the platform pose
623 predicted by the Adept Quattro controller instead of using one external measurement device (such as a
624 lasertracker).

625 5.2.2. Experimental validations

626 **Testing the convergence of the robot to the desired pose**

627 We replay now experimentally the convergence tests presented in Section 5.1.3. The starting and desired final
628 points are the same as previously. The results are presented in the Tables 2 to 4 and illustrated by the Figs. 32
629 to 34. It should be mentioned that, for cross-validating the results on those pictures, the plotted values of the
630 error norms are computed using the values of the leg directions given by the Quattro controller.

631 Due to the presence of high measurement noise, the robot can of course not converge to the final desired
632 pose. Therefore, in these Tables, information on the tolerable maximal error on the pose attained attained in
633 simulations is given. Please note that, due to the large value of the error on the measured angle, the model
634 defined in Section Appendix A.1 is no longer valuable and we have preferred to use a more refined non
635 linearized model proposed in [52].

Table 2: Results on the experiments carried out for testing the convergence of the robot when legs 1 and 4 are observed (the positions are in meter, the angles in radians).

| | |
|--|--|
| Desired final pose | $\{x = -0.2, y = 0, z = -0.56, \phi = 0\}$ |
| Final pose in simulation | $\{x = -0.2, y = 0, z = -0.91, \phi = 0\}$ |
| Tolerable position error | 0.11 m |
| Tolerable orientation error | 2.00 rad |
| Final pose in experiments | $\{x = -0.11, y = 0.01, z = -0.86, \phi = -2.15\}$ |
| Distance to the final pose in simulation | 0.10 m |
| Orient. err. w.r.t. the final pose in simulation | 2.15 rad |

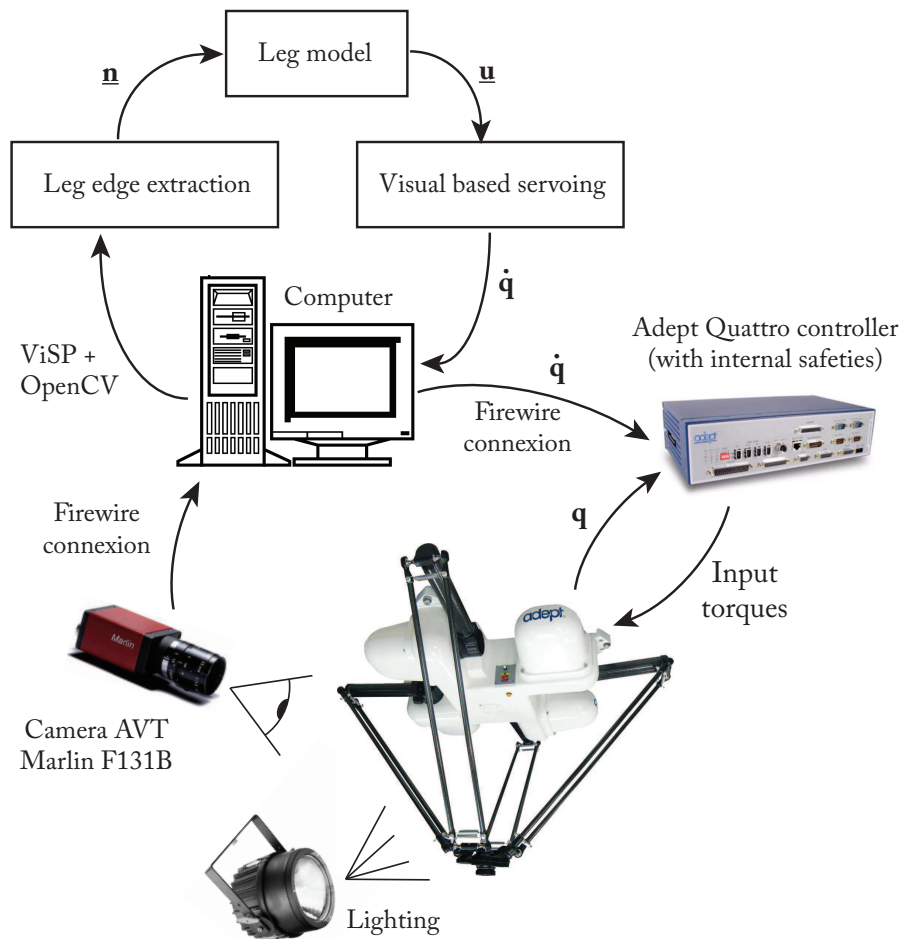


Figure 30: Experimental bench

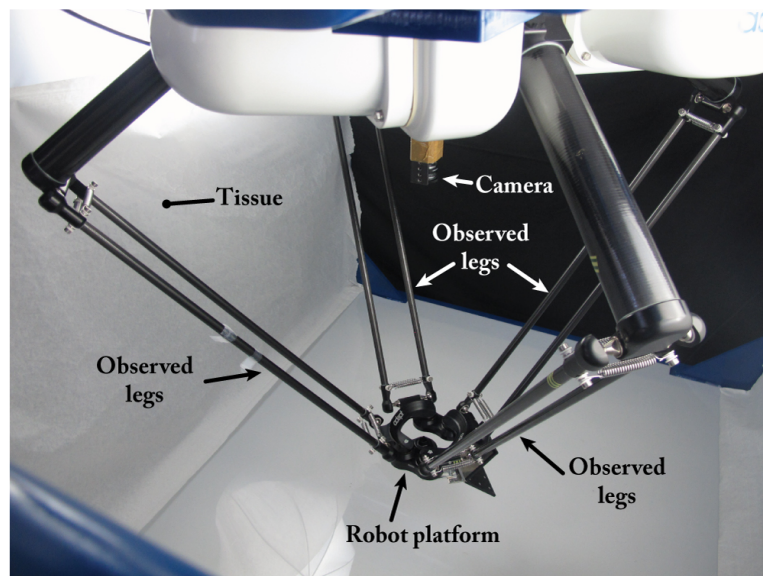
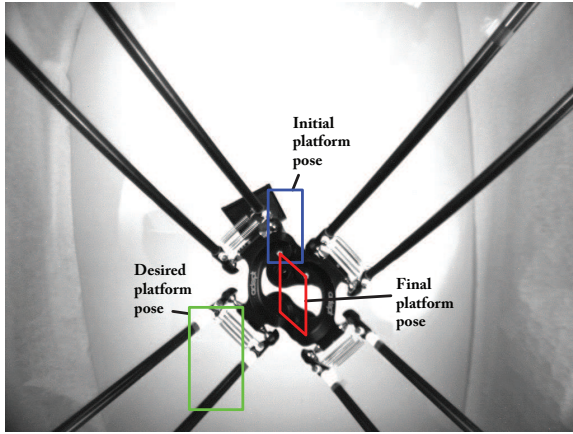
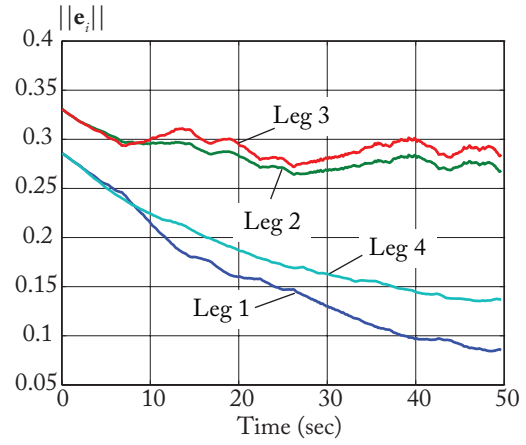


