



Commande de systèmes redondants et évitement d'obstacles

Bernard Espiau

► To cite this version:

Bernard Espiau. Commande de systèmes redondants et évitement d'obstacles. [Rapport de recherche] RR-0495, INRIA. 1986. inria-00076059

HAL Id: inria-00076059

<https://inria.hal.science/inria-00076059>

Submitted on 24 May 2006

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.



CENTRE DE RENNES

IRISA

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél. (3) 954 90 20

Rapports de Recherche

N° 495

**COMMANDE DE SYSTÈMES
REDONDANTS ET ÉVITEMENT
D'OBSTACLES**

Bernard ESPIAU

Mars 1986

Campus Universitaire de Beaulieu
Avenue du Général Leclerc
35042 - RENNES CÉDEX
FRANCE
Tél. : (99) 36.20.00
Télex : UNIRISA 95 0473 F

Publication Interne n° 280

52 pages
Janvier 1986

COMMANDE DE SYSTEMES REDONDANTS ET EVITEMENT D'OBSTACLES

Bernard ESPIAU

RESUME

Ce rapport effectue une synthèse des diverses approches possibles concernant la commande des robots redondants rigides. Une première partie est consacrée à une présentation générale du problème incluant un tour d'horizon de la littérature. La deuxième partie propose, dans un cadre dynamique, une approche globale de la redondance, par augmentation de la dimension des variables de la tâche, et l'on montre que ceci est équivalent à effectuer un certain choix d'inverse généralisée. La dernière partie présente une application à l'évitement d'obstacles par utilisation de capteurs de proximité, et donne plusieurs résultats de simulation.

REDUNDANCY CONTROL AND OBSTACLE AVOIDANCE

Bernard ESPIAU

ABSTRACT

This research report realizes a synthesis of the possible approaches for controlling rigid redundant robots. The first part is devoted to a general presentation of the problem, including an overview of the works reported through the literature. The second part proposes, from a dynamic viewpoint, a global method for redundancy resolution, by increasing the dimension of the taskspace. This is shown to be equivalent to a particular choice of a generalized inverse. The last one presents an application to an obstacle avoidance task, by using optical proximity sensors, and gives some simulation results.

COMMANDE DE SYSTEMES REDONDANTS ET EVITEMENT D'OBSTACLES

Bernard ESPIAU

IRISA/INRIA
Campus de Beaulieu
35042 RENNES CEDEX
FRANCE

INTRODUCTION

Ce rapport se propose d'effectuer une synthèse des diverses approches possibles concernant la commande des systèmes redondants, et présente une application à l'évitement d'obstacles avec capteurs de proximité. Une première partie est consacrée à une présentation générale du problème, en se plaçant d'emblée dans un cadre dynamique complet, contrairement à l'usage répandu dans la littérature. La deuxième partie propose une approche globale de la redondance par augmentation de la dimension des variables de tâche. La troisième partie est consacrée à la prise en compte de capteurs dans le cas de l'évitement d'obstacles

1. POSITION DU PROBLEME

1.1. Redondance et régularité

Nous considérons ici un robot rigide à n degrés de liberté q_i . Soit un espace vectoriel définissant les tâches, de dimension m (nous verrons plus loin que m peut être supérieur à 6). Le Jacobien J est la matrice d'une application linéaire de $\mathbb{R}^n$ dans $\mathbb{R}^m$, de rang ρ (dimension de l'image de J). Les propriétés de l'application J caractérisent les classes de problèmes, selon le tableau ci-dessous:

L'APPLICATION J EST, A CHAQUE INSTANT :	INJECTIVE =) J est de rang plein	NON INJECTIVE
SURJECTIVE	① La tâche instantanée est réalisable Image (J) = $\mathbb{R}^m$ $\rho = m = n$ système régulier	② $\rho = n(n)$ système redondant de rang plein
NON SURJECTIVE	③ La tâche instantanée n'est pas réalisable $\rho < m$ $\rho = n(m)$ système surdéterminé	④ rang non plein système singulier

TABEAU 1

J étant une fonction de q , à m et n donnés, il existe un risque de sauter de n'importe quelle situation 1,2 ou 3 à la situation 4 : c'est le passage par une *singularité*. Ces singularités peuvent d'ailleurs être intrinsèques (par exemple celles provenant de la géométrie du manipulateur vis à vis de la tâche à réaliser), ou parfois externes (cas d'une représentation des rotations par angles d'Euler). Seuls les systèmes de classe 2 permettent parfois d'éviter cette perturbation du fonctionnement, par exploitation de la redondance comme nous le verrons par la suite.

D'un point de vue de l'utilisateur, les tâches à effectuer peuvent être classées en trois catégories:

- 1- Les tâches *prioritaires*, auxquelles on impose, lorsque cela est possible, d'être totalement réalisables, dans le sens suivant: soit $x_r(t)$ la trajectoire temporelle désirée d'un vecteur représentant cette tâche; soit

$$x = f_x(q) \quad (1)$$

la relation entre les variables articulaires et les variables réelles correspondant à la tâche. On cherchera alors à maintenir petit et, si possible suivant les types de commande, à faire tendre vers zéro : $\varepsilon_x = x_r - x$, par exemple avec une dynamique donnée sur chaque composante. Nous noterons par la suite J_x le Jacobien associé à cette tâche.

- 2- Les tâches *non prioritaires*, sur lesquelles on pourra accepter une erreur non nulle, c'est à dire en pratique une solution de compromis. Elles sont notées $y_r(t)$ et correspondent à l'évolution souhaitée des variables réelles :

$$y = f_y(q) \quad (2)$$

- 3- Les tâches *supplémentaires*, c'est à dire que l'on ne peut réaliser qu'en utilisant explicitement la redondance. Nous les caractériserons par la minimisation d'une fonction scalaire positive p , en général fonction de q , mais parfois aussi selon les approches d'autres variables telles $\dot{q}$ ou $\ddot{q}$.

Souvent, les tâches de type 2 et 3 peuvent être regroupées en une seule, la mesure du compromis sur les tâches 2 pouvant également s'exprimer à l'aide d'une fonction à minimiser. Soit donc:

$$c = \varepsilon_y^T A \varepsilon_y + \lambda^2 p \quad (3)$$

la fonction de coût regroupant l'ensemble des tâches non prioritaires, à minimiser à chaque instant ou sur un horizon donné. A est une matrice symétrique définie positive, et ε_y est l'erreur $y_r - g(q)$.

Hormis les techniques de détermination *hors ligne* optimales ou évitant des obstacles modélisés, que nous ne considérerons pas ici, toutes les approches existantes ressortent de ce schéma. Cependant, le point de vue de la *commande* n'est que peu fréquemment abordé: la plupart des auteurs considèrent en effet le problème comme réglé dès qu'ils ont trouvé une solution inverse au système $\dot{x} = J_x \dot{q}$, $\dim J_x = m, n$. Or, il est bien évident que la prise en compte des redondances doit s'inscrire dans un schéma complet de commande. Dans ce rapport, nous chercherons donc en particulier à analyser de ce point de vue les approches possibles.

1.2. Introduction aux schémas de commande avec inverses généralisées.

1.2.1. Cas général.

Considérons tout d'abord un système de classe 1. Soit l'équation dynamique en espace articulaire :

$$\Gamma = M(q) \ddot{q} + N(q, \dot{q}) \quad (4)$$

ou M , $\dim(n, n)$ est la matrice d'inertie, et N , $\dim(n, 1)$, regroupe l'ensemble des termes de gravité, de frottements, de forces centrifuge et de Coriolis. Formellement, on peut rassembler la plupart des commandes dans l'expression:

$$\Gamma = \hat{M}(q) u_q + \hat{N}(q, \dot{q}) \quad (5)$$

Γ étant le vecteur des couples articulaires de commande, $\hat{M}$ et $\hat{N}$ les modèles choisis par l'utilisateur pour M et N , et u_q est l'accélération de commande. Au sens de l'automaticien, une commande consiste à fixer une dynamique, généralement linéaire, sur une variable ou une grandeur d'erreur. Lorsque le système est de classe 1, c'est à dire carré et régulier, les deux types classiques de commandes consistent à fixer la dynamique désirée, soit dans l'espace des q , soit dans celui de la tâche.

1.2.1.1. Dynamique linéaire en q

Dans ce cas, nous avons la forme générale:

$$u_q = -\zeta_p(q - q_r) - K_v(\dot{q} - \dot{q}_r) + \ddot{q}_r \quad (6)$$

ou $\{q_r(t), \dot{q}_r(t), \ddot{q}_r(t)\}$ est la trajectoire de référence en q à suivre. Classiquement, $\dot{q}_r = 0$, et $q_r = q_c$, $\dot{q}_r = \dot{q}_c$. Si la tâche est paramétrée par (1), on a alors généralement:

$$\begin{cases} q_c = f^{-1}(x_r) \\ \dot{q}_c = J_x^{-1}(q) \dot{x}_r \end{cases} \quad (7)$$

Si le modèle est parfait, l'évolution est alors linéaire en q :

$$\ddot{q} = u_q \quad (9)$$

Lorsque seule une consigne, x_c , est imposée, une commande appelée "cinématique" est parfois utilisée:

$$\dot{q}_c = J_x^{-1}(x - x_c) \quad (8)$$

1.2.1.2. Dynamique linéaire dans l'espace de la tâche

Ce sont les schémas de la forme:

$$u_q = J_x^{-1}(q) (u_x - \dot{J}_x \dot{q}) ; \dot{J}_x = \frac{dJ_x}{dt} \quad (10)$$

qui conduisent à l'évolution:

$$\ddot{x} = u_x \quad (11)$$

ou u_x a par exemple la forme :

$$u_x = -K_p \varepsilon - K_v \dot{\varepsilon} + \ddot{x}_r \quad (12)$$

ε étant l'écart entre la trajectoire réelle et la trajectoire de référence. Par exemple, si, à nouveau, la trajectoire est représentée en x , alors :

$$\varepsilon = x - x_r$$

REMARQUE : Lorsque la tâche comporte position et orientation d'un repère, il faut paramétrer les rotations pour obtenir la dynamique linéaire ci-dessus. Un choix non paramétrique ([11],[28]) présente cependant l'avantage de ne pas introduire de singularités artificielles, mais la dynamique n'est alors généralement que localement linéaire .

1.2.1.3. Commande en espace opérationnel

Nous la mentionnons pour mémoire, car, dans le cas régulier et avec une modélisation exacte elle se ramène au schéma général.

Considérons tout d'abord le cas où les seules forces extérieures exercées F_{ext} sont exprimées dans l'espace de la tâche; alors, le modèle dynamique "opérationnel" ([20]) s'écrit:

$$F = M' u_x + N' + F_{ext} \quad (13)$$

où:

$$M' = (J_x M^{-1} J_x^T)^{-1} \quad (14)$$

(matrice d'inertie opérationnelle),

et :

$$N' = J_x^{-T} N - M' J_x \dot{q} \quad (15)$$

Sachant $\Gamma = J_x^T F$, il est immédiat de voir qu'en l'absence de forces extérieures ce schéma est équivalent à celui du paragraphe 1.2.1.1. Lorsque F_{ext} est non nul, l'évolution est alors:

$$\ddot{x} = u_x + (M')^{-1} F_{ext} \quad (16)$$

La tâche spécifiée en x devient ainsi non prioritaire, puisque perturbée par F_{ext} . Considérer F_{ext} comme une force d'attraction vers le but dérivant d'un potentiel élastique est alors purement artificiel, car, pour obtenir la stabilité asymptotique, il faut rajouter un terme d'amortissement en $\dot{x}$, par exemple dans u_x . La trajectoire obtenue en x n'est cependant ni découplée ni linéaire du fait de M' . Par ailleurs, la tâche peut également être perturbée par d'autres termes F_k issus d'autres espaces, par exemple ceux des variables "opérationnelles" associées à chaque corps du robot, sur lesquels s'exercent les F_k . En y associant les jacobiens J_k , l'évolution en x devient :

$$\ddot{x} = u_x + (M')^{-1} F_{ext} + J_x M^{-1} \sum_k J_k^T F_k \quad (17)$$

Toutes ces approches sont compatibles en choisissant des jeux d'écritures convenables, par exemple pour u_x . Le schéma général (5) et (10) se prêtant mieux à l'analyse du point de vue des trajectoires, nous préférons nous y tenir dans la suite.

Les propriétés de toutes ces commandes ont été étudiées en [32], en particulier du point de vue de leur robustesse vis à vis des approximations effectuées sur les modèles. Pour focaliser l'exposé sur les problèmes spécifiques de la redondance, nous supposons par la suite les commandes suffisamment robustes, pour éviter d'avoir à prendre en compte explicitement le caractère approché des modèles. Ceci revient donc pour nous à les considérer comme exacts dans la suite.

1.2.2. Utilisation intuitive des inverses généralisées

Considérons à présent les systèmes non réguliers. Alors, une extension "naturelle" des commandes précédentes consiste à remplacer où cela est nécessaire J_x^{-1} par une inverse généralisée J_x^g (*).

Cette approche se justifie en première analyse par le simple raisonnement suivant: sachant

$$\dot{x} = J_x \dot{q} \quad (18)$$

et donc:

$$\ddot{x} - J_x \ddot{q} = J_x \dot{q}, \quad (19)$$

soit, du point de vue de la commande:

$$u_x - J_x \dot{q} = J_x u_q \quad (20)$$

La solution générale en u_q de (20) s'écrit:

$$u_q = J_x^g(u_x - J_x \dot{q}) + u_K \quad (21)$$

pour tout $u_K \in \text{Ker}(J_x)$, noyau de l'application J_x lorsqu'il est non nul.

L'équation d'évolution correspondante en q devient:

$$\ddot{q} = J_x^g(u_x - J_x \dot{q}) + u_K \quad (22)$$

soit donc, dans l'espace de la tâche, avec (19):

$$\ddot{x} = J_x J_x^g u_x + (I - J_x J_x^g) J_x \dot{q} \quad (23)$$

Si J_x^g est une inverse généralisée réflexive, J_x^r , ce que nous supposons dans la suite, alors la propriété (A1-4) (annexe 1) permet d'écrire:

$$J_x^r(\ddot{x} - u_x) = 0 \quad (24)$$

c'est à dire:

$$\ddot{x} - u_x \in \text{Ker}(J_x^r)$$

Cette relation exprime le comportement en x du système, quelle que soit sa classe, lorsque l'on lui applique une commande de type (21): par exemple :

a- Si le système est de classe 3, la solution de norme euclidienne minimale est alors:

$$(J_x^T J_x)^{-1} J_x^T (\ddot{x} - u_x) = 0 \quad (25)$$

b- Si le système est redondant de rang plein ou régulier, J_x est surjective, donc J_x^r est injective de par sa définition, et $\text{Ker}(J_x^r)$ est nul. (24) se réduit alors à l'équation (11) qui, pour un système redondant, exprime la priorité de la tâche définie par x .

A partir de maintenant et sauf mention contraire, nous nous intéressons aux systèmes de classe 2. Nous pouvons alors toujours écrire:

$$u_K = P^r u_x, \quad u_x \in \mathbb{R}^n \quad (26)$$

où P^r est une matrice de projection sur le noyau $\text{Ker}(J_x)$:

(*): L'annexe 1 rappelle les définitions et principales propriétés des inverses généralisées, tandis que l'annexe 2 donne différents exemples d'inverses généralisées utilisables et utilisées dans la littérature.

$$P^r = I - J_x^r J_x$$

Alors, (22) implique, toujours d'après la propriété A1-4:

$$P^r (\ddot{q} - u_z) = 0 \quad ($$

Si J_x^r est la classique pseudo-inverse J_x^+ , alors la norme utilisée est euclidienne (pondération identité), et P est la matrice de projection orthogonale sur $\text{Ker}(J_x)$.

