Force-feedback control and non-contact sensing: a unified approach

Bernard Espiau, Jean-Pierre Merlet, Claude Samson

INRIA
Centre de Sophia-Antipolis
2004 Route des Lucioles 06560 Valbonne, France

Abstract

We propose in this paper a global approach to the problem of proximity and force-based control in robotics applications. A general model of interactions between a sensor and its environment allows to define the concept of interaction screw. Then, a general approach to the control of redundant systems is applied to the generalization of the classical hybrid control scheme. An analysis of the robustness of this scheme shows the importance of knowing the null space of a jacobian operator associated with the problem. After shortly recalling how the scheme applies to the case of non-contact sensors, the problem of force control is treated through an example. The similitude of the two problems reflects in the fact that either virtual or real forces and torques can be considered. Calculation of a hybrid control requires determining a compliance frame and the directions which are either force or position controlled. In our approach the compliance frame and its partition are automatically determined from the measurements by estimating the null space aforementioned. Some applications of this improved control to surface following tasks are presented.

1 Introduction

We are interested in this paper in the design of sensor-based control loops with the point of view of Automatic Control. We consider the class of high data rate sensors like force, proximity, local range, and local visual sensing. The robotics literature in this domain is mostly centered on the concept of 'hybrid control', issued from studies in force control (see for example [3, 4, 6, 7, 12]). The use of proximity sensors in a control loop has been less investigated, although work has also been done in this area ([11], for example).The use of a vision system inside a control loop rises also as a new domain of interest ([2, 8]).

Our objective here is to show how most sensor-based applications, despite certain specific aspects, may be treated within a unified control framework, and to point out the importance of the knowledge of the null space of a certain Jacobian operator associated with the problem. The paper is organized as follows. Firstly basic concepts about task functions and stability are recalled; then, the main features of the approach are briefly presented and applied to sensor-based tasks. The specific use of force sensors is studied through an example and some experimental results are given. For more details on the approach, only briefly presented here, the reader is referred to [9] and [10].

2 Background

2.1 The Concept of Task Function

Let us consider a rigid robot, the dynamic equation of which is

$$\Gamma = M(q)\ddot{q} + N(q, \dot{q}, t) \;\; with \;\; dim\ (q) = \;\; dim(\ \Gamma\ ) = n \tag{1}$$

Equation 1 is a *state* equation of the system (with natural state vector $(q, \dot{q})$), which fully describes its dynamics. The user's objective, i.e. the task to be performed, may be characterized by an associated *output* function, or, more precisely, may be expressed as the problem of regulating some C^2 n-dimensional function $e(q, t)$, called **task-function**, to zero during the given time interval $[0, T]$ of the task. The simplest example of task-function is $e(q, t) = q - q_d(t)$, where $q_d(t)$ is a desired trajectory specified in the joint space of the robot.. As explained in [9], where other situations are discussed, the regulation of the task-function $e(q, t)$ is a well-conditioned problem only if this function possesses some properties. One of them is the existence and uniqueness of a C^2 *ideal trajectory* $q_r(t)$ such that $e(q_r(t), t) = 0$, with $t \in [0, T]$, and $q_r(0) = q_0$ where q_0 is a given initial condition. Another one, very important, is the regularity of the *task-Jacobian matrix* $\frac{\partial e}{\partial q}$ in the vicinity of $q_r(t)$. When all the required conditions are satisfied, efficient and robust control laws may then be derived.