Figure 31: The Quattro recovered by the cloth

636 All these experimental results match with the simulation results presented above and confirm the presence
 637 of the virtual robot hidden within the controller that must be studied in order to avoid the convergence prob-
 638 lems due to inadequate stacking of interaction matrices.

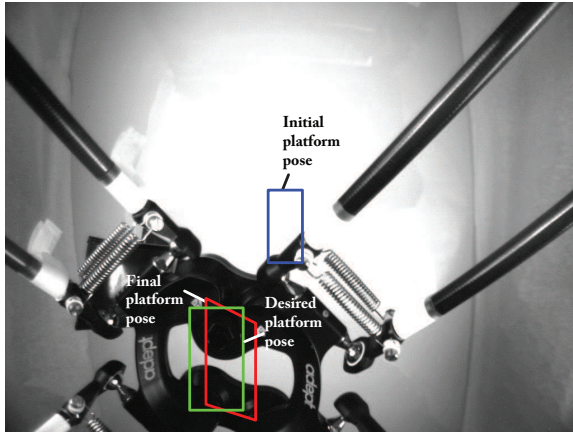


(a) top view of the platform

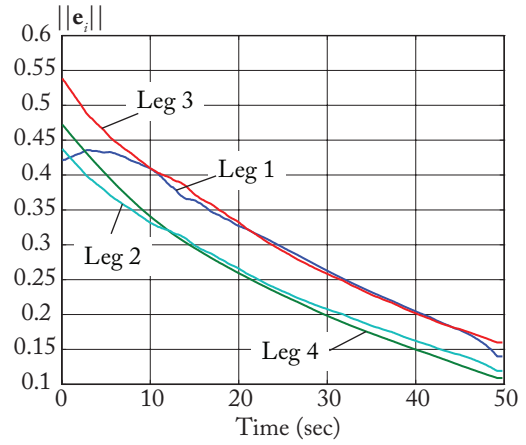


(b) error norm on each leg $\|e_i\|$

Figure 32: Convergence of the robot when legs 1 and 4 are observed (desired pose: $\{x = -0.2, y = 0, z = -0.56, \phi = 0\}$).



(a) top view of the platform



(b) error norm on each leg $\|e_i\|$

Figure 33: Convergence of the robot when legs 2 and 3 are observed (desired pose: $\{x = -0.2, y = 0, z = -0.56, \phi = 0\}$).

Table 3: Results on the experiments carried out for testing the convergence of the robot when legs 2 and 3 are observed (the positions are in meter, the angles in radians).

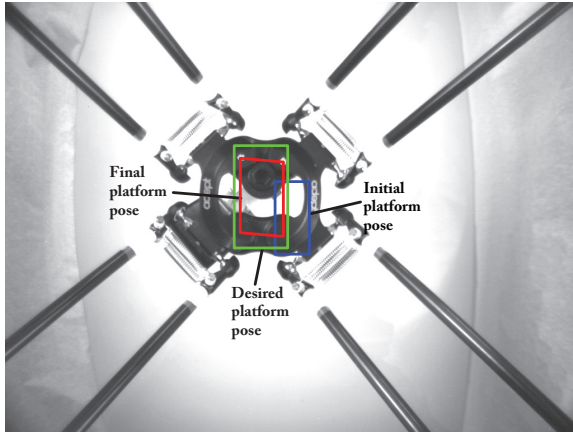
| | |
|--|--|
| Desired final pose | $\{x = -0.2, y = 0, z = -0.56, \phi = 0\}$ |
| Final pose in simulation | $\{x = -0.2, y = 0, z = -0.56, \phi = 0\}$ |
| Tolerable position error | 0.23 m |
| Tolerable orientation error | 1.23 rad |
| Final pose in experiments | $\{x = -0.12, y = 0.05, z = -0.55, \phi = -0.90\}$ |
| Distance to the final pose in simulation | 0.10 m |
| Orient. err. w.r.t. the final pose in simulation | 0.90 rad |

639

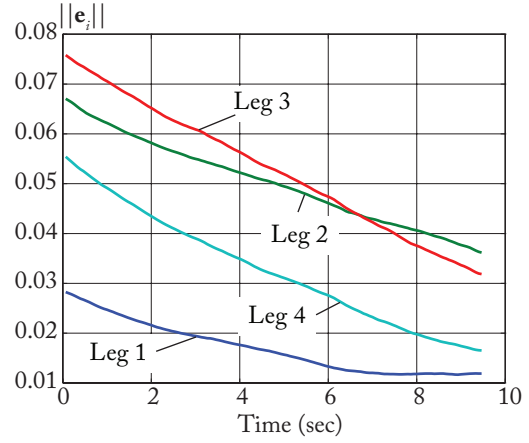
640 Testing the presence of local minima

641 Unfortunately, we were not able to do such experiments as the robot controller is designed with safeties that
 642 cannot be suppressed and that prevent going into singularities. However, as the presence of local minima that
 643 are located in the Type 1 singularities was demonstrated in simulations, we think that this numerical proof
 644 brings enough strength to our demonstration concerning this point.