Une autre inverse généralisée, naturelle mais peu utilisée, peut être aisément reliée à la commande en espace opérationnel: en effet, en combinant les expressions (5) et (13) à (15) en l'absence de forces extérieures, on obtient naturellement l'expression de u_q correspondant au modèle opérationnel sous la forme:

$$u_q = M^{-1} J_x^T M' (u_x - \dot{J}_x \dot{q}) \quad (29)$$

Comme précédemment, l'extension au cas redondant se fait en écrivant:

$$u_q = J_x^g (u_x - \dot{J}_x \dot{q}) + u_K \quad (30)$$

où:

$$J_x^g = M^{-1} J_x^T (J_x M^{-1} J_x^T)^{-1} \quad (31)$$

Or, il est facile de montrer que (31) correspond à la solution $\dot{q} = J_x^g \dot{x}$ de $\dot{x} = J_x \dot{q}$ qui minimise à chaque instant l'énergie cinétique $\dot{q}^T M \dot{q}$ (cf annexe 2).

Pour résumer (provisoirement), (24) et (28) caractérisent donc complètement le comportement d'un système de n'importe quelle classe soumis à une commande de type (5) et (21), avec un modèle dynamique exact, soit donc:

a- Commande

$$l' = M(q) u_q + N(q, \dot{q}) \quad (5)$$

$$u_q = J_x^r (u_x - \dot{J}_x \dot{q}) + u_K \quad (21)$$

$$u_K \in \text{Ker}(J_x) = P^r u_z$$

b- Trajectoires

$$J_x^r (\ddot{x} - u_x) = 0 \quad (24)$$

$$P^r (\ddot{q} - u_z) = 0 \quad (28)$$

et l'équation d'évolution en q se décompose sur deux sous espaces complémentaires:

$$J_x \ddot{q} = u_x - \dot{J}_x \dot{q} \quad , \quad \dim(u_x) = m \quad (32)$$

$$P^r \ddot{q} = P^r u_z \quad , \quad \text{rang } n - m \quad (33)$$

u_z est une commande supplémentaire que l'on utilisera pour réaliser au mieux les tâches secondaires. Dans le cas redondant, il est facile de voir (cf Annexe 2) que ceci revient à trouver u_q qui minimise $\|u_q - u_z\|_q^2$ sous la contrainte $u_x = J_x u_q + J_x \dot{q}$.

A noter d'emblée que, dans le contexte dynamique dans lequel nous nous trouvons, il faut assurer la stabilité, d'une part de (32), ce qui se fait à travers la commande de x , d'autre part de (33), ce qui implique que l'on ne peut pas mettre n'importe quoi dans u_z , et qu'en particulier un terme d'amortissement y est

bienvenu pour assurer la stabilité asymptotique.

Enfin, et ceci justifiera le choix de considérer les modèles comme exacts dans la suite, examinons rapidement les conséquences d'un mauvais choix de $\hat{M}$ et $\hat{N}$ sur les trajectoires précédentes: l'évolution en q s'écrit:

$$\dot{q} = M^{-1} \hat{M} [J_x^T (u_x - J_x \dot{q}) + u_K] + M^{-1} (\hat{N} - N) \quad (34)$$

ce qui conduit, en x , à:

$$\ddot{x} = J_x M^{-1} \hat{M} J_x^T u_x + e(q, \dot{q}, u_K) \quad (35)$$

avec :

$$e(q, \dot{q}, u_K) = (I - J_x M^{-1} \hat{M} J_x^T) J_x \dot{q} + J_x M^{-1} \hat{M} J_x^T u_K + J_x M^{-1} (\hat{N} - N) \quad (36)$$

au lieu de (11).

L'équation de projection (28) devient quant à elle:

$$P^g (M^{-1} \hat{M} \dot{q} - u_x - M^{-1} (\hat{N} - N)) = 0 \quad (37)$$

Il n'y a donc plus, entre autres, découplage parfait entre tâche principale et tâche secondaire, la première étant perturbée par la seconde en raison du terme (36). (37) est également une forme perturbée de (28), et l'utilisation de grands gains peut devenir nécessaire (par analogie avec [32]).

1.3. Les principaux schémas de commande de systèmes redondants dans la littérature.

L'annexe 2 donne quelques détails sur diverses façons d'utiliser et de calculer des inverses généralisées. Nous nous contenterons donc dans ce paragraphe d'effectuer un rapide survol de la littérature sur le thème de la redondance.

Les premiers travaux sur le sujet furent proposés par Liégeois [29] et Fournier [14], dans un contexte cinématique. Ils utilisent la solution générale à l'inversion de $\dot{x} = J \dot{q}$:

$$\dot{q} = J_x^+ \dot{x} + (I - J_x^+ J_x) z \quad (38)$$

où:

$$J_x^+ = J_x^T (J_x J_x^T)^{-1} \quad (39)$$

et en choisissant:

$$z = -\lambda \frac{dc}{dq} \quad (40)$$

c étant le critère à minimiser. Cette approche s'interprète comme étant la projection d'une méthode de gradient:

$$P(\dot{q} + \lambda \frac{dc}{dq}) = 0 \quad (41)$$

Cela est lié à la minimisation de c sous la contrainte de réalisation de la tâche principale. Cette idée a, depuis, été fortement exploitée, principalement en choisissant des fonctions c variées.

On peut trouver en [25] une analyse des approches utilisant les pseudo-inverses; en référence [24], le même auteur applique cette méthode à un problème d'évitement d'obstacle, en assignant au point du robot le plus proche de l'obstacle une vitesse désirée d'éloignement. Dans le même article, Klein

compare quelques mesures de *dextérité*, et suggère d'utiliser la valeur singulière minimale.

Cette notion de dextérité est également considérée par Yoshikawa [34] [35], qui propose de maximiser la *manoeuvrabilité*

$$w = \sqrt{\det(JJ^T)} \quad (42)$$

qui n'est autre que le produit des valeurs singulières de J . D'autres indices de performances peuvent d'ailleurs être construits à partir de ces valeurs singulières ([33]). Remarquons à ce propos que, d'une façon générale, éviter les singularités consiste à éviter de rendre J non surjectif, donc de lui faire diminuer son rang. Toute mesure plus ou moins directe du rang de J est donc un critère possible; ainsi, la décroissance vers zéro d'une ou plusieurs valeurs singulières (donc de w), pourrait représenter une tendance à la chute du rang.

Dans le même article, l'auteur donne également un critère simple pour l'évitement d'obstacles, en définissant comme objectif un vecteur q_r "nominal" déterminé par apprentissage. En [34], le même auteur utilise la manoeuvrabilité pour optimiser à priori la structure géométrique de manipulateurs simples. Enfin, en [33], il étend cette notion à la manoeuvrabilité dynamique, qui caractérise en quelque sorte la capacité d'accélération du système: en supposant que les couples articulaires Γ_i sont bornés en valeur absolue par Γ_i^m , à vitesse nulle, l'ellipsoïde de manoeuvrabilité est défini par:

$$\bar{\gamma}^T (\bar{M}J^+)^T \bar{M}J^+ \bar{\gamma} \leq 1 \quad (43)$$

où:

$$\bar{J} = \text{diag} [1 / \gamma_j] J \quad (44)$$

$$\bar{M} = \text{diag} [1 / \Gamma_i^m] M \quad (45)$$

$$\bar{\gamma}_j = \frac{\gamma_j}{\gamma_j^m} \quad \forall j \quad (46)$$

$$\gamma = \ddot{x} - J\ddot{q} \quad (47)$$

$$\bar{\Gamma} = \Gamma - N(q, \dot{q}) \quad (48)$$

γ_j^m étant une sorte d'accélération maximale désirable. Un indice de manoeuvrabilité correspondant est alors :

$$w' = (\det \bar{J} (\bar{M}^T \bar{M})^{-1} \bar{J}^T)^{1/2} \quad (49)$$

La manoeuvrabilité est aussi un critère considéré par Nakamura. En [31], il utilise, en premier lieu, une approche analogue à celle de Klein, puis propose une méthode d'optimisation globale en minimisant (numériquement) à l'aide du principe du maximum:

$$c = \int_{t_0}^t \left(\frac{1}{w} + \dot{q}^T \dot{q} \right) dt \quad (50)$$

Remarquons que cette optimisation n'est pas effectuée sur un modèle dynamique, ce qui réduit l'intérêt du terme "énergétique" $\dot{q}^T \dot{q}$.

Toujours dans un cadre cinématique, mentionnons l'approche de Kircanski [23], qui introduit dans la matrice de pondération W définissant la norme que doit minimiser la pseudo-inverse, des termes dépendant d'une "distance" mesurée entre robots et obstacles.

En [3], Bailleul commence par étudier les singularités d'un système plan de dimension trois et montre comment certaines singularités sont ou non évitables. Ainsi, dans le cas du système présenté figure 1, les singularités évitables sont :

$$(q_2, q_3) = \begin{cases} (0, \pi) \\ (\pi, \pi) \end{cases} \quad (51)$$

et les singularités inévitables sont:

$$(q_2, q_3) = \begin{cases} (0, 0) \\ (\pi, 0) \end{cases} \quad (52)$$

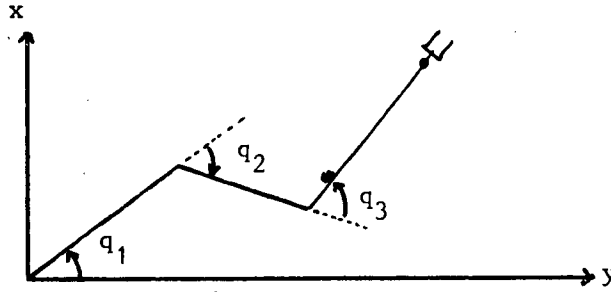


Figure 1

Puis, après avoir contredit l'assertion selon laquelle minimiser $||\dot{q}||$ permet d'éviter les singularités, il propose une méthode dans le cas de dimensions $m=2$, $n=3$, qui utilise le fait que $\text{Ker}(J)$ est engendré par: $N = j_1 \times j_2$, où les j_i sont les lignes de J_x : cela revient à choisir, dans (38):

$$z = \frac{N}{||N||^2} \quad (53)$$

Remarquons cependant que cette méthode ne permet pas d'assurer l'arrêt du système: ainsi, si $\dot{x}=0$, alors:

$$\dot{q} = \frac{N}{||N||^2} \quad (54)$$

d'où :

$$||\dot{q}|| = \frac{1}{||N||} \quad (55)$$

Cette expression ne devient nulle que si $||N|| \rightarrow \infty$. Or, $||N||$ est borné si $||j_1||$ et $||j_2||$ le sont...

L'article s'achève sur une proposition de méthode utilisant la redondance pour minimiser à chaque instant une fonction du couple dans un contexte dynamique (voir aussi [19]), soit avec nos notations usuelles:

$$u_x = \min ||MPu_x - \Gamma^*||^2 \quad (56)$$

avec:

$$\Gamma^* = \Gamma^* - \Gamma^* \quad (57)$$

$$\Gamma^* = MJ_x^+(u_x - J_x \dot{q}) + N(q, \dot{q}) \quad (58)$$

où Γ^* est un couple "objectif" instantané, par exemple:

$$\Gamma^* = \frac{1}{2}(\Gamma_{\min}(\cdot) + \Gamma_{\max}(\cdot)) \quad (59)$$

Les auteurs considèrent également le cas d'une minimisation pondérée du critère, à l'aide d'une matrice diagonale dont les éléments sont les inverses des carrés des couples limites. Examinons à travers un exemple simple le comportement induit par cette approche: soit $M=I$, $N=0$. Alors:

$$u_z = P\Gamma^* \quad (60)$$

et, d'après (28), le comportement de la tâche supplémentaire est donné par :

$$P(\ddot{q} - \Gamma^*) = 0 \quad (61)$$

Γ^* étant donné de façon instantanée, nous retrouvons les problèmes signalés plus haut concernant la stabilité de cette équation. Ainsi, le choix $\Gamma^*=0$ revient à utiliser la seule pseudo-inverse dans la commande, et n'assure donc jamais l'arrêt du système. Par contre, curieusement, un choix plus réaliste de Γ^* peut introduire un amortissement bénéfique: considérons en effet un modèle simplifié de moteur électrique; le couple disponible à chaque instant est limité dissymétriquement par la force contre-électro-motrice, de telle sorte que ([9]):

$$\Gamma_1^i = -a_i - b_i \dot{q}_i \leq \Gamma_i \leq a_i - b_i \dot{q}_i = \Gamma_2^i, \quad a_i, b_i > 0 \quad (62)$$

Alors, le choix

$$\Gamma^* = \frac{1}{2}(\Gamma_1 + \Gamma_2) = -diag(b_i) \dot{q}_i \quad (63)$$

peut stabiliser asymptotiquement (61).

Toujours dans un contexte dynamique, il existe un ensemble de méthodes, orientées vers l'évitement d'obstacles, utilisant la notion de potentiels. Introduites par Khatib ([20],[22]), ces méthodes associent à des objets modélisés des volumes enveloppants permettant de générer un potentiel newtonien, répulsif pour les obstacles, attractif pour la cible (voir aussi Hogan [18]). Les forces dérivant de ce potentiel sont appliquées en certains points x_k du robot. En l'absence de minimum local, la trajectoire permettant d'atteindre le minimum du potentiel est obtenue en ajoutant aux couples de contrôle Γ le couple:

$$\Gamma_{ext} = \sum_k (J_k^t)^T(x_k) F_k \quad (64)$$

où J_k^t est le Jacobien associé au point x_k .

L'extension au cas d'un système redondant avec tâche secondaire se fait naturellement en choisissant dans notre modèle général (21):

$$u_z = M^{-1}(q) \sum_{k=k_1}^{k_2} J_k^T F_k \quad (65)$$

Par exemple, $k_1=1$ et $k_2=n-1$ permettent de considérer l'évitement d'obstacles par les corps 1 à $n-1$ comme une tâche non prioritaire, par exemple par rapport au mouvement de l'organe terminal vers une cible. La stabilité de (28) implique alors, encore une fois, d'introduire un terme dissipatif dans F_k .

Une extension de la méthode des potentiels a été proposée par Krogh ([27]) : l'auteur remarque qu'il n'est peut-être pas judicieux d'exercer par exemple le même effort répulsif sur un bras immobile que sur un corps s'approchant ou s'éloignant rapidement de l'obstacle. Il propose donc une méthode tenant compte dans la définition des potentiels de la vitesse instantanée des points x_k .

Terminons cette brève revue en mentionnant le schéma d'anti-collision proposé par Freund ([15], [16]). Bien que le problème traité soit plus celui de la coordination entre bras que celui de l'évitement d'obstacles, sa méthode

présente l'intérêt d'être appliquée sur les équations dynamiques du système, ce qui est plutôt rare dans la littérature. Enfin, précisons pour le lecteur intéressé que les références suivantes peuvent être consultées avec profit pour tout ce qui concerne les inverses généralisées : [4], [6], [7], [14], [26].

Quelles remarques générales pouvons-nous à présent tirer de ce bref survol de la littérature? En premier lieu, aucune des méthodes proposées ne semble avoir été appliquée en ligne à un système réel : toutes les illustrations sont obtenues en simulation, et, dans la plupart des cas, dans des situations relativement simples : systèmes plans, peu de degrés de liberté, modèles d'environnement très simples, peu de capteurs, ceux-ci étant d'ailleurs considérés comme idéaux. En outre, les phénomènes dynamiques sont rarement, ou seulement partiellement pris en compte ; l'approche la plus fréquente consiste à trouver une solution inverse au modèle variationnel en exploitant la redondance pour réaliser une tâche secondaire. On crée alors une sorte de trajectoire désirée $\dot{q}_c(t)$, que l'on entrera comme consigne dans une commande "en vitesse". Cette démarche provient sans doute (historiquement) du fait que les actionneurs électriques de robots ont d'abord été (et sont encore généralement) des moteurs électriques classiques asservis en vitesse. Dans notre schéma général de commande, ceci revient à choisir, de façon simplifiée:

$$\Gamma = Ku_q \quad (66)$$

$$u_q = -K_v(\dot{q} - \dot{q}_c) \quad , K_v > 0 \quad (67)$$

$$\dot{q}_c = J^+ \dot{x}_c + Pz \quad (68)$$

D'où:

$$\ddot{q} = M^{-1}(Ku_q - N) \Rightarrow \quad (69)$$

$$\ddot{q} + M^{-1}KK_v\dot{q} = M^{-1}KK_vJ^+\dot{x}_c + M^{-1}KK_vPz - M^{-1}N \quad (70)$$

Soit donc :

$$\ddot{x} = -JM^{-1}KK_v\dot{q} + JM^{-1}KK_vJ^+\dot{x}_c + JM^{-1}KK_vPz - JM^{-1}N \quad (71)$$

qui ne produira le comportement désiré que si K_v est suffisamment grand. Avec un peu de dynamique ($K=M$), et le choix $K_v = \lambda I$, les choses s'améliorent toutefois, et l'on peut obtenir:

$$\ddot{x} = -\lambda(\dot{x} - \dot{x}_c) + J\dot{q} - JM^{-1}N \quad (72)$$

et:

$$P(\ddot{q} + \lambda\dot{q} - \lambda z) = -PM^{-1}N \quad (73)$$

Dans ces conditions, il vaut mieux prendre d'emblée en compte les phénomènes dynamiques, comme nous l'avons proposé au paragraphe 1.2. Nous allons à présent détailler la démarche à suivre dans le cas des systèmes de classe 2.