2.2 The Central Stability Condition

Given equation 1, and a task function $e(q, t)$, a control scheme has to be designed. A general control expression is obtained by extending the exact decoupling/linearization approach to the case where 'approximations' are allowed. This leads to ([9]):

$$\Gamma = -k\hat{M}(\hat{\frac{\partial e}{\partial q}})^{-1}G(\mu De + \hat{\frac{\partial e}{\partial q}}\dot{q} + \hat{\frac{\partial e}{\partial t}}) + \hat{N} - \hat{M}(\hat{\frac{\partial e}{\partial q}})^{-1}\hat{f} \tag{2}$$

where f is a term appearing in the second differentiation of e ($\ddot{e} = \frac{\partial e}{\partial q}\dot{q} + f$), G and D are positive matrices, and μ and k are positive scalar control gains. Except for μ, D and G, all terms are allowed to be nonlinear functions. The 'hatted' terms are allowed to be approximations of the corresponding functions. Sufficient conditions for the closed-loop system to be stable are given and discussed in [9]. A particularly important condition of matrix positivity is :

$$\frac{\partial e}{\partial q}\ (\hat{\frac{\partial e}{\partial q}})^{-1}\ >\ 0 \tag{3}$$

which indicates how the term $\hat{\frac{\partial e}{\partial q}}$ should be chosen with regard to the 'true' task Jacobian matrix $\frac{\partial e}{\partial q}$.

3 Use of Exteroceptive Sensors

3.1 Modeling of Interactions

A sensor (**S**) is linked to a rigid body (**B**) with related frame F_S. The environment with which (**S**) interacts is assumed to be an object (**T**) associated with a frame F_T. The output of (**S**) is a $\mathbf{p}-$dimensional vector sensor signal denoted $\bar{s}$. A fixed reference frame F_0 is also given. The position of (**B**) is an element $\bar{r}$ of the Lie group of the displacements, SE_3. This is a 6-dimensional differential manifold. Its tangent space is se_3 and the dual se_3^*. A screw, H, is an element of se_3, characterized by its vector u and the value $H(O)$ of its field in some point O. We will write: $H = (\ H(O), u\ )$. When expressed in a given frame, H is represented by a vector in $\mathbf{R^6}$. The velocity screw of a frame F_k with respect to a frame F_l is denoted as T_{kl}.

Let us consider one component s_j of $\bar{s}$. Our basic assumption is that s_j is a C^2 function of the relative position $\bar{r}$ of (**S**) with respect to (**T**) only:

$$s_j : \bar{r} \in SE_3 \longmapsto s_j(\bar{r}) \in R \tag{4}$$

Now, we suppose that (**S**) belongs to a robot, so that the joint coordinates $\mathbf{q}$ constitute a local chart of SE_3. As for the motion of (**T**) with respect to F_0 it can be parametrized by the independent time variable t. Then, we may write:

$$s_j = s_j(q, t) \tag{5}$$

Since the differential of s_j is a mapping from $\mathbf{se_3^*}$ to $\mathbf{R}$, we also have:

$$\dot{s}_j = \frac{\partial s_j}{\partial \bar{r}} \bullet \frac{d\bar{r}}{dt} = H_j \bullet T_{ST} \tag{6}$$

where $\bullet$ denotes the screw product operation. Let us now suppose that the sensor (**S**) is positioned on the last body, ($\mathbf{B_6}$), of a six-jointed robot. Then, in F_0 and according to 6:

$$\dot{s}_j = [u_j^T\ H_j^T(O_6)](q,t) J_6(q)\ \dot{q}\ -\ [u_j^T\ H_j^T(T)](q,t) \begin{bmatrix} V_T(t) \\ \omega_{F_T}(t) \end{bmatrix} \tag{7}$$

where J_6 is the Jacobian matrix associated with the frame F_6 linked to ($\mathbf{B_6}$), with origin O_6 , and T is a fixed point of (**T**).

The equation 6 (or 7) shows that all information about the variations of the sensor/environment interactions is carried by the *interaction screw* H_j . Unfortunately, our knowledge of H_j is only usually partial. As a consequence, only *approximate models* of H_j are used in practice.

Consider now the complete sensor (**S**) with output vector $\bar{s}$.We will say that a set of p compatible constraints: $\bar{s}(\bar{r}) - \bar{s}_d = 0$ determines a **virtual linkage** between (**B**) and (**T**). When a frame and a basis are chosen, the virtual linkage is characterized by the 6 $\times$ p *interaction matrix*:

$$L = \begin{bmatrix} u_1 & \dots & u_p \\ H_1(P) & \dots & H_p(P) \end{bmatrix} \tag{8}$$

evaluated at some point P.The concept of virtual linkage determines the motions which may be sensor-controlled and the ones which remain 'free'. When using a force sensor, this virtual linkage corresponds to a physical linkage.