645



(a) top view of the platform



(b) error norm on each leg $\|e_i\|$

Figure 34: Convergence of the robot when all legs are observed (desired pose: $\{x = 0.03, y = 0.03, z = -0.59, \phi = 0\}$).

Table 4: Results on the experiments carried out for testing the convergence of the robot all legs are observed (the positions are in meter, the angles in radians).

| | |
|--|--|
| Desired final pose | $\{x = 0.03, y = 0.03, z = -0.59, \phi = 0\}$ |
| Final pose in simulation | $\{x = 0.03, y = 0.03, z = -0.65, \phi = 0\}$ |
| Tolerable position error | 0.08 m |
| Tolerable orientation error | 1.54 rad |
| Final pose in experiments | $\{x = 0.05, y = 0.03, z = -0.72, \phi = 0.05\}$ |
| Distance to the final pose in simulation | 0.07 m |
| Orient. err. w.r.t. the final pose in simulation | 0.05 rad |

646 Testing the importance of the selection of the observed legs on the robot accuracy

647 We replay now experimentally the accuracy tests presented in Section 5.1.3. The starting and desired final
 648 points are the same as previously, as well as the observed legs. Each experiment is run five times and we
 649 present here the maximal values obtained on the position and orientation error. The results are shown in the
 650 Tables 5 to 6.

Table 5: Results on the experiments carried out for testing the accuracy of the robot when legs $\{2, 3\}$ or $\{2, 4\}$ are observed.

| Desired final pose | $\{x = -0.2m, y = 0.01m, z = -0.7m, \phi = 0deg\}$ | |
|--------------------|--|------------|
| Legs | $\{2, 3\}$ | $\{2, 4\}$ |
| Position error | 0.11 m | 0.23 m |
| Orientation error | 0.06 rad | 0.68 rad |

Table 6: Results on the experiments carried out for testing the accuracy of the robot when legs $\{1, 4\}$, $\{1, 3, 4\}$ or $\{1, 2, 3, 4\}$ are observed.

| Desired final pose | $\{x = 0.03m, y = 0.03m, z = -0.65m, \phi = 0deg\}$ | | |
|--------------------|---|---------------|------------------|
| Legs | $\{1, 4\}$ | $\{1, 3, 4\}$ | $\{1, 2, 3, 4\}$ |
| Position error | 0.11 m | 0.09 m | 0.07 m |
| Orientation error | 0.39 rad | 0.31 rad | 0.05 rad |

651 Once again, all these experimental results match with the simulation results presented above and confirm
 652 the necessity to carefully select the set of legs to observe in order to obtain the best accuracy possible. How-
 653 ever, it must be recalled that, even if observing all the legs lead to a better accuracy, this result must not hide

654 the fact that some convergence problems can still appear, as shown previously.

655

656 All these experiments validate the theory presented in Section 3.

657 5.3. Discussions and future works

658 All the results presented above show the validity of the approach and also its importance: stacking several
659 interaction matrices to derive a control scheme without doing a deep analysis of the intrinsic properties of
660 the controller is clearly not enough. For avoiding the singularity problem due to the mapping between the
661 robot space and leg space, whatever the number of observed legs (as, even if all legs are observed, there may
662 be singularities of the mapping), the hidden robot kinematics must be analyzed to avoid the convergence and
663 inaccuracy problems.

664 We would also like to add that, in this paper, we have deliberately chosen not to define the controller in
665 the image space. There exist several reasons which justify that choice:

- 666 • the proof that the robot can be controlled using only the observation of the leg edges has been given
667 in [15] and gives similar results as for the control with the leg directions, except that the robot accuracy
668 was better. This can be explained by the fact that, as mentioned previously, the leg edges are only used
669 as a measure of $\underline{\mathbf{u}}_i$. So *the problem is the same*, except in the fact that we must consider the singularity
670 of the mapping between the edges and the leg directions, but this problem is well handled [15].
- 671 • in our benchmark, it can be shown that the variations of the vectors ${}^c \underline{\mathbf{n}}_i^j$ is very small in the whole
672 workspace. The difference on vectors ${}^c \underline{\mathbf{n}}_i^j$ for two distinct robot configurations – whatever they are – is
673 about 0.01 rad. So we must be sure that the measurement noise is much lower than this value by refining
674 the data extraction from the image (e.g. by using some subpixellic approach) and this is not currently
675 our main goal. Moreover, this small variation on the values of ${}^c \underline{\mathbf{n}}_i^j$ make us think that:
 - 676 – we should modify the controller so that the second component of the Bi-normalised Plücker co-
677 ordinates of the lines corresponding to the edges is considered in order to get all the information
678 contained in the image (direction and position of the lines). In such a case, other hidden robots
679 should be found,
 - 680 – as the Quattro was not designed to be controlled using such a visual servoing approach, it may be
681 unrealistic to try to obtain good results in terms of accuracy using vision. However, an interesting
682 point would be, before the robot is designed, to take into account the control approach that will
683 be used so that the leg positions (and as a result the robot architecture) can be optimized w.r.t. the
684 desired visual servoing approach.

685 These two points are parts of our future work.

686 It should also be mentioned that, in the present paper, we have deliberately focused on the study of the
687 hidden robot models of several classes of parallel robots. This work is crucial in order to simplify the use of
688 this tool by control engineers willing to develop of leg-observation-based controller. Others challenges are to
689 use the hidden robot model tool:

- 690 • in order to propose a classification of the robots which cannot or can be controlled with leg-observation-
691 based controllers and also.
- 692 • in order to help the control engineer to understand what types of information are missing in order to
693 ensure the controllability of their robots.