2. COMMANDE DYNAMIQUE DES SYSTEMES REDONDANTS

2.1. Introduction

Les expressions générales de commande (5) et (21) ne résolvent pas par elles-mêmes le problème de la synthèse de la commande du système pour une application particulière : il reste en effet, d'une part à choisir l'inverse généralisée J^r , d'autre part à construire la commande $u_k = P^r u_z$ correspondant à la tâche non prioritaire. En pratique, on voit mal comment appréhender aisément ces deux aspects. Par ailleurs, l'interprétation précise de l'équation (28) en terme de dynamique de la tâche non prioritaire est malaisée, en raison de la présence du terme P^r . Nous allons donc essayer de relier l'approche par inverses généralisées à une démarche plus intuitive permettant de construire des commandes.

Remarquons en premier lieu que, parmi les systèmes caractérisés dans le tableau 1, ceux de classe 1 posent le moins de problèmes. Il est donc tentant de chercher à s'y ramener ; pour cette raison, certains auteurs ont proposé d'ajouter au système $n-m$ contraintes indépendantes, de la forme $b(q)=0$, de façon à le réduire à la classe 1, en dimension m , et, par un choix adéquat de $b(q)$, assurer par exemple que les singularités sont évitées : soit la partition:

$$q^T = [q_1^T \ q_2^T] , \quad \dim q_1 = m \quad (74)$$

En choisissant $b(q)$ telle que l'on puisse expliciter:

$$b(q_1, q_2) = 0 \Rightarrow q_2 = d(q_1) \quad (75)$$

alors, (1) s'écrit:

$$x = f_x(q_1, d(q_1)) = f_1(q_1) \quad (76)$$

et le Jacobien associé est de dimension m . Par exemple, dans l'exemple de la figure 1, il suffit d'imposer

$$q_2 + q_3 = \alpha \neq k\pi \quad (77)$$

pour éviter les singularités.

Cette idée se heurte néanmoins à une réalité évidente : à moins d'effectuer un montage mécanique réalisant la contrainte celle-ci à toutes les chances de n'être qu'approximativement satisfaite par les seuls asservissements, ce qui peut conduire à des erreurs si l'on fait confiance au modèle (76).

D'où l'idée de considérer la satisfaction de contraintes comme de nouvelles *variables* à contrôler au même titre que les autres. Alors le nouveau système de classe 1 sera de dimension $n \times n$. C'est ce point de vue que nous allons développer à présent.

2.2. Equivalence entre augmentation de la dimension de la tâche et inverses généralisées

Ajoutons donc à l'équation (1) les $n-m$ variables supplémentaires:

$$s = f_s(q) , \quad (78)$$

le jacobien associé étant:

$$J_s = \left[\frac{df_s}{dq} \right] , \quad (79)$$

Le jacobien global

$$J = \begin{bmatrix} J_x \\ J_s \end{bmatrix} \quad (80)$$

est de dimension $n \times n$. Remarquons d'emblée que J peut présenter des singularités, différentes de celles qui font chuter le rang de J_x , et dues à J_s . Cela signifie qu'en ces domaines singuliers les tâches décrites par x et s ne sont pas indépendantes.

Supposons pour l'instant la régularité de J . Alors, il existe toujours une et une seule partition:

$$J^{-1} = (J_x^g \ J_s^g) \quad (81)$$

telle que :

$$\begin{cases} \text{Ker } J_s = \text{Im } J_x^g \\ \text{Ker } J_x = \text{Im } J_s^g \end{cases} \quad (82)$$

Une façon générale d'écrire J_x^g et J_s^g est donnée par:

$$J_x^g = J_x^T W^{-1} - J_s^T V^{-1} J_s J_x^+ = (I - P_x J_s^T V^{-1} J_s) J_x^+ \quad (83)$$

$$J_s^g = J_s^T V^{-1} - J_x^T W^{-1} J_x J_s^+ = (I - P_s J_x^T W^{-1} J_x) J_s^+ \quad (84)$$

où:

$$V = J_s P_x J_s^T \quad (85)$$

$$W = J_x P_s J_x^T \quad (86)$$

et:

$$P_x = I - J_x^+ J_x = R_x (R_x^T R_x)^{-1} R_x^T \quad (87)$$

$$P_s = I - J_s^+ J_s = R_s (R_s^T R_s)^{-1} R_s^T \quad (88)$$

R_i étant une matrice complète de vecteurs de base de $\text{Ker } J_i$, et J_i^+ étant donnée par une pseudo-inverse du type (39), pour $i = x$ et s . Toutes ces expressions peuvent être obtenues de diverses manières, par exemple en utilisant le Lagrangien:

$$\Lambda = \frac{1}{2\alpha} \dot{q}^T \dot{q} + \lambda^T (\dot{x} - J_x \dot{q}) + \mu^T (\dot{s} - J_s \dot{q}) \quad (89)$$

Donnons d'emblée un exemple de calcul de J_x^g et J_s^g : supposons qu'il existe un réordonnancement des composantes q_i de q tel que la partition:

$$J_x = \begin{bmatrix} J_{x1} & J_{x2} \end{bmatrix} \quad (90)$$

assure la régularité de J_{x1} (dimensions $m \times m$). Alors:

$$J_s^g = R_x H^{-1} \quad (91)$$

où J_s^g est de dimensions $n \times n - m$, et H , supposée régulière (voir paragraphe 2.3.3), est donnée par:

$$H = J_s R_x = J_{s2} - J_{s1} J_{x1}^{-1} J_{x2} \quad (92)$$

avec:

$$J_s = \begin{bmatrix} J_{s1} & J_{s2} \end{bmatrix}, \quad \dim(J_{s2}) = n - m \times n - m \quad (93)$$

et:

$$R_x = \begin{bmatrix} -J_{x1}^{-1} J_{x2} \\ I \end{bmatrix}, \quad (94)$$

matrice de vecteurs de base de $\text{ker } J_x$. Alors:

$$J_x^g = \begin{bmatrix} I + J_{x1}^{-1} J_{x2} H^{-1} J_{s1} \\ -H^{-1} J_{s1} \end{bmatrix} J_{x1}^{-1} \quad (95)$$

Ces relations peuvent également s'écrire, en imposant cette fois la régularité de J_{s2} :

$$J_x^g = \begin{bmatrix} I \\ -J_{s1} J_{s2}^{-1} \end{bmatrix} K^{-1} \quad (96)$$

$$J_s^g = \begin{bmatrix} -K^{-1} J_{x2} \\ I + J_{s1} J_{s2} K^{-1} J_{x2} \end{bmatrix} J_{s2}^{-1} \quad (97)$$

$$K = J_{x1} - J_{x2} J_{s2}^{-1} J_{s1} \quad (98)$$

On vérifie aisément que ces expressions satisfont:

$$J_x J_x^g = J_s J_s^g = I \quad (99)$$

$$J_x J_s^g = J_s J_x^g = 0 \quad (100)$$

Par ailleurs, si on définit les projecteurs sur $\text{Ker } J_x$ et $\text{Ker } J_s$:

$$P_x^g = I - J_x^g J_x \quad (101)$$

$$P_s^g = I - J_s^g J_s \quad (102)$$

alors :

$$P_x^g J_x^g = P_s^g J_s^g = 0 \quad (103)$$

$$P_s^g J_x^g = J_x^g \quad (104)$$

$$P_x^g J_s^g = J_s^g \quad (105)$$

Nous choisissons donc, dans (21):

* J_r donnée par J_x^g dans (95),

* $u_K = u_x = J_s^g u'_s$ (106)

où J_s^g est donnée par (91) et

$$u'_s = u_s - \dot{J}_s \dot{q} \quad (107)$$

u_s étant la commande associée à s , comme u_x est associée à x .

Ces expressions nous permettent donc de réaliser les trajectoires désirées:

$$\ddot{x} = u_x \quad (108)$$

$$\ddot{s} = u_s \quad (109)$$

L'utilisation des inverses généralisées explicite donc en quelque sorte les contributions respectives des tâches principale et secondaire, et fournit, d'une certaine manière une méthode d'inversion de J .

Remarque importante

Les (rares) commandes dynamiques redondantes de la littérature utilisent le schéma :

$$u_q = J_x^+(u_x - \dot{J}_x \dot{q}) + P_x z \quad (110)$$

où J_x^+ est la pseudo-inverse minimisant la norme euclidienne, parfois avec une pondération. Ceci n'est bien sûr pas équivalent au schéma qui vient d'être

exposé, car pour assurer le découplage entre les tâches, il faut utiliser les bonnes inverses généralisées, données par (91) et (95).

Les deux approches ne deviennent complètement équivalentes que si l'on choisit s de telle sorte que $J_s = R_x^T$, ce qui n'est pas à priori le cas général ; par contre les deux termes $P_x z$ et $J_s^g u'_s$ peuvent être identifiés en choisissant simplement $z = J_s^g u'_s$. Alors, comme z appartient à $\text{Ker } J_x$:

$$P_x z = z = J_s^g u'_s \quad (111)$$

2.3. Choix des variables supplémentaires

Comme nous l'avons indiqué, les tâches non prioritaires auxquelles sont liées les variables supplémentaires sont habituellement définies à partir des objectifs suivants :

- s'éloigner des singularités
- s'éloigner des butées articulaires
- maximiser la manipulabilité
- éviter les obstacles ou les collisions avec d'autres robots
- s'approcher d'une trajectoire de référence donnée, obtenue par exemple par apprentissage
- minimiser l'énergie dépensée.

Dans certains cas, il est immédiat de déterminer les variables supplémentaires requises: ainsi, dans l'exemple de la figure 1, on peut prendre :

$$s = q_2 + q_3 - a \quad (112)$$

ou encore :

$$s = n^T n - b \quad (113)$$

$n^T n$ représente la somme des carrés des valeurs singulières de J_x .

Plus généralement, et comme indiqué au paragraphe 1.1, on cherchera naturellement à exprimer des variables supplémentaires correspondant à la recherche du minimum d'une fonction $c(q, t)$. Comme il s'agit d'une tâche secondaire, cette minimisation doit être effectuée sous contrainte de réalisation de la tâche principale.

2.3.1. Minimum statique sous contrainte

Considérons donc le problème *instantané* :

$$\left| \begin{array}{l} \text{minimiser } c(q) \\ \text{sous la contrainte } x = f(q) \end{array} \right|$$

Plaçons nous dans les hypothèses requises par le théorème des fonctions implicites, dont l'énoncé est ([8]):

Soient $f_i(x, y)$ p fonctions scalaires définies et continûment différentiables dans un voisinage $U \times V$ d'un point $(a_1, \dots, a_m, b_1, \dots, b_p)$ de $E \times F$, telles que $f_i(a_1, \dots, a_m, b_1, \dots, b_p) = 0$ pour $i \leq p$, et telles que le

le jacobien $\frac{\partial(f_1, \dots, f_p)}{\partial(y_1, \dots, y_p)}$ soit différent de 0 au point $(a_1, \dots, a_m, b_1, \dots, b_p)$. Alors, il existe un voisinage ouvert $W_0 \subset U$ de $(a_1, \dots, a_m)$ tel que, pour tout voisinage ouvert connexe $W \subset W_0$ de $(a_1, \dots, a_m)$, il existe un système unique de p fonctions scalaires $g_i (1 \leq i \leq p)$, et:

$f_i(x_1, \dots, x_m, g_1(x_1, \dots, x_m), \dots, g_p(x_1, \dots, x_m)) = 0$ pour $1 \leq i \leq p$ et tout $(x_1, \dots, x_m) \in W$. De plus, les fonctions g_i sont continuellement différentiables dans W , et la matrice jacobienne $(D_j g_i(x))$ est égale à $-B^{-1}A$, où A (resp B) est obtenue en remplaçant y_i par $g_i(x_1, \dots, x_m)$ ($1 \leq i \leq p$) dans la matrice jacobienne $(\partial f_i / \partial x_k)$ (resp $(\partial f_i / \partial y_j)$).

Ainsi, reprenons la partition:

$$J_x = \left\{ \frac{\partial f}{\partial q} \right\} = \begin{bmatrix} J_{x1} & J_{x2} \end{bmatrix}$$

$$\dim J_{x1} = m \times m, \quad \dim J_{x2} = m \times n - m$$

Alors, si $\det J_{x1} \neq 0$, et en partitionnant:

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \quad \dim q_1 = m, \quad \dim q_2 = n - m, \quad (114)$$

il existe h telle que:

$$q_1 = h(x, q_2) \quad (115)$$

D'où:

$$c(q) = c(h(x, q_2), q_2) = d(x, q_2) \quad (116)$$

et le minimum sous contrainte q_2^* est obtenu en écrivant:

$$\frac{\partial d}{\partial q_2}(q_2^*) = 0 \quad (117)$$

Il s'agit bien d'un minimum si le Hessien correspondant est positif. Sous forme Lagrangienne, ce problème s'exprime en cherchant le minimum de:

$$\Lambda = c(q) + \lambda^T (x - f(q)) \quad (118)$$

D'où, en posant $g(q) = \frac{dc}{dq}$:

$$g(q) - J_x^T(q)\lambda + \left(\frac{d\lambda}{dq}\right)(x - f(q)) = 0 \quad (119)$$

En imposant la satisfaction de la contrainte et en projetant par P_x sur $\ker J_x$, il vient:

$$P_x g = 0 \quad (120)$$

qui exprime le résultat bien connu selon lequel le minimum sous contrainte est obtenu lorsque le gradient est orthogonal au noyau.

Les n équations de (120) ne sont pas indépendantes. On obtient $(n-m)$ équations indépendantes en multipliant (120) par une matrice $n-m \times n$ convenable, de rang plein, par exemple R_x^T matrice de vecteurs de base de $\ker J_x$.

Alors (120) devient :

$$R_x^T g = 0 \quad (121)$$

Ainsi, en utilisant la valeur de R_x donnée par (94), et toujours sous la condition de régularité de J_{x1} , (121) s'écrit :

$$g_2^T - g_1^T J_{x1}^{-1} J_{x2} = 0 \quad (122)$$

$(g_1^T \ g_2^T)$ étant la partition de g^T correspondant à (114). Cette expression peut d'ailleurs être directement dérivée de (117).