4 Sensor-based Tasks

From now on, we suppose $n = 6$, consider $\mathbf{SE_3}$ as our working space, and express all screw in a frame linked to the sensory system. In many applications the regulation of sensor outputs has to be combined with other objectives such as, for example, trajectory tracking. Such tasks are often referred to as *hybrid tasks*. Usually, a partial specification of the global task leads to the derivation of a sensor-based task vector $e_1(q,t)$ made of $m \leq n$ independent components. It appears that, generally, whenever the regulation of sensor signals constitutes the main task which is to be complemented by a secondary objective, we are in fact dealing with a problem of *redundancy*. In the next subsection, we recall and summarize some general results about redundancy, taken from [9].

4.1 Some Results About Redundant Tasks

It is known that a way of treating the problem of redundancy consists of minimizing a secondary cost-function under the constraint of achieving a primary task constraint (see [10] for example). Thus, we denote as $J_1 = \frac{\partial e_1}{\partial \bar{r}}$ the $(m \times n)$ Jacobian matrix associated with e_1, assumed to be full-rank, and as h_s a secondary cost function to be minimized under the constraint $e_1 = 0$, with gradient $g_s = \frac{\partial h_s}{\partial \bar{r}}$.

In order to be able to minimize h_s under $e_1 = 0$ we have to determine the subspace of motions which are left free by this constraint. This means that one has to know the *Null Space* $N(J_1)$ of J_1 or equivalently the *Range Space* $R(J_1^T)$ along the ideal trajectory, $q_r(t)$ of the robot. In practice, this knowledge is equivalent to the knowledge of a $m \times n$ matrix-valued function W such that: (property P_1):

$$R(W^T) = R(J_1^T) \quad (\text{or } N(W) = N(J_1)\ ) \qquad along q_r(t) \tag{9}$$

or equivalently such that:

$$\begin{cases} rank(W) = m \\ (I - W^\dagger W) J_1^T = 0 \quad along\ q_r(t) \end{cases} \tag{10}$$

Once such a matrix is known, a possible global task function that can be associated with the problem is:

$$e = W^\dagger e_1 + \alpha (I - W^\dagger W) g_s \tag{11}$$

where α is a positive scalar function. If, in addition to the property P_1, the chosen matrix W also satisfies the property P_2:

$$J_1 W^T > 0 \quad along\ q_r(t)$$

then (see [9] for more details), it is possible to show that the global task-Jacobian matrix $\frac{\partial e}{\partial \bar{r}}$ is such that:

$$\frac{\partial e}{\partial r}(I_n + \gamma\ (I_n - W^\dagger W)\ ) > 0 \quad along\ q_r(t) \tag{12}$$

for any positive scalar function γ equal to, or larger than some positive function γ_m which depends on α. If α is small 'enough', then $\gamma_m = 0$ and $\frac{\partial e}{\partial \bar{r}}$ is positive.

The reason for trying to satisfy the property 12 is that, in this case, it is not necessary to know the Jacobian $\frac{\partial e}{\partial \bar{r}}$ explicitly in order to derive a stable control. Indeed, the important control stability condition 3, $\frac{\partial e}{\partial q}\ (\hat{\frac{\partial e}{\partial q}})^{-1} > 0$ (with $\frac{\partial e}{\partial q} = \frac{\partial e}{\partial \bar{r}} \frac{\partial \bar{r}}{\partial q}$) is then satisfied by choosing:

$$(\hat{\frac{\partial e}{\partial q}})^{-1} = (\frac{\partial \bar{r}}{\partial q})^{-1}\ (I_n + \gamma\ (I_n - W^\dagger W)\ ) \tag{13}$$

For small values of α, one can also take $\hat{\frac{\partial e}{\partial q}} = \frac{\partial \bar{r}}{\partial q} = J_6$. Therefore, the property P_2 is particularly important when the exact computation of $\frac{\partial e}{\partial q}$ is not possible (the case of most sensor-based tasks) or when it is wished to simplify the control expression without creating unstability. This property also explains why some hybrid control schemes may still work when no variational model of the sensor outputs is used in the control expression.