694 Insights on these two last points can be found in [51].

695 6. Conclusions

696 This paper has presented a tool named the “Hidden robot concept” that is well addressed for analyzing
697 the controllability of parallel robots in leg-observation-based visual servoing techniques. It has been shown
698 that the mentioned visual servoing techniques involves the existence of a virtual robot model, hidden into the
699 controller, that is different from the real controlled robot. Considering this hidden robot model allowed the
700 finding of a minimal representation for the leg-observation-based control of the studied robots that is linked
701 to a virtual hidden robot which is a tangible visualization of the mapping between the observation space and
702 the real robot Cartesian space. It has been shown that the hidden robot model can be used to:

- 703 1. explain why the observed robot which is composed of n legs could be controlled in most cases using
704 the observation of only m leg directions ($m < n$), knowing the fact that the minimal number of observed
705 legs should be, for 3D unit vectors, an integer greater than $n/2$,
- 706 2. prove that there does not always exist a global diffeomorphism between the Cartesian space and the leg
707 direction space,
- 708 3. simplify the singularity analysis of the mapping between the leg direction space and the Cartesian space
709 by reducing the problem to the singularity analysis of a new robot,
- 710 4. certify that the robot will not converge to local minima, through the application of tools developed for
711 the singularity analysis of robots.

712 A general way to find the find the hidden robot models corresponding to the real robot controlled via
713 leg-observation-based visual servoing techniques has been shown and the hidden robot models of some well
714 known classes of parallel robots have been studied. It has been proven that, using this concept, it is possible
715 to demonstrate, using tools developed by the mechanical design community, that the robot can be controlled
716 or not with the aforementioned visual servoing techniques.

717 Finally, numerical simulations and experimental validations made on an Adept Quattro robot have demon-
718 strated the validity of the theoretical developments.

719 Thus, the concept of hidden robot model, associated with mathematical tools developed by the mechanical
720 design community, is a powerful tool able to analyze the intrinsic properties of some controllers developed
721 by the visual servoing community. Moreover, this concept showed that in some visual servoing approaches,
722 stacking several interaction matrices to derive a control scheme without doing a deep analysis of the intrinsic
723 properties of the controller is clearly not enough. Further investigations are required.

724 Appendix A. Selection of the controlled legs

725 Depending on the chosen interaction matrices, i.e. on the choice of the observed legs, the geometry of
726 the hidden robot models will vary, as well as its singularities and assembly modes. As singularities divide
727 the workspace into distinct aspects [2], it is necessary to study the motion feasibility by selecting a set of legs
728 that can allow the robot displacement. Moreover, even if the motion is feasible, if the robot goes close to a
729 singularity, the positioning error can considerably grow.

730 Therefore, it is necessary to find the best set of legs to observe in order to get the best performances of the
731 robot w.r.t. a desired task. This is the main goal of this appendix.

732 Appendix A.1. Definition of criteria for selecting the legs to observe

733 Several indices can be used for characterizing the neighborhood of singularities as well as the robot perfor-
734 mances (e.g. the condition number, the dexterity [53], etc.). Here, as generally the visual servoing is used for
735 improving the robot accuracy, it is proposed to use accuracy as an index for the characterization of singularity
736 proximity. Obviously, other criteria could be chosen, but the global approach for selecting the legs will remain
737 the same.

738 For characterizing the performances of the robot in terms of accuracy, the following model can be used.
739 From (7), and using the first order approximation of the forward geometric model [53], it is possible to write

$$740 \quad {}^c\delta\mathbf{p} = \mathbf{M}^{T+} {}^c\delta\mathbf{u} \tag{A.1}$$

741 where ${}^c\delta\mathbf{p}$ is the platform positioning error (expressed in the camera frame), ${}^c\delta\mathbf{u}$ is the error on the observation
 742 of the leg direction (expressed in the camera frame), and \mathbf{M}^{T+} is the pseudo-inverse of the matrix \mathbf{M}^T that can
 743 be obtained by stacking the matrices \mathbf{M}_i^T of the observed legs. Obviously, when using the leg direction based
 744 visual servoing, this matrix is the Jacobian matrix of the hidden robot and, as a result, will degenerate near the
 745 singular configurations presented in Section 3.

746 The error on the platform can be expressed in the robot frame using:

$$747 \quad {}^b\delta\mathbf{p} = {}^b\overline{\mathbf{R}}_c {}^c\delta\mathbf{p} = {}^b\overline{\mathbf{R}}_c \mathbf{M}^{T+} {}^c\delta\mathbf{u} \quad (\text{A.2})$$

748 where

$$749 \quad {}^b\overline{\mathbf{R}}_c = \begin{bmatrix} {}^b\mathbf{R}_c & \mathbf{0} \\ \mathbf{0} & {}^b\mathbf{R}_c \end{bmatrix} \quad (\text{A.3})$$

750 in which ${}^b\mathbf{R}_c$ is the 3×3 rotation matrix between the base and camera frames.

751 It should be mentioned here that it is decided to develop a simple model for computing the robot accuracy,
 752 but any other more realistic models can be used (e.g. non linearized models [52], models that take into account
 753 flexibilities [54], clearances [55], etc.). However, this model is enough for giving a global idea the accuracy
 754 problems of the controlled robots, as was done in Section 5.

755 Once the performance criteria are chosen, it is necessary to define a methodology that can help select the
 756 legs to observe. Such a methodology is developed below.

757 *Appendix A.2. Proposition of a methodology for selecting the legs to observe*

758 The Sections 4 and Appendix A.1 showed the importance of the legs chosen for the control scheme: the
 759 singularities of the hidden robots depend on the chosen observed legs and the accuracy will be poor near
 760 them. Several questions naturally arise here. The first one concerns the number of legs to observe. In terms of
 761 accuracy, it is obvious that observing more legs than the minimal requested number, i.e. adding measurement
 762 redundancy, will tend to improve the pose accuracy of the robot. However:

- 763 • increasing the number of legs to observe leads to an increase of the computational time and may be
 764 applied with difficulty when high sampling periods are required. Thus, a compromise must be found
 765 between the sampling period and the computational time for any given application.
- 766 • it will be shown in Section 5 that, even when all legs are observed, some singular configurations may
 767 still exist, with areas of poor robot accuracy around them.