2.3.2. Variables supplémentaires associées au problème de minimisation.

Une idée naturelle consiste à choisir comme variables supplémentaires les $n-m$ expressions (121) ou (122). Ce dernier choix donne

$$s = (g_2^T - g_1^T J_{x1}^{-1} J_{x2})^T \quad (123)$$

La commande de s vers 0 se fait alors avec la boucle:

$$u_s = -K_p s - K_v \dot{s} \quad , \quad K_p, K_v > 0 \quad (124)$$

soit donc :

$$u_s = -K_p R_x^T g - K_v J_s \dot{q} \quad (125)$$

qui conduit, dans (106), à :

$$u_x = J_s^T (-K_p R_x^T g - K_v J_s \dot{q} - J_s \ddot{q}) \quad (126)$$

soit donc la forme générale :

$$u_x = -A_1 g - A_2 \dot{q} - C(q, \dot{q}) \quad (127)$$

où :

$$A_1 = R_x H^{-1} K_p R_x^T \quad (128)$$

$$A_2 = J_s^T K_v J_s \quad (129)$$

$$C(q, \dot{q}) = J_s^T g = J_s^T J_s \dot{q} \quad (130)$$

à comparer à l'approche pseudo-inverse dans laquelle on choisit naturellement :

$$A'_1 = -\lambda P_x \quad (131)$$

(ou $-P_x M^{-1}$ dans les commandes de type opérationnel)

$$A'_2 = K_v \quad (132)$$

$$C(q, \dot{q}) = 0 \quad (133)$$

On voit d'emblée apparaître quelques difficultés de base liées à l'utilisation de (126):

1- Sans condition supplémentaire, rien n'indique que l'on atteint effectivement un minimum.

2- Si d'aventure J_{x1} devient singulier, il faut modifier l'ordre des q_i , de façon à obtenir un nouveau J'_{x1} inversible, et cette opération introduit des discontinuités dans le système.

3- La complexité de s peut conduire à des expressions analytiques relativement lourdes, en particulier dans le calcul de J_s .

Les problèmes essentiels tournent toutefois autour des singularités du système, et nous allons les examiner plus en détail.

2.3.3. Convergence de la minimisation sous contraintes

Revenons pour quelque temps au modèle variationnel :

$$\dot{q} = J_x^T \dot{x} + J_s^T \dot{s} \quad , \quad (134)$$

de façon à examiner le problème de convergence.

Supposons choisi pour s un comportement exponentiel découplé :

$$\dot{s} = -\lambda s \quad (135)$$

et plaçons-nous dans le cas où la tâche principale, $\dot{x}$, est nulle et parfaitement réalisée.

Alors, comme

$$\dot{c} = g^T \dot{q} \quad , \quad (136)$$

il vient, avec (91), (135), (136) :

$$\dot{c} = -\lambda R_x H^{-1} s = -\lambda R_x H^{-1} R_x^T g \quad (137)$$

Une condition immédiate pour assurer la décroissance de c , donc la descente vers le minimum, est :

$$H > 0$$

Cela est logique, car $H = J_s R$ n'est autre que le Hessien correspondant au gradient réduit ([17]) $\frac{\partial d}{\partial q_2} = g'$ (cf(117)) de la minimisation sous contrainte.

L'inversibilité de H est liée à celle de J . Ce résultat se démontre aisément en partitionnant J^{-1} ; cherchons en effet A, B, C, D tels que :

$$\begin{pmatrix} J_{x1} & J_{x2} \\ J_{s1} & J_{s2} \end{pmatrix} \begin{pmatrix} A & B \\ C & D \end{pmatrix} = I \quad (138)$$

L'explicitation du produit conduit à :

$$\begin{cases} (J_{s2} - J_{s1} J_{x1}^{-1} J_{x2}) C = -J_{s1} J_{x1}^{-1} \\ (J_{s2} - J_{s1} J_{x1}^{-1} J_{x2}) D = I \end{cases} \quad (139)$$

et l'inversion est possible si et seulement si :

$$J_{s2} - J_{s1} J_{x1}^{-1} J_{x2} = H \neq 0 \quad (140)$$

Les singularités de J et de $J_s R_x$ sont donc identiques.

Ainsi, si le Hessien $H(q^*)$ est supposé positif pour que q^* soit bien le minimum sous contrainte, celui-ci s'annule au passage par une singularité de J , et peut devenir négatif dans certaines parties de l'espace. A ce passage, nous perdons la régularité de l'application faisant correspondre s et q_2 . Remarquons que, si J_x était constant, nous aurions :

$$H = R_x^T G R_x \quad (141)$$

G étant le Hessien de $c(q)$, supposé positif ; la décroissance serait donc toujours assurée. Si, de plus, $G = I$, alors :

$$J_s^T = P_x \quad ; \quad J_x^T = J_x^+ \quad (142)$$

La décroissance est également assurée dans l'approche classique, où, avec $\dot{x}=0$:

$$\dot{c} = -\lambda g^T P_x g \leq 0 \quad (143)$$

Nous nous trouvons ici face à un problème bien connu en analyse numérique, où l'on sait que les méthodes au premier ordre (gradient simple) descendent dans la bonne direction si la fonction est unimodale, alors que les méthodes au deuxième ordre (Newton-Raphson) peuvent diverger si le Hessien est négatif (fonction non strictement convexe), mais convergent mieux au voisinage du minimum. Prenons un exemple très simple pour illustrer ce phénomène: soit :

$$x = \frac{1}{2}(q_1^2 + q_2^2) \quad (144)$$

et:

$$c(q) = \frac{1}{2}((q_1-1)^2 + (q_2-1)^2) \quad (145)$$

Alors :

$$J_x = (q_1 \ q_2) \quad (146)$$

$$J_x^+ = (q_1^2 + q_2^2)^{-1} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \quad (147)$$

$$P_x = (q_1^2 + q_2^2)^{-1} \begin{bmatrix} q_2^2 & -q_1 q_2 \\ -q_1 q_2 & q_1^2 \end{bmatrix} \quad (148)$$

$$g^T = (q_1 - 1 \quad q_2 - 1) \quad (149)$$

$$R_x^T = \left(-\frac{q_2}{q_1} \quad 1 \right) \quad (150)$$

avec la condition $q_1 \neq 0$.

La singularité de J_x est : $\{ q_1 = q_2 = 0 \}$, et nous supposons l'éviter. Soit $x = \dot{x} = 0$, supposés exactement réalisés. L'approche classique donne:

$$\dot{c} = -\lambda g^T P_x g = -\lambda \frac{(q_1 - q_2)^2}{q_1^2 + q_2^2} \quad (151)$$

L'utilisation de variables supplémentaires conduit à :

$$R_x^T g = \frac{q_2 - q_1}{q_1} \quad (152)$$

qui s'annule pour $q_1 = q_2$, ce qui nous suggère la variable :

$$s = q_2 - q_1 \quad (153)$$

D'où :

$$J_s = (-1 \quad 1) ; J_s R_x = \frac{q_1 + q_2}{q_1} \quad (154)$$

La nouvelle singularité est donc :

$$q_1 + q_2 = 0 \quad (155)$$

et le bouclage (135) donne :

$$\dot{c} = -\lambda \frac{(q_1 - q_2)^2}{q_1 + q_2} \quad (156)$$

Les deux comportements correspondants sont illustrés figure 2.

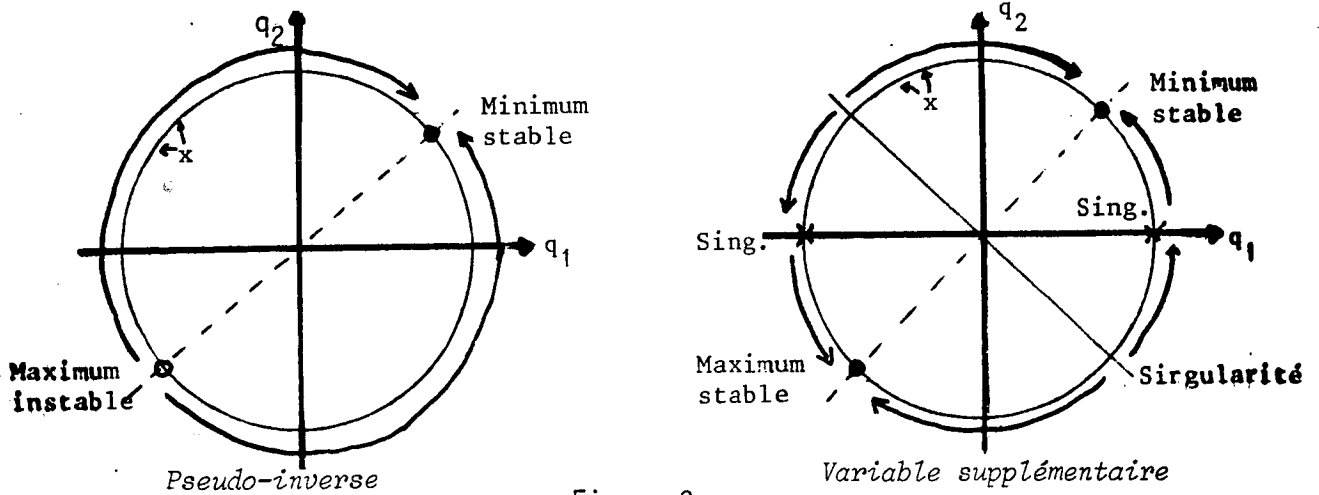


Figure 2

Nous avons ainsi, dans le cas classique (pseudo-inverse), une méthode qui tendra toujours à converger vers le minimum. Lorsque la dimension de $\text{Ker } J_x$ est 1, il n'y a souvent pas de raison majeure pour préférer l'approche par variables supplémentaires. Par contre, en dimension supérieure, apparaît toute la différence entre un algorithme à gain scalaire (gradient simple) et un algorithme à gain matriciel (cas des variables supplémentaires). Ce dernier réalise en effet, dans le cas linéaire, et, pour nous, dans le sous-espace sur lequel s'effectue la minimisation, un découplage parfait, c'est à dire une *sphérisation* de la convergence. Prenons à nouveau un exemple simple :

$$J_x = \begin{pmatrix} 1 & 2 & 1 \end{pmatrix} \quad (157)$$

$$c = \frac{1}{2}[(q_1 - 1)^2 + 2(q_2 - 1)^2 + (q_3 - 1)^2] \quad (158)$$

avec toujours $x = \dot{x} = 0$. Alors, la convergence donnée par la pseudo-inverse, une fois éliminée q_1 , est de la forme :

$$\dot{q}_2 = -\frac{4}{3}\lambda q^2 \quad (159)$$

$$\dot{q}_3 = -\lambda(q_2 + q_3) \quad (160)$$

Par contre, (135) conduit à :

$$\dot{q}_2 = -\lambda q_2 \quad (161)$$

$$\dot{q}_3 = -\lambda q_3 \quad (162)$$

D'une façon générale, (123) et (135) conduisent donc, quelque soit $\dot{x}$ à un algorithme du deuxième ordre :

$$\dot{q}_2 = -\lambda H^{-1} g' \quad (163)$$

alors que l'approche classique : pseudo-inverse + gradient simple ne réalise pas du tout ce découplage. En outre, cette dernière approche impose en pratique un

mode de convergence numérique selon la direction la plus lente, ce qui peut être peu efficace si la "vallée" autour de q^* est très étirée. En dynamique, le problème se pose également car (131), (132), et (133) donnent :

$$P_x(\ddot{q} + K_v \dot{q} + \lambda g) = 0 \quad (164)$$

qui est seulement la *projection* sur $\text{Ker } J_x$ d'une équation stable et convergente vers q^* .

2.4. Conclusion

Il faut donc résoudre le dilemme découplage / décroissance du critère posé par les deux approches. En pratique, deux solutions s'offrent à nous :

- Modifier le critère C de façon à assurer la positivité de H , par exemple en s'arrangeant pour être toujours proche du minimum, et appliquer intégralement la méthode des variables supplémentaires.

- utiliser une approche employée en analyse numérique, c'est à dire une méthode de gradient simple loin du minimum, et, dès que H devient significativement positive, une méthode au deuxième ordre. Rappelant que l'équivalence avec l'approche pseudo-inverse est obtenue lorsque $J_s = R_x^T$, une variante possible consiste à remplacer dans les équations (91) à (95) J_s par :

$$J'_s = \alpha(t) J_s + (1 - \alpha(t)) R_x^T, \quad 0 \leq \alpha \leq 1 \quad (165)$$

Le scalaire α est choisi à chaque instant de façon à rendre $H' = J'_s R_x$ positif, et aussi "près" que possible de H . Une situation extrême est la simple commutation : $\alpha = 0$ ou 1 , cette dernière valeur n'étant appliquée qu'une fois passées les singularités de J_s .

3. UTILISATION DE CAPTEURS POUR L'ÉVITEMENT D'OBSTACLES DES ROBOTS REDONDANTS

Cette dernière partie présente, à travers quelques résultats de simulation, une approche fondée sur la prise en compte de capteurs de proximité pour l'évitement d'obstacles. Le principe, succinctement décrit en [12], et plus largement développé en [5], consiste à construire une fonction coût dont le gradient peut être évalué à l'aide d'informations capteurs. Cette approche peut être grosso modo comparée à une méthode de potentiels dans laquelle les forces extérieures seraient fictives, mais d'intensité fonction des sorties capteurs réelles.

Au cours de ce paragraphe, nous commencerons par rappeler les principes utilisés, puis nous établirons les algorithmes employés, et enfin nous donnerons des résultats de simulation.

3.1. Synthèse des informations capteurs

Nous nous intéressons ici aux capteurs d'anti-collision locaux, c'est à dire portés par le robot. La mesure extérieure, par exemple par caméras, est évidemment possible, mais dans ce dernier cas ressort plus de l'analyse de

scène que de notre problématique de commande.

Comme indiqué en [11], nous supposons qu'à chaque corps L_k , $k=1, \dots, n$, est associé un ensemble de capteurs C_k^i pouvant générer en un ensemble de points x_k^i les actions répulsives élémentaires a_k^i (figure 3). La direction de chacune est fixée dans le repère R_k lié au corps L_k , mais l'amplitude dépend des sorties instantanées y_k^i . Nous pouvons donc associer à chaque corps le torseur :

$$S_k : \begin{cases} V_k = \sum_i a_k^i \\ \omega_k = \sum_i O_k \vec{x}_k^i \times \vec{a}_k^i \end{cases} \quad (166)$$

O_k étant l'origine d'un repère R_k lié à L_k .

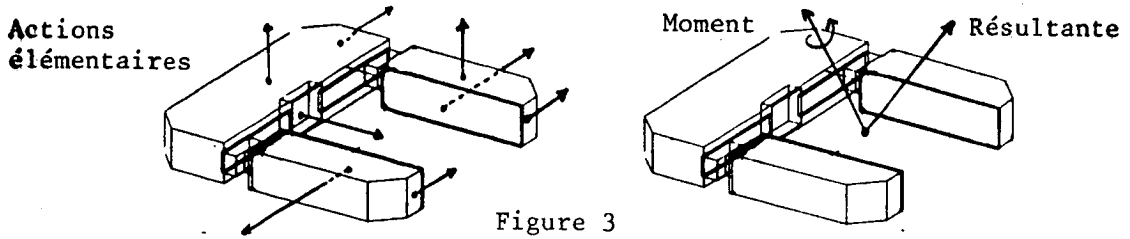


Figure 3

Soit J_k la matrice jacobienne associée à $\{ O_k, R_k \}$, exprimée dans R_k , et dans laquelle les composantes en translation et en rotation sont respectivement notées J_k^t et J_k^r ([11]). Nous supposons en outre que les sorties capteurs dépendent de la "distance" entre x_k^i et le premier obstacle situé dans la direction d'observation du capteur (cf figure 4). En pratique, pour une disposition d'obstacles donnée, les y_k^i ne dépendent donc que de q , le temps de réponse du capteur lui même étant négligé, hypothèse confirmée expérimentalement. Une situation intéressante pour l'évitement d'obstacles est celle où y_k^i a une allure voisine de $(d_k^i)^{-2}$ ([10]).