A particular case, often encountered in practice, is when the subspace $N(J_1(q,t))$ is invariant on the set of solutions of $e_1(q,t) = 0$.Then it is possible to choose W as a *constant* matrix.

In summary, when W is chosen so as to satisfy P_1, the regulation of $e(q,t)$ to zero yields the (constrained) minimization of h_s. By further restricting the choice of W so as to satisfy P_2, it is possible to provide the global task Jacobian matrix with a positivity property which can in turn be exploited at the control level. Obviously, when J_1 is known, both properties P_1 and P_2 are automatically satisfied by choosing $W = J_1$. However, the satisfaction of P_1 only requires the knowledge of $R(J_1^T)$ along $q_r(t)$. The invariance of this subspace on the manifold defined by the primary constraint $e_1 = 0$ indicates that W can be calculated independently of the secondary objective. Furthermore, even when P_1 is slightly transgressed in practice, the regulation of the task function 11 may still constitute a well-conditioned control problem, although the secondary cost function is no longer usually minimized.

The application of these general results to the specific case of sensor-based tasks is discussed in the next section.

4.2 Application to Sensor-based Tasks

In order to simplify the notation, it will be assumed that all screws and related matrices are evaluated at the same point Q and expressed in the basis of a single frame F .

Recall that $L(\bar{r},t)$ $(= (\frac{\partial \bar{s}}{\partial \bar{r}})^T)$ is the $6 \times p$ interaction matrix associated with the problem. Its rank m ($m \leq \min\,(p,6)$) is assumed to be constant and known along the ideal trajectory of the robot. We will also restrict the discussion to the following set of sensor-based task functions:

$$e_1(q,t) = D(t)\bar{s}(\bar{r},t) - \sigma(t) = D(t)\tilde{s} \tag{14}$$

where

* $D(t)$ is a $(m \times p)$ full rank combination matrix, defined by the user and such that the $(m \times 6)$ matrix DL^T is also of full rank m along the trajectory of the robot.

* $\sigma(t)$ is a m-dimensional reference vector chosen by the user.

* $\tilde{s} = \bar{s} - D^\dagger \sigma$

When D is a square matrix (i.e. when $p = m$), the task associated with e_1 is equivalent to having each sensor output s_j regulated around an ideal value $s_{r,j}$.

The Jacobian matrix associated with the sensor-based task is:

$$J_1 = \frac{\partial e_1}{\partial \bar{r}} = DL^T \tag{15}$$

From 15, and since DL^T is of rank m, we have:

$$R(J_1^T) = R(L) \tag{16}$$

This last relation reminds us that the knowledge of the vector subspace $R(J_1^T)$ is equivalent to the knowledge of the screw subspace $\mathcal{S}$ spanned by the interaction screws.

According to 11 and 14, one may then consider the following global task function:

$$e = W^\dagger D\tilde{s} + \alpha(I_6 - W^\dagger W)g_s \tag{17}$$

where W should ideally be chosen so as to satisfy P_1 along $q_r(t)$. According to 16, this is equivalent to having:

$$R(W^T) = R(L) \tag{18}$$

Because of the dimension of W, this property is also equivalent to having $W^T = (w_1.....w_m)$, where the set $\{w_j\}$, $j = 1...m$ is any basis of the interaction screw subspace $\mathcal{S}$. Because of 15, the property P_2 becomes in this case:

$$DL^T W^T > 0 \tag{19}$$

Then, if the choice of W is $W = DL^T$ or if D is determined from the prior knowledge of a basis of $\mathcal{S}$ according to $D = WL$, properties P_1 and P_2 are both satisfied.