768 The second question is about the selection of the legs to observe. For example, in the case of the Quattro
 769 (Section 4.2.2), with only two legs among four to observe, six different $2\text{-}\Pi\text{-}\{2\text{-}\underline{UU}\}$ robots can be defined.
 770 What is thus the best hidden robot model to use?

771 If the control law proposed in Section 2 is applied, it is first necessary to guarantee that, for the used set
 772 of legs:

- 773 • obviously, the legs must be observable during the whole robot displacement.
- 774 • the initial and final robot configurations must be included in the same aspect. If not, the controller will
 775 not be able to converge to the desired end-effector pose, even if the observed leg directions do. In this
 776 last case, the problem can be solved by applying special trajectories that cross Type 2 singularities [56].

777 One problem here is to check that two robot configurations belong to the same aspect. This problem is
 778 complex for most of robots, but can be solved using some advanced tools such as Interval Analysis (IA) [2] or
 779 Cylindrical Algebraic Decomposition (CAD) [57].

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

- 782 1. knowing the leg orientations at the initial and final robot configurations, compute the solutions of the
 783 fkp of the hidden robots,

- 784 2. find, using the advanced tools mentioned previously, the solutions of the fkp that belong to the same
785 assembly mode; if, for one given hidden robot, initial and final platform configurations do not belong to
786 the same aspect, discard it; if no hidden robot exists for which initial and final configurations belong to
787 the same aspect, the displacement is not feasible, except if special trajectories are planned as mentioned
788 previously,
789 3. for all remaining virtual hidden robots, knowing the observation errors $\delta \mathbf{u}$, compute the positioning
790 error using (A.2); retain the set of legs that guarantee the best accuracy;
791 4. test the controller (in simulation) with the retained set of legs; if there is no problem of convergence and
792 that the legs are observable during the whole displacement, the problem is solved; if not, discard this
793 set of leg and redo point 3; if no hidden robot exists for which initial and final configurations belong to
794 the same aspect, the displacement is not feasible, except with special trajectories.

795 Obviously, this methodology can be extended to any number of observed legs or modified by weighting
796 the interaction matrices to obtain better robot properties. One should also be aware that instead of giving
797 the initial and final robot configurations to the controller, it is better to define a trajectory between these two
798 points in order to avoid crossing singularities inadvertently. In such a case, it is possible to check (numerically
799 using any closeness-to-singularity criteria [53] or algebraically with *CAD* [57]) that the robot does not cross a
800 singularity on the trajectory.

801 Appendix B. Kinematics of the Quattro

802 Appendix B.1. Usual inverse kinematics of the Quattro

803 The following notations are used:

- 804 • point B_i (C_i , resp.) is at the middle of segment $B_{i1}B_{i2}$ ($C_{i1}C_{i2}$, resp.) (Fig. 12(a)),
- 805 • point P , the controlled point of the platform, is the barycenter of points C_i ; its coordinates are denoted
806 as \mathbf{x}_c and its velocity as τ_c ,
- 807 • the platform orientation is parameterized by the angle ϕ between the axis \mathbf{x} of the robot base frame and
808 the vector $\overrightarrow{D_1D_2}$,
- 809 • \mathbf{A}_i (\mathbf{B}_i , \mathbf{C}_i , resp.) is the vector of coordinates of point A_i (B_i , C_i , resp.),
- 810 • q_i is the angular coordinate of the actuator i , and is defined as the angle between the axis \mathbf{x}_i (the projec-
811 tion of vector $\overrightarrow{A_iB_i}$ in the horizontal plane (\mathbf{Oxy})) and $\overrightarrow{A_iB_i}$ around \mathbf{y}_i (Fig. 12(a)),
- 812 • l_1 is the length of the proximal link, and l_2 the length of one rod of the parallelogram,

813 It is to be noticed that the Adept Quattro has the following geometric characteristics:

- 814 • $l_1 = 0.380\text{m}$, $l_2 = 0.825\text{m}$,
- 815 • ${}^b\mathbf{A}_i = 0.275 [\cos \theta_i \ \sin \theta_i \ 0]^T$ (in meters), where $\theta_i = \{-3\pi/4, -\pi/4, \pi/4, 3\pi/4\}$ (in radians)
- 816 • ${}^b\overrightarrow{D_1C_1} = [-0.066 \ -0.048 \ 0]^T$, ${}^b\overrightarrow{D_1C_2} = [0.066 \ -0.048 \ 0]^T$, ${}^b\overrightarrow{D_2C_3} = [0.066 \ 0.048 \ 0]^T$ and ${}^b\overrightarrow{D_2C_4} =$
817 $[-0.066 \ 0.048 \ 0]^T$ (in meters),
- 818 • ${}^b\overrightarrow{E_1D_1} = \overrightarrow{E_4D_2} = [0.057 \ 0 \ 0]^T$, ${}^b\overrightarrow{E_2D_1} = \overrightarrow{E_3D_2} = [-0.057 \ 0 \ 0]^T$ (in meters),
- 819 • ${}^b\overrightarrow{PE_2} = 0.043 [\sin(\phi + \pi/2) \ -\cos(\phi + \pi/2) \ 0]^T$, ${}^b\overrightarrow{PE_2} = 0.043 [\sin(\phi + \pi/2) \ \cos(\phi + \pi/2) \ 0]^T$ (in me-
820 ters).

821 Moreover, the superscript ‘ i ’ will be used before the vectors to indicate that the vector coordinates are ex-
 822 pressed in the leg local frame ($\mathbf{Ox}_i\mathbf{y}_i\mathbf{z}_i$). If no superscript is used, the vector is expressed in the base frame.