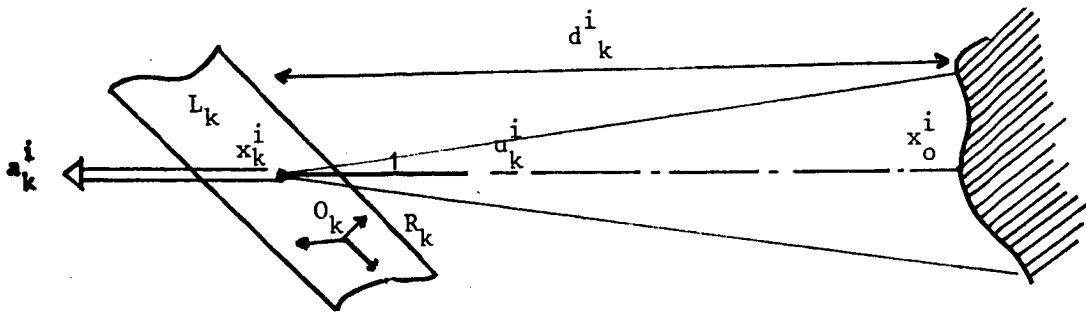


Figure 4

3.2. Approximation d'une méthode de potentiel

Chaque torseur S_k peut être interprété comme correspondant à un ensemble de forces ponctuelles a_k^i exercées en x_k^i sur le robot. L'approche donnée par les équations (64) et (65) peut alors s'appliquer en choisissant, pour le u_K de l'équation (30) :

$$u_K = P_x^g(M^{-1}(q)\Gamma_{ext} + K_v \dot{q}) \quad (167)$$

où :

$$\Gamma_{ext} = \sum_k J_k^T S_k \quad (168)$$

$K_v \dot{q}$ est un terme d'amortissement rendu nécessaire par le fait que S_k ne dépend que de q . L'évolution obtenue est alors :

$$P_x(\ddot{q} - M^{-1}\Gamma_{ext} + K_v \dot{q}) = 0 \quad (169)$$

et le couple de commande appliqué est :

$$\Gamma = M(J_x^g(u_x - \dot{J}_x \dot{q})) + N + M P_x^g(M^{-1}\Gamma_{ext} + K_v \dot{q}) \quad (170)$$

à comparer à :

$$\Gamma = M(J_x^{-1}(u_x - \dot{J}_x \dot{q}) + N + \Gamma_{ext}) \quad (171)$$

dans l'approche classique non redondante sans tâche secondaire.

Cette méthode prend sa justification lorsque les capteurs fournissent effectivement un signal voisin de $(d_k^i)^{-2}$ ([10]), car on se rapproche alors qualitativement du cas théorique où le potentiel associé aux obstacles est Newtonien. Les capteurs fournissent dans ce cas une mesure partielle et approchée du champ gradient correspondant.

Vu le caractère grossier de l'approximation effectuée, nous pouvons aller un peu plus loin et simplifier encore la méthode en se passant du terme M^{-1} de (167), qui ne sert qu'à transformer formellement les couples en accélérations. En reprenant le schéma de commande habituel du paragraphe 1.2, nous pouvons donc chercher u_q qui minimise :

$$\|u_q - \Gamma_{ext} + K_v \dot{q}\|_q^2$$

sous la contrainte

$$u_x = J_x u_q + \dot{J}_x \dot{q}$$

et l'on obtient ainsi :

$$u_q = J_x^g(u_x - \dot{J}_x \dot{q}) + P_x^g(\Gamma_{ext} - K_v \dot{q}) \quad (172)$$

où l'inverse généralisée dépend de la q -norme utilisée.

3.3. Gradient associé

L'hypothèse selon laquelle les capteurs permettent plus ou moins de mesurer le champ gradient associé à un potentiel répulsif semble a priori un peu forte; nous allons donc apporter quelques précisions.

Considérons donc (cf figure 4) un modèle élémentaire de capteur, dont la sortie est donnée par :

$$y_k^i = \frac{1}{\|d_k^i\|^2} \quad (173)$$

$$d_k^i = x_o^i - x_k^i \quad (174)$$

x_o^i étant supposé fixe.

le vecteur unitaire associé à la direction d_k^i est :

$$u_k^i = \frac{d_k^i}{||d_k^i||} \quad (175)$$

Choisissons donc de minimiser :

$$c_k^i - (d_k^{iT} d_k^i)^{-1} \quad (176)$$

Alors :

$$\frac{\partial c_k^i}{\partial x_k^i} \approx \frac{1}{||d_k^i||^4} d_k^i \quad (177)$$

Un algorithme de gradient associé à ce problème est :

$$\frac{dx_k^i}{dt} = -\mu_k^i \frac{1}{||d_k^i||^4} d_k^i \quad (178)$$

soit donc , finalement, avec (173) et (175) :

$$\vec{a}_k^i = \frac{dx_k^i}{dt} = -\mu_k^i (y_k^i)^{3/2} u_k^i \quad (179)$$

Cette expression permet donc de définir les actions élémentaires $\vec{a}_k^i$ de répulsion servant à construire le torseur S_k comme étant proportionnelles au gradient de c_k^i . Selon l'interprétation choisie, $\vec{a}_k^i$ est le gradient du potentiel, donc une *force* répulsive, ou bien tend à réaliser localement l'algorithme de gradient (177), et est donc une *vitesse* désirée d'éloignement. Remarquons que (179) diffère légèrement du choix intuitif :

$$\vec{a}_k^i = \frac{dx_k^i}{dt} = -\mu_k^i y_k^i u_k^i$$

qui ne correspond pas exactement au gradient de c_k^i , bien qu'étant naturellement dans la bonne direction.

Enfin, pour construire la synthèse globale, par exemple Γ_{ext} , il n'est pas nécessaire de passer par tous les jacobiens partiels associés aux points x_k^i . Le torseur S_k s'interprète en effet, du moins pour ce qui concerne sa composante V_k , comme une action de répulsion vis à vis d'un obstacle virtuel, rassemblant les contributions de tous les obstacles partiels.

3.4. Minimisation globale

3.4.1. Point de vue cinématique

Etant donné l'ensemble des torseurs S_k , une première approche consiste à chercher à obtenir des vitesses effectives les plus proches possibles de ceux-ci, sous la contrainte de réalisation de la tâche principale . Soit donc :

minimiser

$$\sum_k ||\dot{x}_k - S_k||_q^2$$

avec :

$$\dot{x} = J_x \dot{q} \quad (180)$$

et:

$$\dot{x}_k = \frac{d}{dt} O_0 \vec{O}_k = J_k \dot{q} \quad (181)$$

Les solutions à ce problème sont données en annexe A2. En particulier, si $k=1, \dots, n$, et pour une norme euclidienne, on obtient:

$$\dot{q} = J_x^g \dot{x} + P_x^g E^{-1} \sum_{k=1}^n J_k^T S_k \quad (182)$$

où:

$$E = \sum_{k=1}^n J_k^T J_k \quad (183)$$

supposée régulière, et :

$$J_x^g = E^{-1} J_x^T (J_x E^{-1} J_x^T)^{-1} \quad (184)$$

Considérons à présent le cas où $x_k^i = O_k$ pour tout i . Alors, une deuxième approche consiste à chercher globalement à minimiser un critère:

$$c = \sum_{k=1}^n c_k = \sum_{k=1}^n (d_k^T d_k)^{-1} \quad (185)$$

Dans cette expression, $d_k^T d_k$ représente la distance à l'obstacle *virtuel* déjà évoqué. Alors :

$$\frac{dc_k}{dq} = \left(\sum_j \frac{dc_k}{dd_j} \frac{dd_j}{dq} \right) \quad (186)$$

Avec l'hypothèse :

$$\frac{dd_k}{dq} \approx (J_k^T)^T \quad (187)$$

il vient :

$$\nabla c = \sum_k (J_k^T)^T \frac{dc_k}{dx_k} = \sum_k (J_k^T)^T V_k \quad (188)$$

La méthode consiste alors, comme en annexe A2, paragraphe 2.2, à minimiser :

$$||\dot{q} + \lambda \nabla c||_q^2$$

sous la contrainte (180). D'où, pour la norme euclidienne:

$$\dot{q} = J_x^+ \dot{x} - P_x \lambda \sum_k (J_k^T)^T V_k \quad (189)$$

avec:

$$V_k = \sum_i \mu_k^i (y_k^i)^{3/2} u_k^i \quad (190)$$

Les approches (182) et (190) ne deviennent équivalentes qu'à deux conditions :

- 1- étendre (188) au torseur complet
- 2- choisir comme q-norme dans l'obtention de (189) celle associée à la métrique de matrice E^{-1} , ce qui revient d'une certaine façon à normaliser la minimisation ramenée en q .

3.4.2. Point de vue dynamique

Il est relativement simple : quelle que soit l'interprétation choisie, l'expression (172) associée à (168) reste la plus facile à mettre en oeuvre. L'utilisation de variables supplémentaires se heurte en effet dans ce cas à la difficulté de calcul de J_s : en choisissant comme variable :

$$s = R^T g = R^T \left(\sum_k J_k^T S_k \right) , \quad (191)$$

il apparaît en effet, compte tenu des approximations sous-jacentes, peu fiable de calculer J_s par une dérivation analytique de (191), dans laquelle certains termes sont mesurés à travers les y_k^i .

Par contre, s'il est possible d'exprimer assez finement le gradient du critère, comme cela peut se produire avec des capteurs télémétriques et des environnements géométriquement simples, le recours aux variables supplémentaires reste possible.

3.4.3. Conclusion

Bien qu'aucune analyse théorique de stabilité n'ait été proposée ici, on conçoit que l'approche présentée dans cette dernière partie ait en pratique de bonnes raisons de fonctionner dans les conditions dans lesquelles nous l'utiliserons, c'est à dire avec des capteurs adaptés, du type de ceux présentés en [10] : d'abord, les signaux capteurs décroissent très vite, et leur zone aveugle est relativement proche du point d'observation, ce qui annule rapidement les torseurs ; ensuite, on voit bien, intuitivement que, localement, la méthode tend effectivement à faire décroître les signaux, si les actions élémentaires sont de *direction fixe opposée* à la direction d'observation du capteur, seule leur amplitude dépendant de la sortie instantanée, laquelle diminue en fonction de la distance. Les simulations que nous allons présenter dans la suite vont confirmer ce point de vue.

3.5. Quelques résultats de simulation.

Les résultats présentés, extraits de [5] et [12], concernent une application réelle, partiellement décrite en [2], et ont été obtenus à l'aide d'un outil logiciel puissant de CAO 3D, décrit en [1] et [5].

Le système robotique utilisé dans ces simulations est constitué en fait d'un télémanipulateur MA 23 à six degrés de liberté, monté sur un porteur lourd à 4 articulations. Le site de travail comporte une cuve à laquelle est relié un tuyau coudé muni d'une bride. La hauteur de l'ensemble est d'environ 2, 50 m. Une

vue globale de la scène est donnée figure 6. Ce site constitue une partie du site réel de travail sur lequel sont effectuées les expérimentations canoniques du pole "Téléopération Avancée" du projet ARA ([2]).

Les capteurs simulés sont du type "proximètres optiques", et le modèle utilisé est un modèle photométrique, les interactions avec l'environnement étant calculées par une méthode de "Ray-casting" ([5]). La forme générale du modèle est (cf figure 5) :

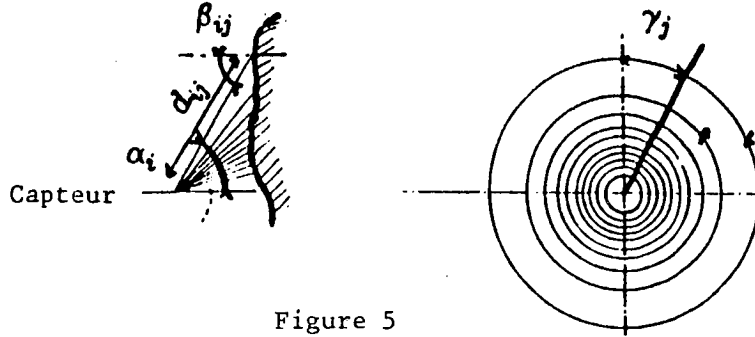


Figure 5

$$y = \sum_i f(\alpha_i) \sum_j \lambda_{ij} g(d_{ij}) h(\beta_{ij}) \quad (192)$$

Dans le cas des capteurs infra-rouge, les expressions suivantes sont utilisées :

$$f(\alpha_i) = \cos^2 \alpha_i \quad (193)$$

$$h(\beta_{ij}) = \cos^2 \beta_{ij} \quad (194)$$

$$g(d_{ij}) = \frac{1}{d_{ij}^2} \quad (195)$$

Les valeurs numériques associées sont compatibles avec les mesures expérimentales ; deux types de capteurs sont simulés :

capteurs à champ moyen :

$$\frac{1}{2}\alpha_i = (3, 6, 10, 15) \text{ deg.}$$

capteurs à champ large :

$$\frac{1}{2}\alpha_i = (5, 10, 15, 25, 35) \text{ deg.}$$

avec, dans les deux cas, $\gamma_j = 40 \text{ deg.}$

La tâche principale consiste à, partant de la configuration initiale présentée figure 6, à faire rejoindre par l'effecteur un point donné, l'orientation finale n'étant pas fixée. L'évitement des obstacles à l'aide des capteurs constitue la tâche secondaire, sauf pour l'effecteur, où les contributions capteurs sont intégrées à la tâche principale. 32 capteurs sont activés sur le robot, dont 13 sur l'effecteur, 13 sur l'avant bras, et 6 sur le bras. Leur disposition précise est donnée en [5]. Enfin, l'algorithme utilisé est (189), (190).

Des divers résultats présentés en [5] et [12], nous extrayons les illustrations suivantes :

FIGURES :

TRAJECTOIRES SANS CAPTEURS:

- 7: Exemple de trajectoire nominale de l'effecteur (vue de dessus)
 - 8: Trajectoire pseudo-inverse de l'avant-bras correspondant à la trajectoire nominale ci-dessus.
- (à noter que ces deux trajectoires intersectent les obstacles)

TRAJECTOIRES AVEC CAPTEURS

- 9: Position initiale de la pince avec interaction capteurs/cuve
- 10: Evolution de deux repères capteurs situés en extrémité de pince (vue de dessus), à comparer à 7
- 11: Evolution d'un repère capteur situé au milieu de l'avant-bras, dans l'évitement de la bride (vue de dessus)
- 12: Meme évolution en vue de face
- 13: Evolution de l'avant-bras (vue de face), à comparer à 8
- 14: Position quasi-finale ; l'objectif principal est atteint. Le minimum de l'objectif secondaire ne l'est pas tout à fait, et les segments vont continuer à s'éloigner des obstacles pendant quelques itérations.
- 15: Evolution du critère global $\sum ||V_k||^2$, les V_k étant les résultantes des torseurs mesurés par les capteurs. Le premier pic est dû au passage de la bride, le deuxième provient des capteurs latéraux de l'avant bras lorsque le tuyau vertical entre dans leur champ.
- 16: Meme critère, mais à une échelle rendant possible la comparaison avec la figure 20
- 17: Torseur référencé capteurs (composante en translation) de l'avant-bras. On distingue sur Y le pic dû à la bride, en Z le pic dû au tuyau vertical.
- 18: Somme du torseur référencé capteur et de la trajectoire nominale de l'effecteur.

COMPARAISON AVEC UNE METHODE DE POTENTIELS

- 19: Points soumis au potentiel d'évitement, généré à partir du modèle CSG 3D du système de simulation

- 20: Evolution du critère, calculé à partir des forces dérivées du potentiel
- 21: Evolution "filaire" du robot
- 22: Evolution de l'avant-bras
- 23: Evolution de la pince
- 24: Position finale

Enfin, les dernières figures, extraites de [9], présentent des résultats obtenus dans un contexte un peu différent. Le système considéré est plus simple, puisqu'il s'agit d'un robot plan à 3 rotations. La tâche principale est le parcours d'une trajectoire rectiligne à vitesse constante (cela symbolise par exemple une application de dépôt de produit), qui doit être réalisé en évitant deux obstacles cylindriques. Enfin, dans ces expériences, on suppose disposer d'une mesure instantanée de la distance entre chaque articulation et les cylindres-obstacles.

La simulation est effectuée dans un contexte dynamique, le modèle du robot utilisé dans la commande $(\dot{M}, \dot{N})$ pouvant inclure des erreurs, telle une charge inconnue. L'algorithme utilisé est du type (30), avec :

$$u_k = P_k^q (-K_v \dot{q} + \lambda \nabla c)$$

où c est un critère de la forme (185), calculé à partir des mesures de distance.