In practice, the 'true' interaction matrix L usually has to be replaced by some approximation $\hat{L}(t)$ evaluated along the trajectory actually followed by the robot. In some cases this estimated matrix may be is determined off-line, or derived from a simplified model of the virtual linkage that seems adequate [9]. In other cases, the off-line estimation may be complemented or replaced by an on-line estimation scheme, as explained further.

Then, a possible choice for D is:

$$D = W\hat{L} \tag{20}$$

In this case the property P_1 is satisfied if $R(\hat{L}) = R(L)$, while the property P_2 is satisfied if $W\hat{L}L^T W^T > 0$. The analogy with force based control loops comes from that, by using 20 in 17 the global task function may also be written:

$$e = S\tilde{F} + (I \ - \ S)\alpha g_s \tag{21}$$

where $S = W^\dagger W$ is the orthogonal projection operator on the subspace of constrained motions, and where $\tilde{F} = \hat{L}\tilde{s}$ can be interpreted as a *virtual* force of contact in the case of non-contact sensors. Then

$$\frac{\partial \tilde{F}}{\partial \bar{r}} = \hat{L}L^T \tag{22}$$

and the matrix $\hat{L}L^T$ can be interpreted as the stiffness matrix of a "virtual generalized spring" composed of the sensing system and the target.

5 Application to Force Based Control Loops. An example

We consider the problem of moving the robot's end-effector along the edge of a planar surface while applying a constant contact force. Let :

- F denote the screw measured by the force sensor. It is a vector in R^6 when expressed in a frame linked to the end-effector, the origin of which is chosen as the point of contact.
- n be the unit vector normal to the edge
- f_d denote the desired intensity of the contact force.

Then a possible task-function to be associated with the force control problem is :

$$e_1 = [n^T \ 0]F - f_d \tag{23}$$

The corresponding Jacobian matrix is:

$$J_1 = [n^T \ 0] \frac{\partial F}{\partial \bar{r}} \tag{24}$$

with a stiffness matrix of the form :

$$\frac{\partial F}{\partial \bar{r}} = k \begin{bmatrix} nn^T & 0 \\ 0 & 0 \end{bmatrix} \quad , k > 0 \tag{25}$$

Therefore:

$$J_1 = [kn^T \ \ 0] \tag{26}$$

and a matrix W which obviously satisfies the properties P_1 and P_2 is :

$$W = [n^T \ 0] \tag{27}$$

As for the secondary objective, it may consist of minimizing a cost-function h_s such as the square of the distance between the cartesian position of the end-effector and some desired position parametrized by the variable t. Examples of functions h_s are given in [9].

By using 23 and 27 in 17, we then obtain the following global task function:

$$e = SF - \begin{bmatrix} n \\ 0 \end{bmatrix} f_d + \alpha(I - S)g_s \tag{28}$$

with the selection matrix :

$$S = W^\dagger W = \begin{bmatrix} n^T n & 0 \\ 0 & 0 \end{bmatrix} \tag{29}$$

This task function may also be written in the familiar form :

$$e = S(F - F_d) + \alpha(I - S)g_s \tag{30}$$

where

$$F_d = \begin{bmatrix} n \\ 0 \end{bmatrix} f_d \tag{31}$$

has the meaning of a desired screw.

Obviously, depending on the relative positionning of the chosen frame linked to the end-effector with respect to the normal of the surface, the selection matrix S may, or may not be, diagonal.

The important fact, pointed out by relations 28 and 29, is the necessity of knowing the vector n normal to the surface in order to be able to calculate the value of the task function at any time. One can think of several methods to estimate n:

- use of the force measurements. If the contact is frictionless the force measurement may be used to determine the normal n (see [4]). If there is friction, the friction force has to be estimated and substracted from the measured force.
- use of a prior knowledge of the normal to the surface in a fixed frame F_0. Then, n can be recovered from the computation of the end-effector orientation with respect to F_0.
- use of the motion of the end-effector when it is in contact with the surface, knowing that the velocity of the end-effector is normal to n.