823 The usual inverse kinematics of the Quattro can be computed using the following loop-closure equations
 824 (Fig. 12(b)):

$$825 \quad {}^i\mathbf{C}_i - {}^i\mathbf{B}_i = l_2 {}^i\mathbf{u}_i \quad (\text{B.1})$$

826 where

$$827 \quad {}^i\mathbf{B}_i = {}^i\mathbf{A}_i + l_1 \begin{bmatrix} \cos q_i & 0 & \sin q_i \end{bmatrix}^T = {}^i\mathbf{A}_i + l_1 {}^i\mathbf{v}_i \quad (\text{B.2})$$

828 and

$$829 \quad {}^i\mathbf{C}_i = {}^i\mathbf{x}_c + {}^i\overrightarrow{PC}_i \quad (\text{B.3})$$

830 Squaring both sides of (B.1) and introducing (B.2) leads to

$$831 \quad (x_{A_iC_i} - l_1 \cos q_i)^2 + y_{A_iC_i}^2 + (z_{A_iC_i} - l_1 \sin q_i)^2 - l_2^2 = 0 \quad (\text{B.4})$$

832 where ${}^i\mathbf{C}_i - {}^i\mathbf{A}_i = [x_{A_iC_i}, y_{A_iC_i}, z_{A_iC_i}]^T$. (B.4) can be finally solved as a second order polynomial in $\tan(q_i/2)$ by
 833 replacing $\cos q_i$ by $(1 - t_i^2)/(1 + t_i^2)$ and $\sin q_i$ by $2t_i/(1 + t_i^2)$, where $t_i = \tan(q_i/2)$. Skipping all mathematical
 834 derivations, it comes that:

$$835 \quad q_i = 2 \tan^{-1} \left(\frac{-\beta_i \pm \sqrt{\alpha_i^2 + \beta_i^2 - \gamma_i^2}}{\gamma_i - \alpha_i} \right) \quad (\text{B.5})$$

836 where

$$837 \quad \alpha_i = -2l_1 x_{A_iC_i}, \beta_i = -2l_1 z_{A_iC_i} \quad (\text{B.6})$$

$$838 \quad \gamma_i = x_{A_iC_i}^2 + y_{A_iC_i}^2 + z_{A_iC_i}^2 + l_1^2 - l_2^2 \quad (\text{B.7})$$

839 The first-order kinematics that relates the platform translational velocity $\boldsymbol{\tau}_c$ to the actuator velocities can
 840 be obtained through the differentiation of (B.4) with respect to time and can be expressed as:

$$841 \quad \mathbf{A}\boldsymbol{\tau}_c + \mathbf{B}\dot{\mathbf{q}} = \mathbf{0} \quad (\text{B.8})$$

842 where the i -th line of \mathbf{A} can be written as

$$843 \quad \mathbf{a}_i = l_2 {}^i\mathbf{u}_i^T \quad (\text{B.9})$$

844 and \mathbf{B} is a diagonal matrix whose i -th diagonal term is

$$845 \quad b_i = l_1 l_2 {}^i\mathbf{u}_i^T {}^i\mathbf{v}_i^\perp, {}^i\mathbf{v}_i^\perp = \begin{bmatrix} -\sin q_i & 0 & \cos q_i \end{bmatrix}^T \quad (\text{B.10})$$

846 It should be mentioned that \mathbf{A} is a (4×3) rectangular matrix. As a result,

$$847 \quad \dot{\mathbf{q}} = -\mathbf{B}^{-1} \mathbf{A}\boldsymbol{\tau}_c = \mathbf{J}_{inv} \boldsymbol{\tau}_c, \text{ or also } \boldsymbol{\tau}_c = \mathbf{J}_{inv}^+ \dot{\mathbf{q}} \quad (\text{B.11})$$

848 where \mathbf{J}_{inv}^+ is the pseudo-inverse of \mathbf{J}_{inv} .

849 Appendix B.2. Kinematics of the Quattro using leg observation

850 The servoing of the Adept Quattro robot using leg observation proposes to observe the parallelogram
851 direction $\underline{\mathbf{u}}_i$ to control the robot displacements. $\underline{\mathbf{u}}_i$ can be obtained directly from (B.1)

$$852 \quad \underline{\mathbf{u}}_i = (\mathbf{C}_i - \mathbf{B}_i) / l_2 \quad (\text{B.12})$$

853 Introducing (B.2) into (B.12) and differentiating (B.12) with respect to time leads to:

$$854 \quad \dot{\underline{\mathbf{u}}}_i = (\boldsymbol{\tau}_c - l_1 \underline{\mathbf{v}}_i^\perp \dot{q}_i) / l_2 \quad (\text{B.13})$$

855 Finally, from (B.11), it comes that:

$$856 \quad \dot{\underline{\mathbf{u}}}_i = (\mathbf{I}_3 + l_1 \underline{\mathbf{v}}_i^\perp \mathbf{a}_i / b_i) / l_2 \boldsymbol{\tau}_c = \mathbf{M}_i^T \boldsymbol{\tau}_c \quad (\text{B.14})$$

857 where \mathbf{I}_3 is the (3×3) identity matrix and matrix \mathbf{M}_i^T is called the interaction matrix. These equations are
858 valuable as long as $b_i \neq 0$ ($b_i = 0$ is a Type 1 singularity condition).

859 Note that the equation (B.14) requires the computation of the input joint variables q_i which can be esti-
860 mated through the observation of the leg direction only (without any use of the encoder measurement).

861 It can be proven that the matrix \mathbf{M}_i^T is of rank 2. As a result, a minimum of two independent legs is
862 necessary to control the end-effector pose. An interaction matrix \mathbf{M}^T can then be obtained by stacking the
863 matrices \mathbf{M}_i^T of k legs ($k = 2 \dots 4$). The conditions for the rank deficiency of the interaction matrix have been
864 presented in Section 4.2.2.

865 The previous equations characterize the inverse kinematics of the hidden robot models of the Quattro.

866 It should be mentioned that the equations for the forward kinematics are not given here for reason of paper
867 compactness as they are tedious. However, the *fkp* can be solved using the proposed geometric approach
868 presented in Section 4.2.2.

869 Acknowledgements

870 This work was supported by the French ANR project ARROW (ANR-2011BS3-006-01), by the French
871 ANR project EquipEx RobotEx (ANR-10-EQPX-44), by the European Union Program “Compétitivité Régionale
872 et Emploi 2007–2013” (FEDER – Région Auvergne) and by the EU project Feder RobotEx.