Nous extrayons de [9] les résultats suivants :

figure 25 : trajectoire dynamique pseudo-inverse ($u_k = 0$)

figures 26, 27, 28 : trajectoire dynamique avec évitement des obstacles.

D'autres résultats peuvent être trouvés en [9], avec en particulier le cas d'erreurs de modélisation mettant en évidence les phénomènes signalés en (34)/(37).

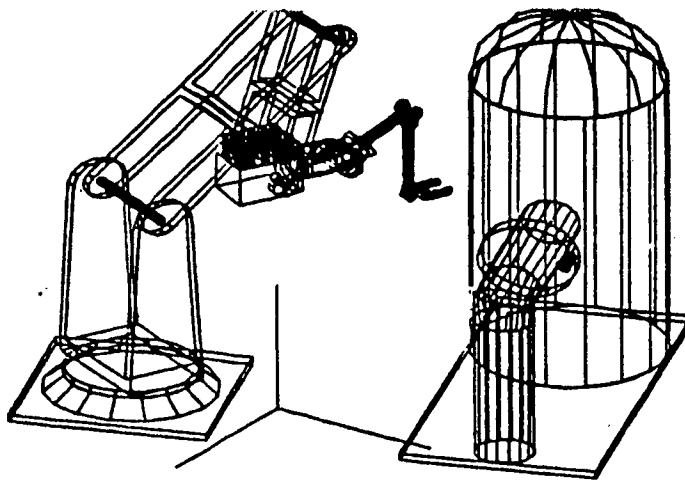


Figure 6

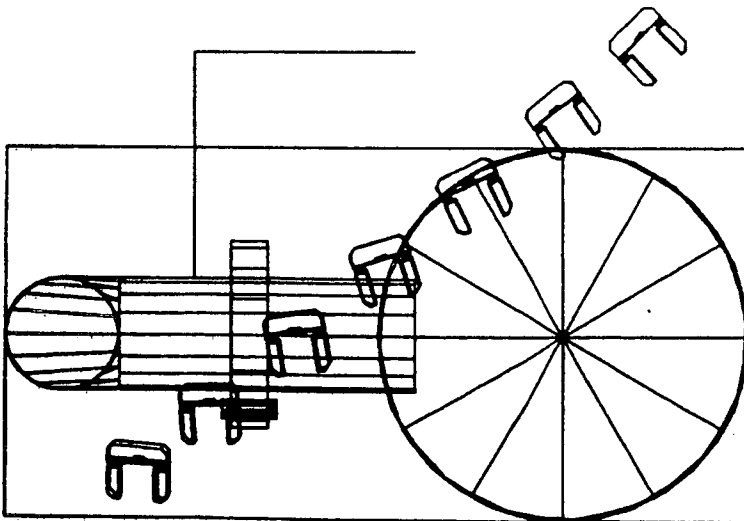


Figure 7

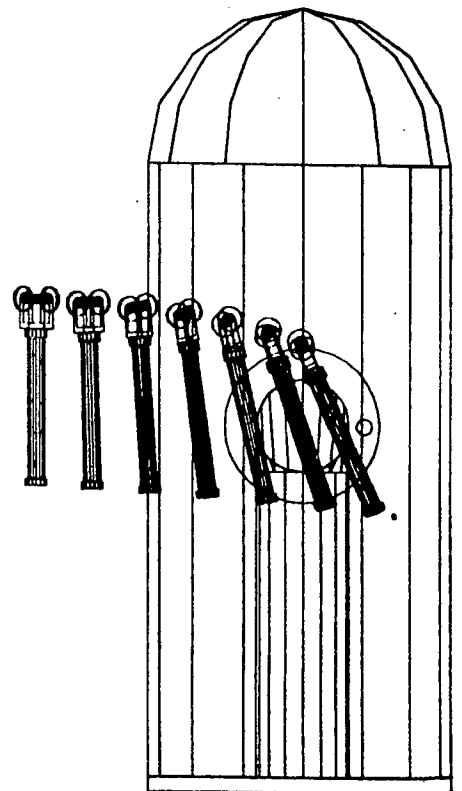


Figure 8

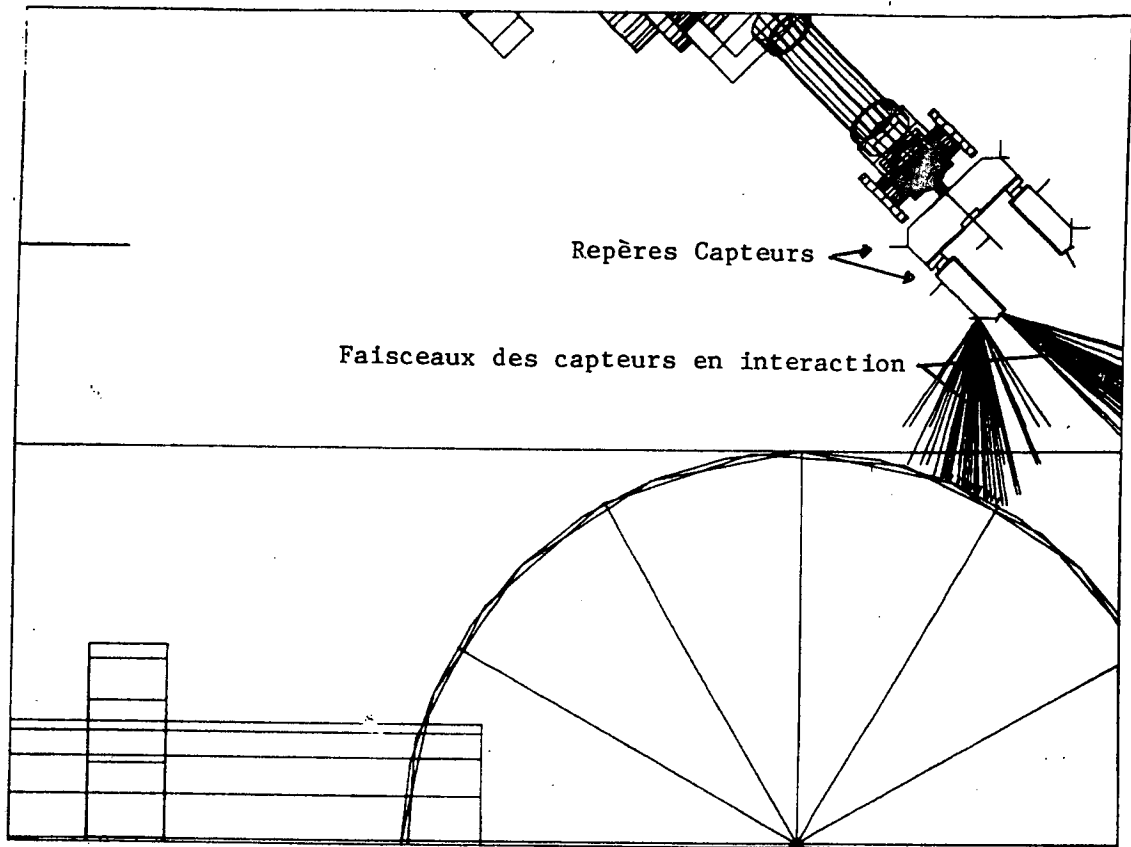


Figure 9

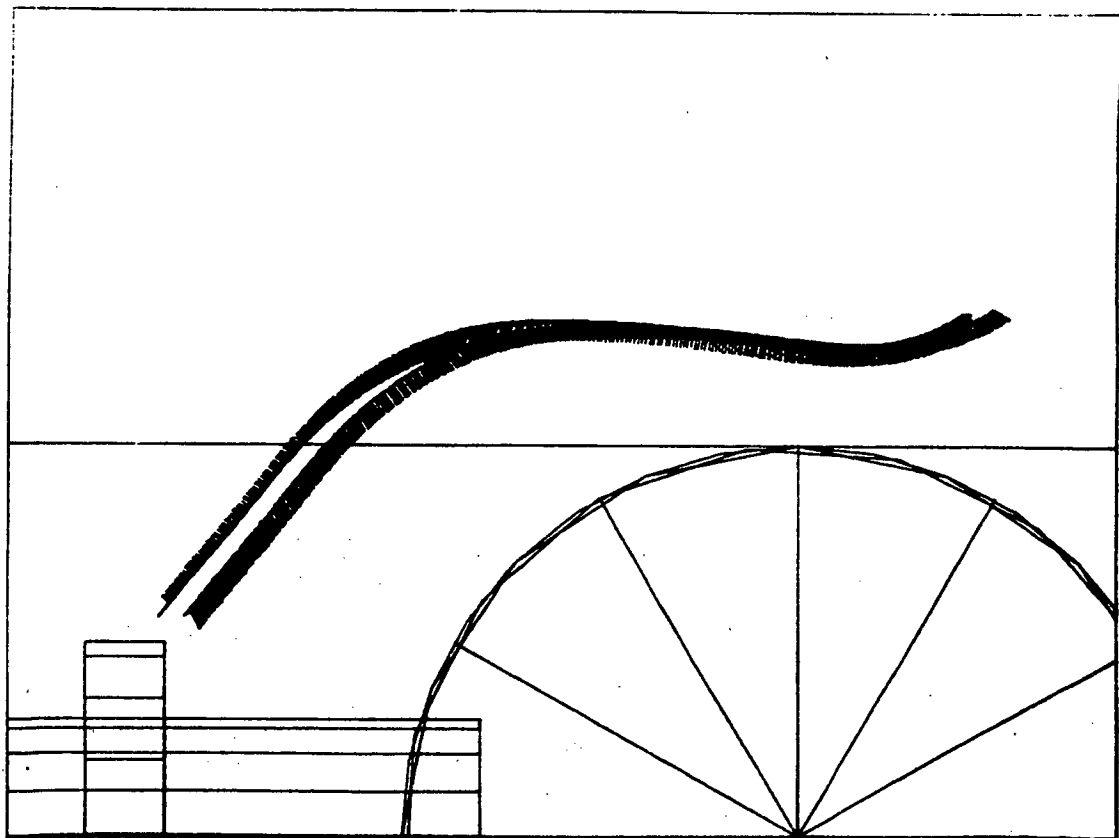


Figure 10

Figure 11

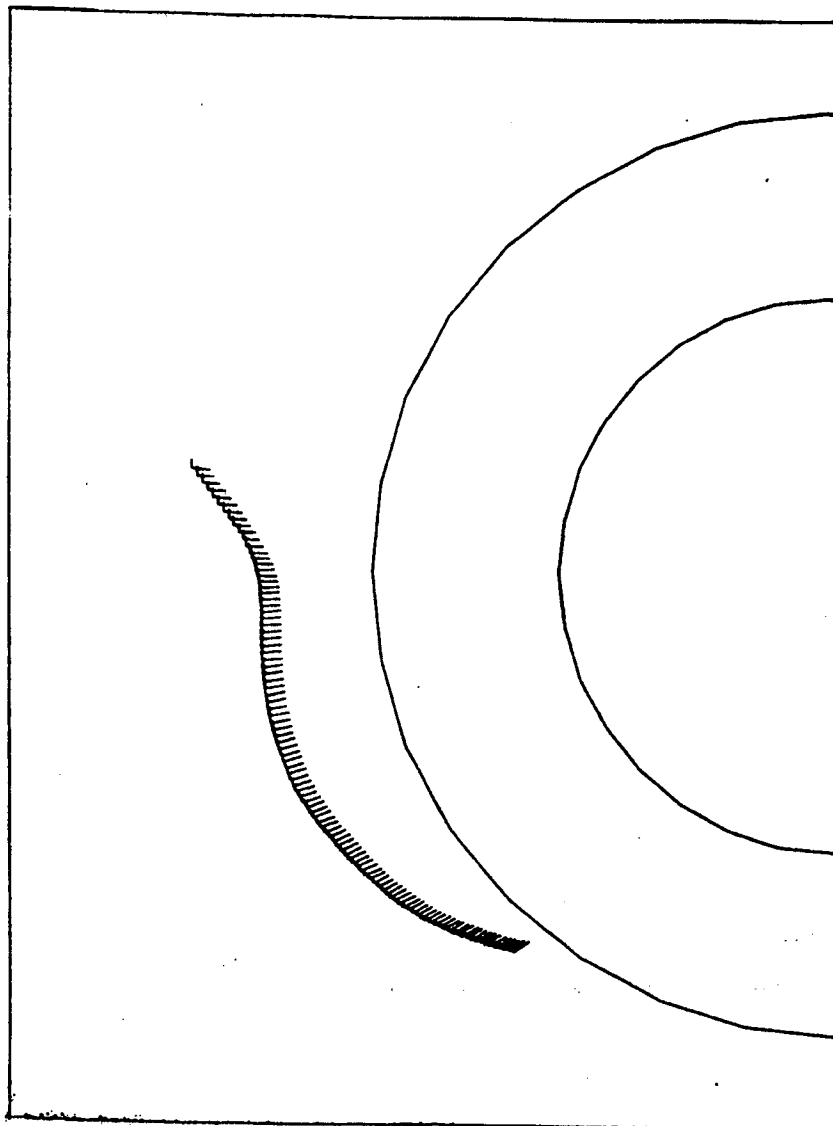
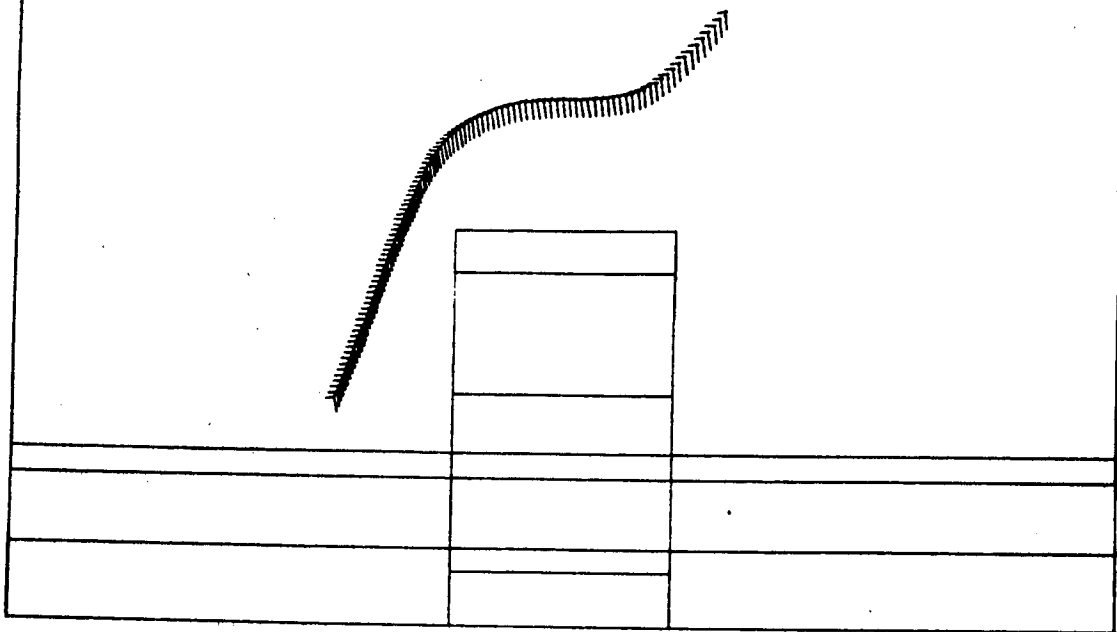