These methods have been experimentally tested with a Scara robot and an AICO 6 componant force sensor with 200 Hz sampling frequency. The robot controller was a multi-processor (68020 and 68881 CPU) AICO controller.

In a first experiment the normal n is initially estimated from the force measurement. Then, during the motion of the robot, and as long as the points are approximately aligned, the normal is calculated from the estimation of a vector tangent to the contour. The trajectory followed by the end-effector is shown in figure 1 and the evolution of the measured force intensity is represented in figure 2. The desired force value was 2.5N and the velocity of the manipulator was 8mm/s. The average measured intensity is 2.61N and the standard deviation equal to 0.658 N^2. This method has been also

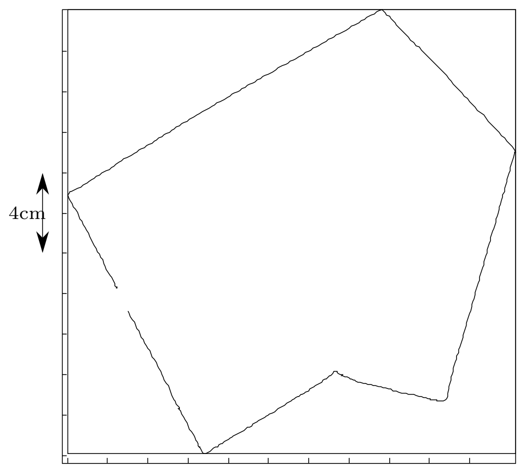

Figure 1: The followed trajectory with the controller using an estimation of the normal through the force measurements

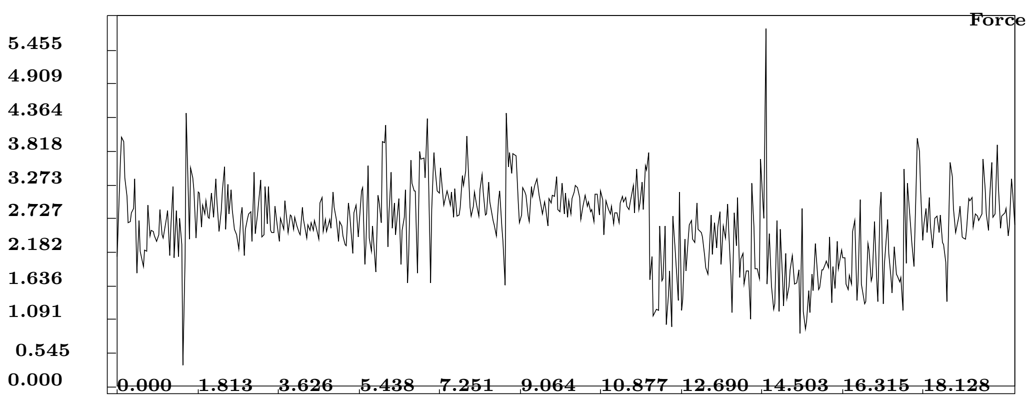

Figure 2: The force measurements with the controller using an estimation of the normal through the force measurements

successfully tested to perform fine insertion with clearance less than 10 μm, no chamfer and an horizontal insertion axis, and to turn a crank [5].

In the second experiment, an a priori model of the contour, obtained after implementing a geometrical method called *probing* is used. This "probing" method [1] is a technique which applies to polygonal contours, and it roughly consists of the following: according to some predetermined optimal strategy the manipulator comes into contact with the contour a finite (and minimal) number of time, so as to determine the location of the vertices. The contact between the manipulator and the contour is detected by the force sensor. The vertices are then connected together so as to obtain a geometrical model of the contour. During the force control experiment this model is in turn used to determine the normal to the contour at any location of the end-effector. The end-effector's velocity is the same as before, and the measured force intensity is shown in figure 3. Its average value is 2.506N with a standard deviation of 0.355 N^2. During,