873 The authors would like to thank Michel Coste from the Institut de Recherche Mathématique de Rennes
874 (University of Rennes, France) for his smart advices about the use of the Bézout theorem and the definition of
875 the Bohemian Dome.

876 References

- 877 [1] T. Leinonen, Terminology for the theory of machines and mechanisms, Mechanism and Machine Theory
878 26.
- 879 [2] J. Merlet, Parallel Robots, 2nd Edition, Springer, 2006.
- 880 [3] D. Dementhon, L. Davis, Model-based object pose in 25 lines of codes, International Journal of Com-
881 puter Vision 15 (1995) 123–141.
- 882 [4] M. Dhome, M. Richetin, J. Lapresté, G. Rives, Determination of the attitude of 3-d objects from a
883 single perspective view, IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (12) (1989)
884 1265–1278.
- 885 [5] D. Lowe, Three-dimensional object recognition from single two-dimensional images, Artificial Intelli-
886 gence 31 (1987) 355–394.
- 887 [6] B. Espiau, F. Chaumette, P. Rives, A new approach to visual servoing in robotics, IEEE Transactions on
888 Robotics and Automation 8 (3).

- 889 [7] R. Horaud, F. Dornaika, B. Espiau, Visually guided object grasping, *IEEE Transactions on Robotics and*
890 *Automation* 14 (4) (1998) 525–532.
- 891 [8] P. Martinet, J. Gallice, D. Khadraoui, Vision based control law using 3D visual features, in: *Proceedings*
892 *of the World Automation Congress, WAC96, Robotics and Manufacturing Systems, Vol. 3, Montpellier,*
893 *France, 1996, pp. 497–502.*
- 894 [9] A. Traslosheros, J. Sebastian, L. Angel, F. Roberti, R. Carelliz, Visual servoing of a parallel robot system,
895 in: *IEEE International Symposium on Intelligent Signal Processing, 2007.*
- 896 [10] Z. Qi, J. McInroy, Improved image based visual servoing with parallel robot, *Journal of Intelligent Robot*
897 *Systems* 53 (2008) 359–379.
- 898 [11] R. Garrido, A. Soria, M. Trujano, Visual pid control of a redundant parallel robot, in: *5th International*
899 *Conference on Electrical Engineering, Computing Science and Automatic Control (CCE 2008), 2008.*
- 900 [12] M. Trujano, R. Garrido, A. Soria, Robust visual control of parallel robots under uncertain camera orien-
901 *tation, International Journal of Advanced Robotic Systems* 9.
- 902 [13] N. Andreff, A. Marchadier, P. Martinet, Vision-based control of a Gough-Stewart parallel mechanism us-
903 *ing legs observation, in: Proceedings of the IEEE International Conference on Robotics and Automation,*
904 *ICRA’05, Barcelona, Spain, 2005, pp. 2546–2551.*
- 905 [14] V. Gough, S. Whitehall, Universal tyre test machine, in: *Proceedings of the FISITA 9th International*
906 *Technical Congress, 1962, pp. 117–317.*
- 907 [15] N. Andreff, T. Dallej, P. Martinet, Image-based visual servoing of gough-stewart parallel manipulators
908 *using legs observation, International Journal of Robotics Research* 26 (7) (2007) 677–687.
- 909 [16] E. Özgür, N. Andreff, P. Martinet, Dynamic control of the quattro robot by the leg edgels, in: *Proceedings*
910 *of the IEEE International Conference on Robotics and Automation, ICRA11, Shanghai, China, 2011.*
- 911 [17] N. Andreff, P. Martinet, Vision-based kinematic modelling of some parallel manipulators for control pur-
912 *poses, in: Proceedings of EuCoMeS, the First European Conference on Mechanism Science, Obergurgl,*
913 *Austria, 2006.*
- 914 [18] F. Chaumette, *The Confluence of Vision and Control*, no. 237 in *LNCIS*, Springer-Verlag, 1998, Ch.
915 *Potential problems of stability and convergence in image-based and position-based visual servoing, pp.*
916 *66–78.*
- 917 [19] H. Michel, P. Rives, Singularities in the determination of the situation of a robot effector from the per-
918 *spective view of 3 points, Tech. rep., INRIA (1993).*
- 919 [20] S. Briot, P. Martinet, Minimal representation for the control of Gough-Stewart platforms via leg obser-
920 *vation considering a hidden robot model, in: Proceedings of the 2013 IEEE International Conference on*
921 *Robotics and Automation (ICRA 2013), Karlsruhe, Germany, 2013.*
- 922 [21] S. Caro, W. Khan, D. Pasini, J. Angeles, The rule-based conceptual design of the architecture of serial
923 *schonflies-motion generators, Mechanism and Machine Theory* 45 (2) (2010) 251–260.
- 924 [22] N. Andreff, B. Espiau, R. Horaud, Visual servoing from lines, *International Journal of Robotics Research*
925 *21 (8) (2002) 679–700.*
- 926 [23] J. Plücker, On a new geometry of space, *Philosophical Transactions of the Royal Society of London* 155
927 *(1865) 725–791.*
- 928 [24] F. Chaumette, *La commande des robots manipulateurs*, Hermès, 2002.