Figure 12

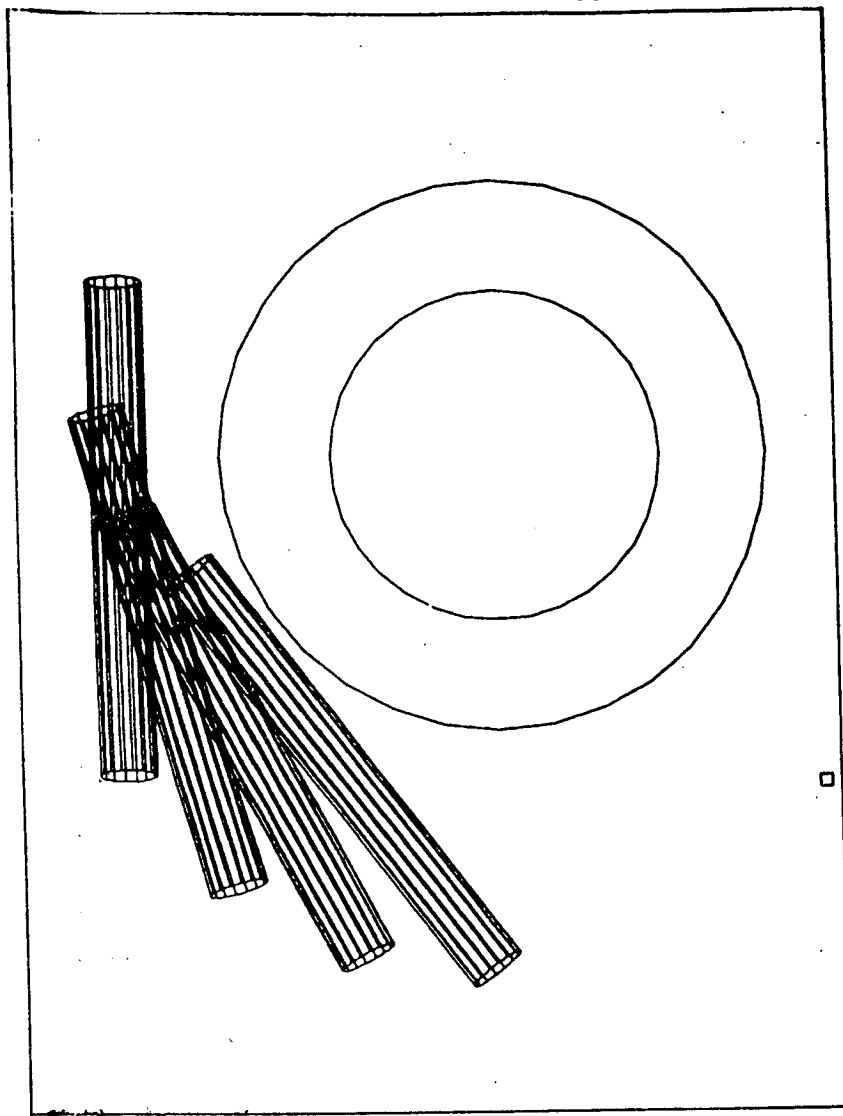


Figure 13

□ Point-cible

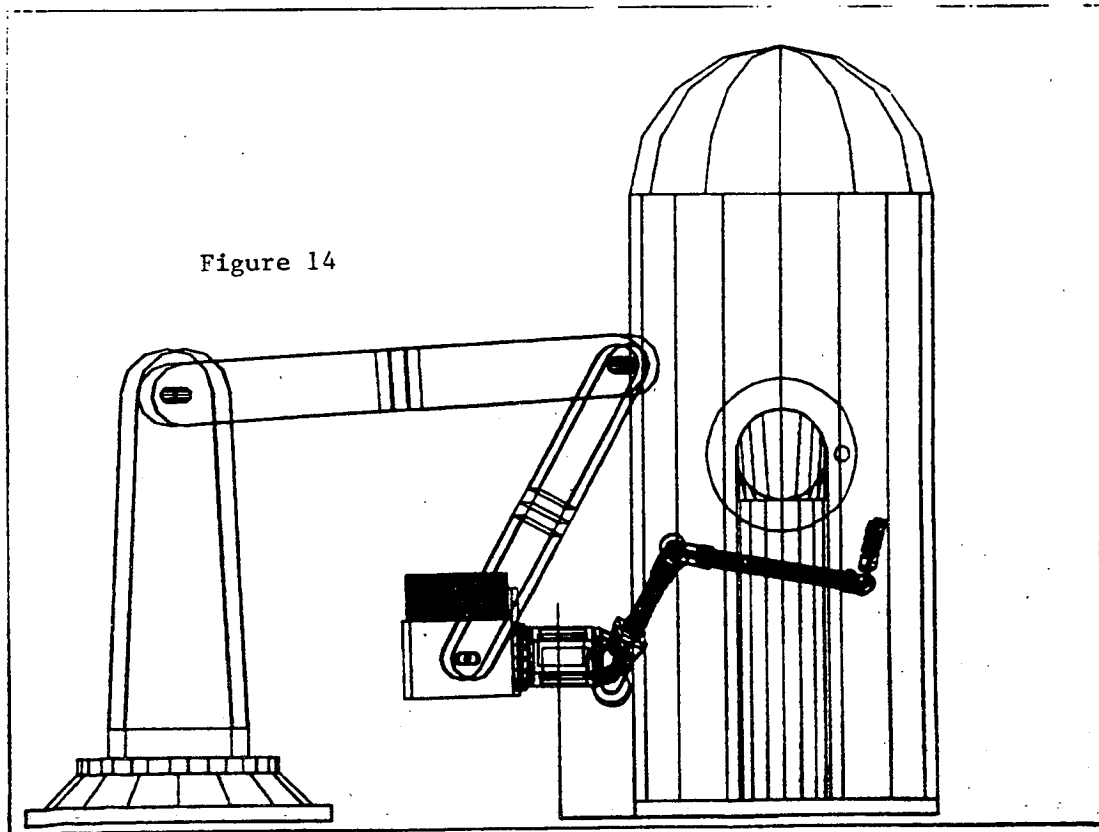


Figure 14

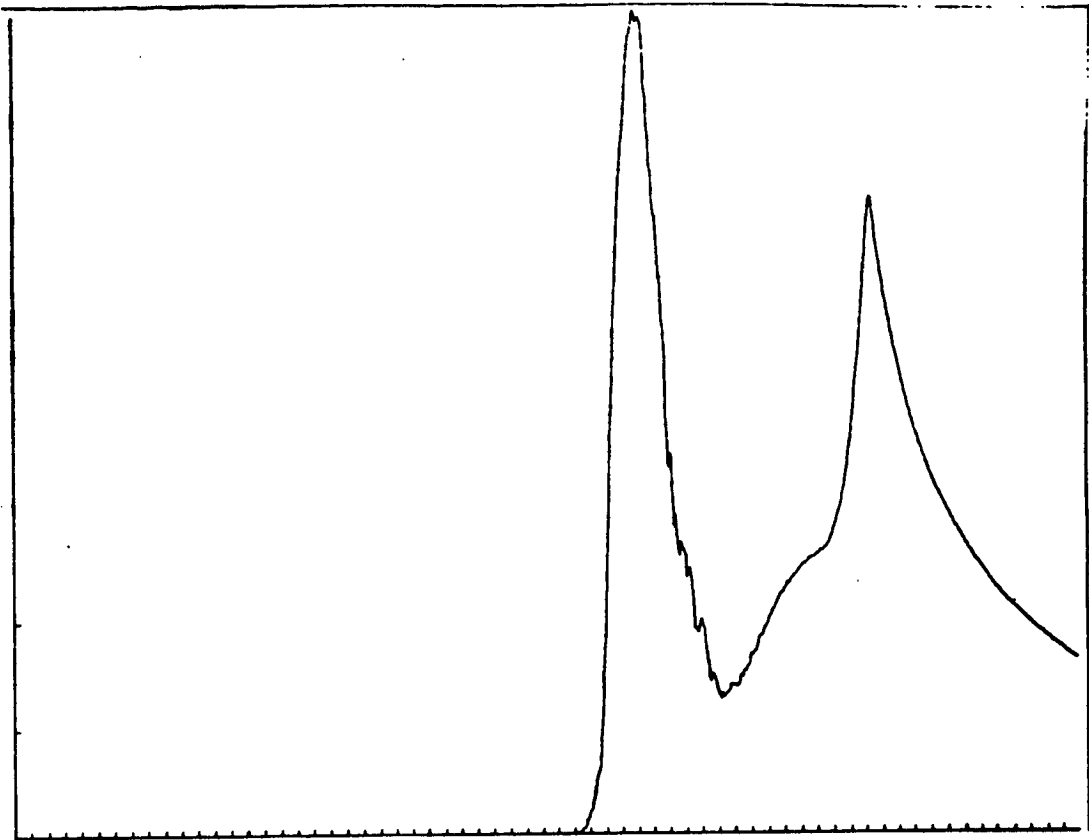


Figure 15

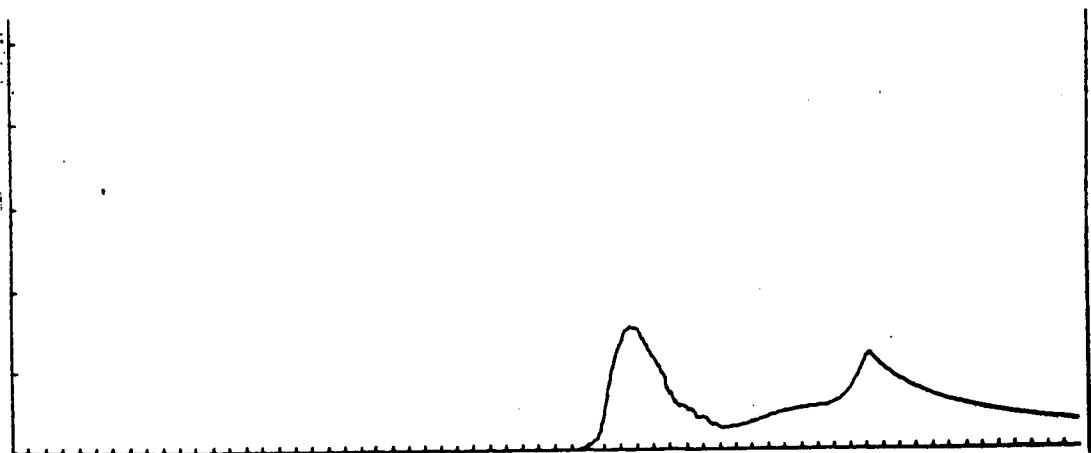


Figure 16

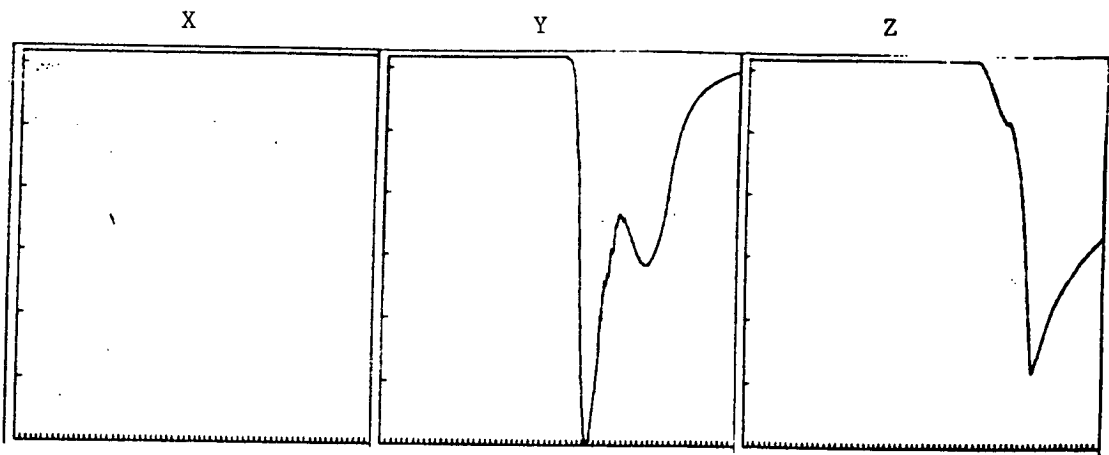


Figure 17

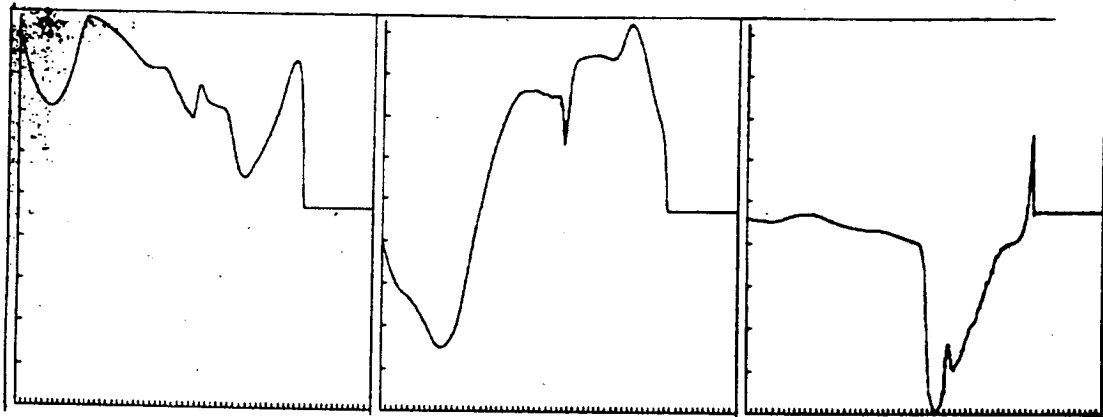


Figure 18

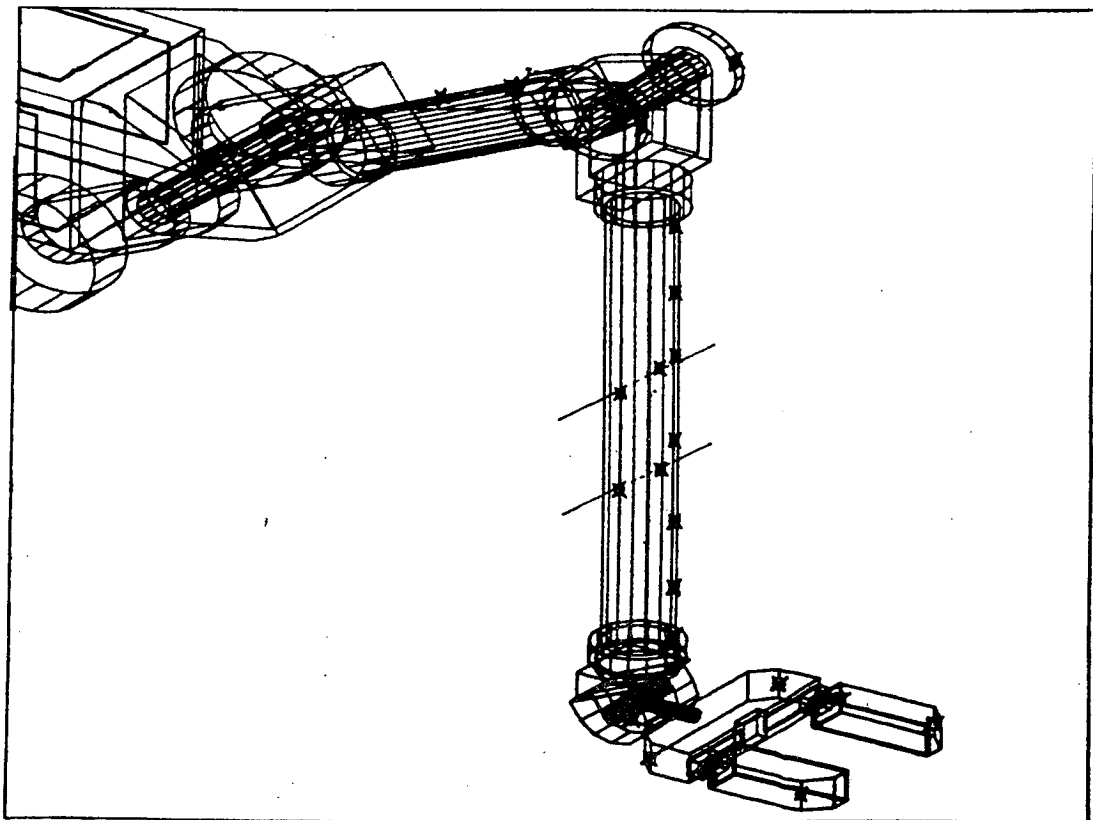