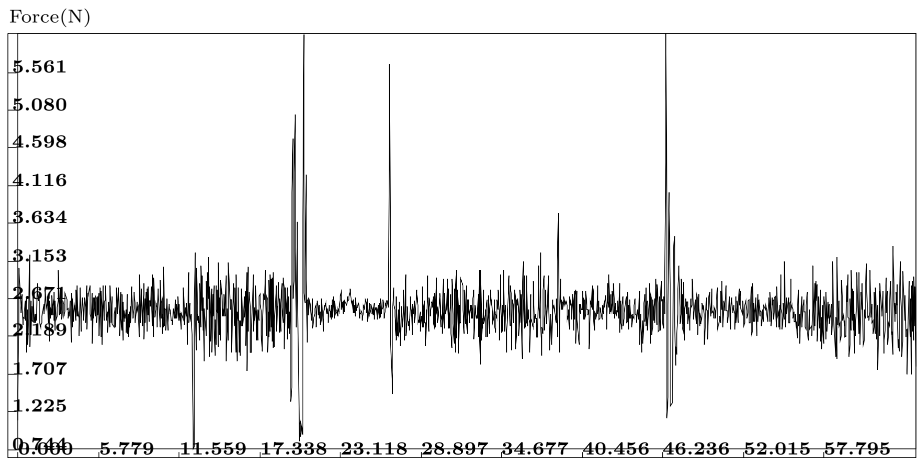

Figure 3: The force measurements with the controller using a reference model of the contour

both experiments, the main difficulty was the passage of the contour edges where the normal to the contour is not uniquely determined and where there is a risk of loosing the contact. The solution of this problem required the addition of specific strategies to the control algorithm. Obviously, any prior knowledge of the contour's geometry is also useful at this point.

6 Conclusion

The unified approach proposed in this paper enables to deal with various sensors (contact or non-contact sensors) using the same basic algorithms. The task function approach enables to design robust control scheme as soon as the null space of the jacobian operator associated to the task is determined. We have shown that, for the case of force sensor, it is possible to calculate this null space, using various methods. Experiments (which are currently under way) for the surface following problem show that this approach is really effective.

References

[1] J.D. Boissonnat, M. Yvinec: **Probing a scene of non convex polyhedra**, INRIA Research Report n°1110, October 1989

[2] J.T. Feddema, C.S.G. Lee, O. Mitchell: **Automatic selection of image features for visual servoing of a robot manipulator**, IEEE Conference on Robotics and Automation, Scottsdale, USA, May 1989

[3] M.T.Mason: **Compliant motion**, in 'Robot Motion, Planning and Control', Brady and al., 1st edition, Cambridge, the MIT Press, 1982

[4] J.P. Merlet: **C-surface theory applied to the design of an hybrid force-position controller**, IEEE Conference on Robotics and Automation, Raleigh, USA, April 1987

[5] J.P. Merlet: **Manipulateurs parallèles, 3eme partie : Applications**, INRIA Research Report n°1003, March 1989

[6] R. P. Paul: **Problems and research issues associated with the hybrid control of force and displacement**, IEEE Conference on Robotics and Automation, Raleigh, USA, April 1987

[7] C. Reboulet, A. Robert: **Hybrid control of a manipulator with an active compliant wrist**, Third International Symposium on Robotics Research, Gouvieux, France, October 1985

[8] P. Rives, B. Espiau, F. Chaumette: **Visual servoing based on a task function approach**, 1st International Symposium on Experimental Robotics, Montreal ,Canada, June 1989

[9] C. Samson, M. Le Borgne, B. Espiau: **Robot control: the task-function approach**, Oxford University Press, to be published

[10] C. Samson, M. Le Borgne, B. Espiau: **Robot redundancy: an automatic control approach**, NATO Advanced Research Workshop on 'Robots with Redundancy: Design, Sensing and Control', Salo, Italy, June 1988

[11] C. Wampler: **Multiprocessor control of a telemanipulator with optical proximity sensors**, International Journal of Robotics Research, Vol 3 No 1, Spring 1984

[12] D.E. Whitney: **Historical perspective and state of the art in robot force control**, IEEE Conference on Robotics and Automation, St Louis, USA, March 1985