- 929 [25] X. Kong, C. Gosselin, *Type Synthesis of Parallel Mechanisms*, Springer, 2007.
- 930 [26] G. Gogu, *Structural Synthesis of Parallel Robots*, Springer, 2008.
- 931 [27] J. Hervé, Lie group of rigid body displacements, a fundamental tool for mechanism design, *Mechanism*
932 *and Machine Theory* 34 (5) (1999) 719–730.
- 933 [28] M. Carricato, V. Parenti-Castelli, Singularity-free fully-isotropic translational parallel manipulators, *International Journal of Robotics Research* 21 (2) (2002) 161–174.
- 934
- 935 [29] X. Kong, C. Gosselin, A class of 3-dof translational parallel manipulators with linear input-output equa-
936 tions, in: *Proceedings of the Workshop on Fundamental Issues and Future Research Directions for Parallel*
937 *Mechanisms and Manipulators*, Québec City, QC, Canada, 2002, pp. 3–4.
- 938 [30] G. Gogu, Structural synthesis of fully-isotropic translational parallel robots via theory of linear transfor-
939 mations, *European Journal of Mechanics. A/Solids* 23 (6) (2004) 1021–1039.
- 940 [31] C. Gosselin, J. Angeles, Singularity analysis of closed-loop kinematic chains, *IEEE Transactions on*
941 *Robotics and Automation* 6 (3) (1990) 281–290.
- 942 [32] D. Zlatanov, I. Bonev, C. Gosselin, Constraint singularities of parallel mechanisms, in: *Proceedings of*
943 *the IEEE International Conference on Robotics and Automation (ICRA 2002)*, 2002.
- 944 [33] I. Bonev, D. Zlatanov, C. Gosselin, Singularity analysis of 3-dof planar parallel mechanisms via screw
945 theory, *ASME Journal of Mechanical Design* 125 (3) (2003) 573–581.
- 946 [34] P. Ben-Horin, M. Shoham, Singularity analysis of a class of parallel robots based on grassmanncayley
947 algebra, *Mechanism and Machine Theory* 41 (8) (2006) 958–970.
- 948 [35] S. Caro, G. Moroz, T. Gayral, D. Chablat, C. Chen, Singularity analysis of a six-dof parallel manipulator
949 using grassmann-cayley algebra and grobner bases, in: *Proceedings of the Symposium on Brain, Body*
950 *and Machine*, Montreal, QC, Canada, 2010.
- 951 [36] S. Briot, I. Bonev, D. Chablat, P. Wenger, V. Arakelian, Self motions of general 3-rpr planar parallel
952 robots, *International Journal of Robotics Research* 27 (7) (2008) 855–866.
- 953 [37] L. Tsai, Kinematics and optimization of a spatial 3-upu parallel manipulator, *ASME Journal of Mechan-*
954 *ical Design* 122 (2000) 439–446.
- 955 [38] R. Clavel, Device for the movement and positioning of an element in space (December 1990).
- 956 [39] V. Nabat, M. de la O Rodriguez, O. Company, S. Krut, F. Pierrot, Par4: very high speed parallel robot for
957 pick-and-place, in: *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots*
958 *and Systems (IROS 2005)*, 2005.
- 959 [40] D. Chablat, P. Wenger, Architecture optimization of a 3-dof parallel mechanism for machining applica-
960 tions, the Orthoglide, *IEEE Transactions on Robotics and Automation* 19 (3) (2003) 403–410.
- 961 [41] I. Bonev, Direct kinematics of zero-torsion parallel mechanisms, in: *Proceedings of the 2008 IEEE*
962 *International Conference on Robotics and Automation (ICRA 2008)*, 2008.
- 963 [42] A. Wolf, M. Shoham, F. Park, Investigation of singularities and self-motions of the 3-UPU robot, in:
964 *Advances in Robot Kinematics*, Dordrecht, Germany, 2002.
- 965 [43] C. Tischler, K. Hunt, A. Samuel, A spatial extension of cardanic movement: its geometry and some
966 derived mechanisms, *Mechanism and Machine Theory* 33 (1998) 1249–1276.
- 967 [44] S. Krut, O. Company, M. Benoit, H. Ota, F. Pierrot, I4: A new parallel mechanism for Scara motions,
968 in: *Proceedings of the 2003 International Conference on Robotics and Automation (ICRA 2003)*, 2003.

- 969 [45] L. Tsai, S. Joshi, Comparison study of architectures of four 3 degree-of-freedom translational parallel
970 manipulators, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA
971 2001), 2001.
- 972 [46] F. Pierrot, M. Uchiyama, P. Dauchez, A. Fournier, A new design of a 6-dof parallel robot, Proceedings
973 of the 23rd International Symposium on Industrial Robots. Journal of Robotics and Mechatronics (1990)
974 308–315.
- 975 [47] M. Honegger, A. Codourey, E. Burdet, Adaptive control of the Hexaglide, a 6 dof parallel manipulator,
976 in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 1997), 1997.
- 977 [48] O. Company, F. Pierrot, Modelling and preliminary design issues of a 3-axis parallel machine-tool,
978 Mechanisms and Machine Theory 37 (2002) 1325–1345.
- 979 [49] M. Tale Masouleh, C. Gosselin, M. Husty, D. Walter, Forward kinematic problem of 5-RPUR parallel
980 mechanisms (3T2R) with identical limb structures, Mechanism and Machine Theory 46 (2011) 945–959.
- 981 [50] E. Bézout, Recherches sur le degré des équations résultantes de l'évanouissement des inconnues, Histoire
982 de l'Académie Royale des Sciences, 1764.
- 983 [51] S. Briot, P. Martinet, V. Rosenzweig, The hidden robot: an efficient concept contributing to the analysis
984 of the controllability of parallel robots in advanced visual servoing techniques, IEEE Transactions on
985 Robotics 31 (6) (2015) 1337–1352.
- 986 [52] S. Briot, I. Bonev, Accuracy analysis of 3T1R fully-parallel robots, Mechanism and Machine Theory
987 45 (5) (2010) 695–706.
- 988 [53] J. Merlet, Jacobian, manipulability, condition number, and accuracy of parallel robots, ASME Transac-
989 tions Journal of Mechanical Design 128 (1) (2006) 199–206.
- 990 [54] A. Pashkevich, D. Chablat, P. Wenger, Stiffness analysis of overconstrained parallel manipulators, Mech-
991 anism and Machine Theory 44 (5) (2009) 966–982.
- 992 [55] N. Binaud, P. Cardou, S. Caro, P. Wenger, The kinematic sensitivity of robotic manipulators to joint clear-
993 ances, in: Proceedings of ASME Design Engineering Technical Conferences, Montreal, QC, Canada,
994 2010.
- 995 [56] S. Briot, V. Arakelian, Optimal force generation of parallel manipulators for passing through the singular
996 positions, International Journal of Robotics Research 27 (8) (2008) 967–983.
- 997 [57] D. Chablat, G. Moroz, P. Wenger, Uniqueness domains and non singular assembly mode changing trajec-
998 tories, in: Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA
999 2011), Shanghai, China, 2011.