Figure 19

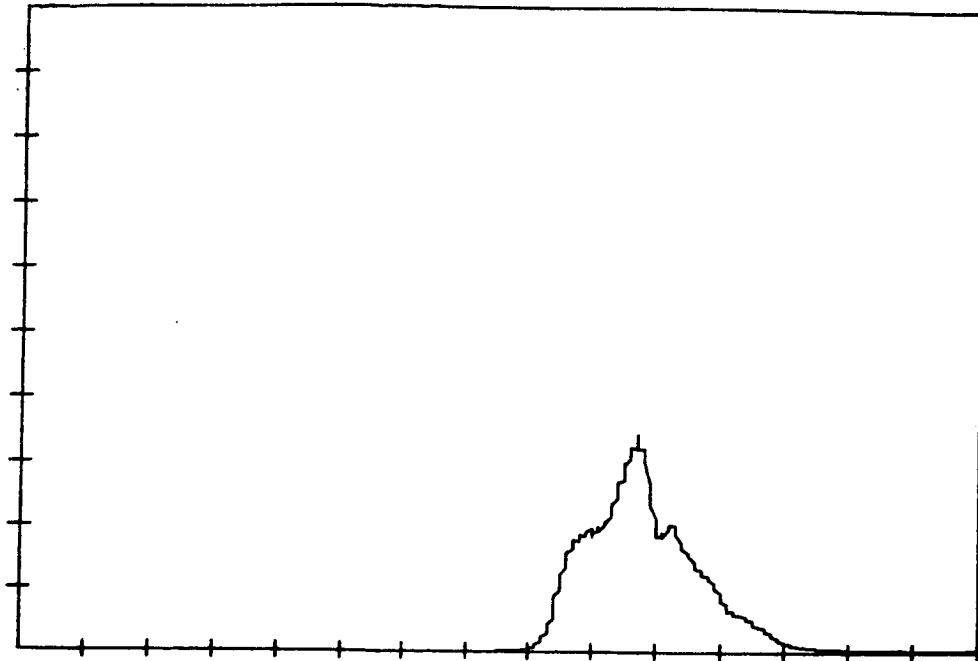


Figure 20

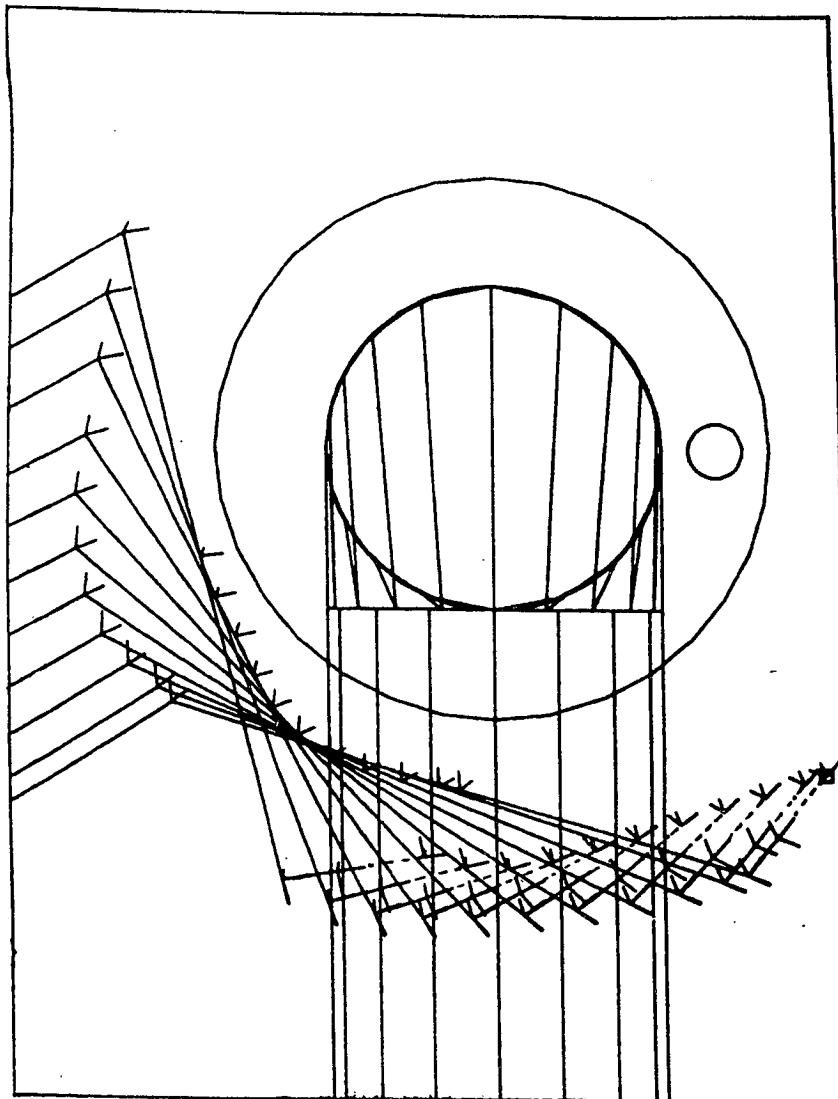


Figure 21

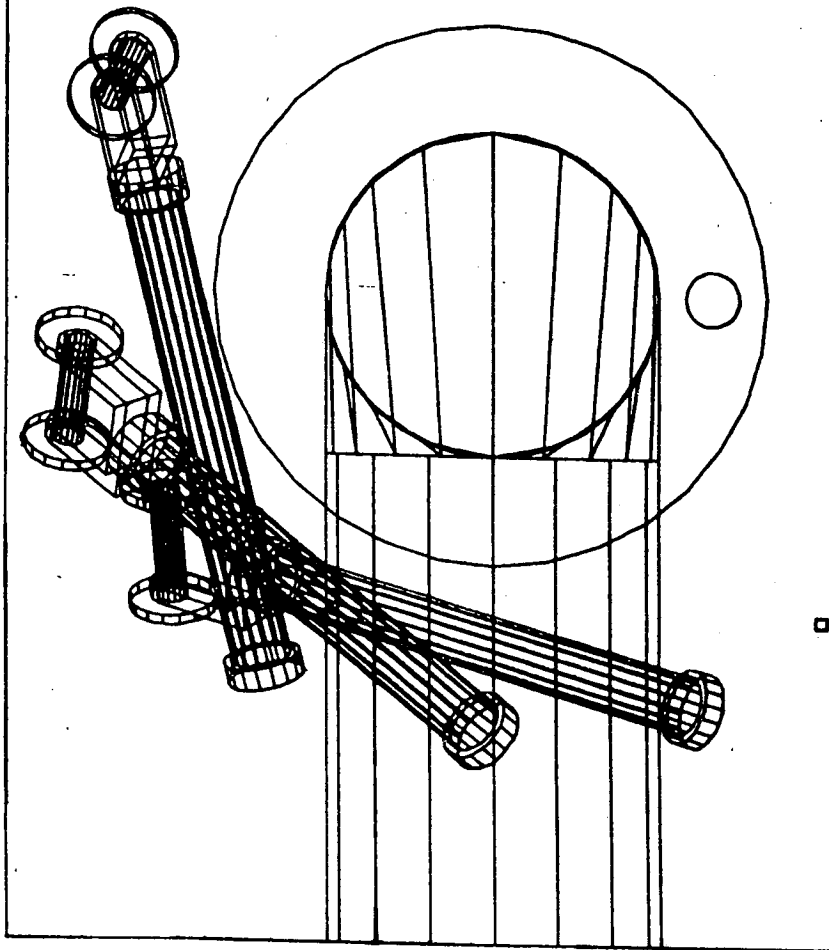


Figure 22

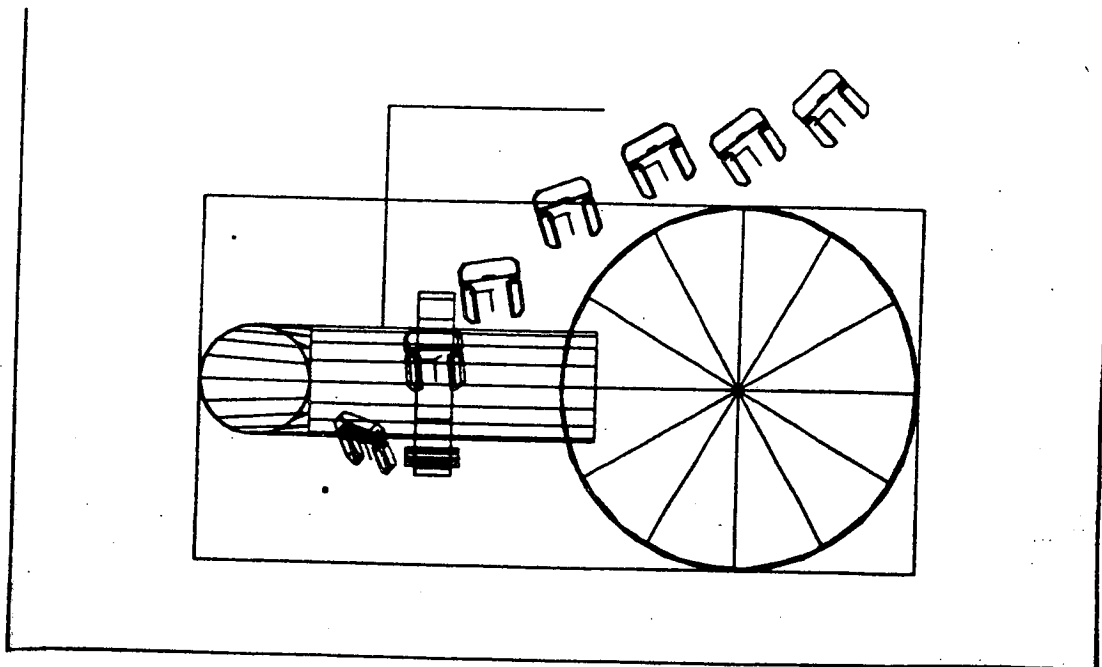


Figure 23.

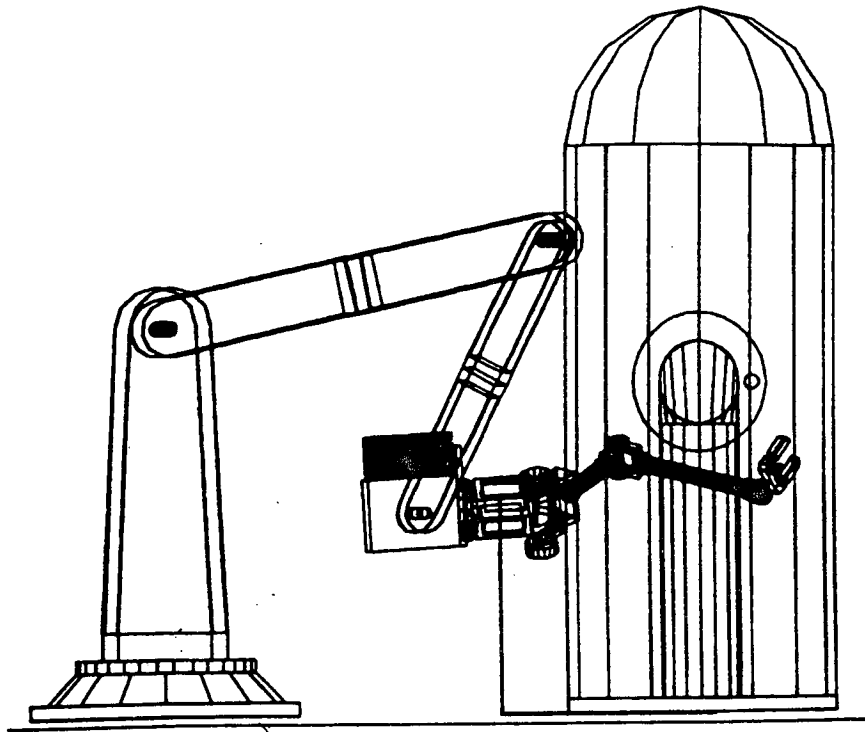


Figure 24

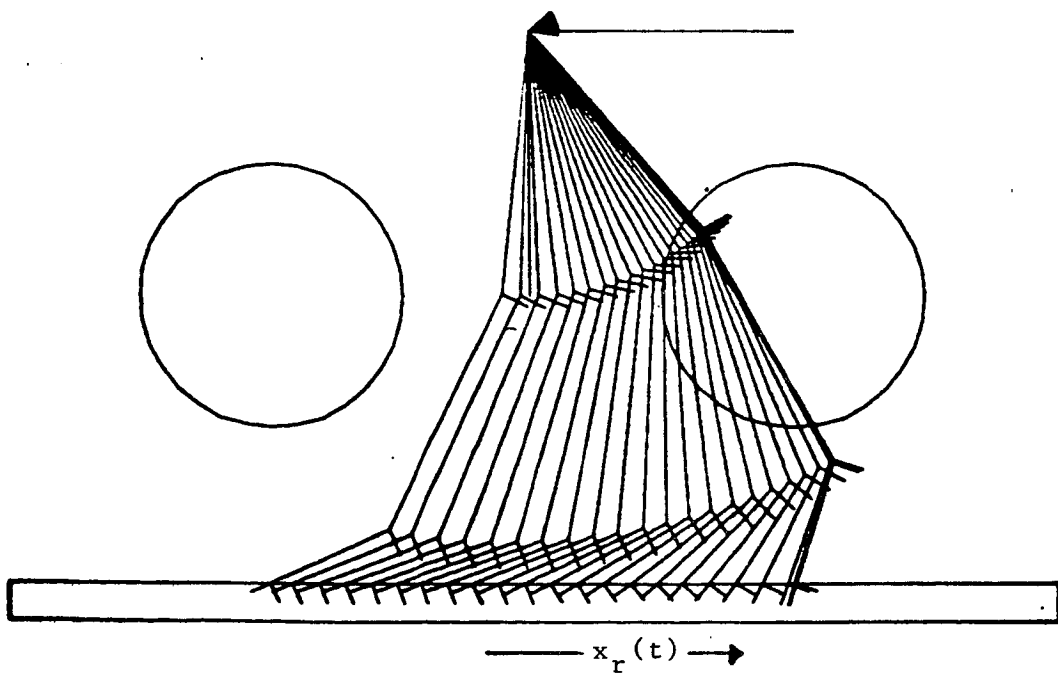


Figure 25

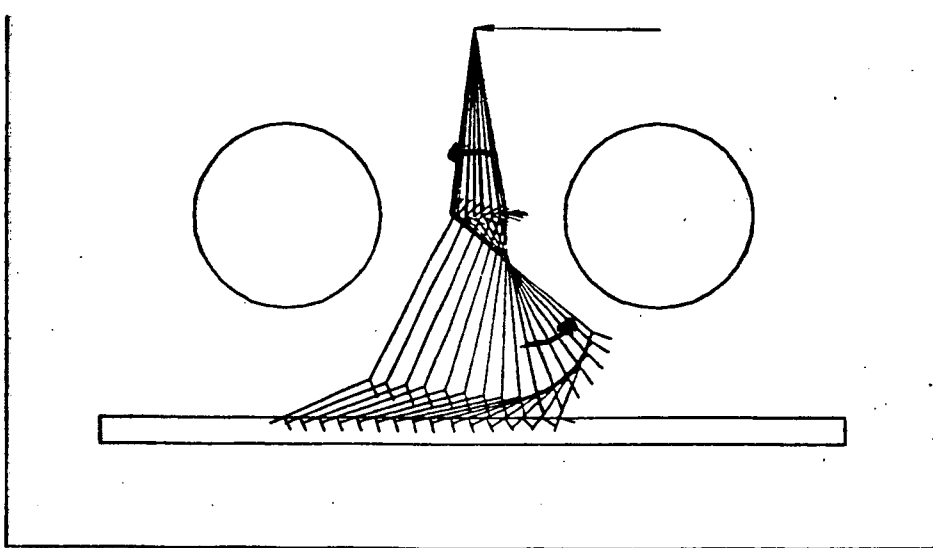


Figure 26

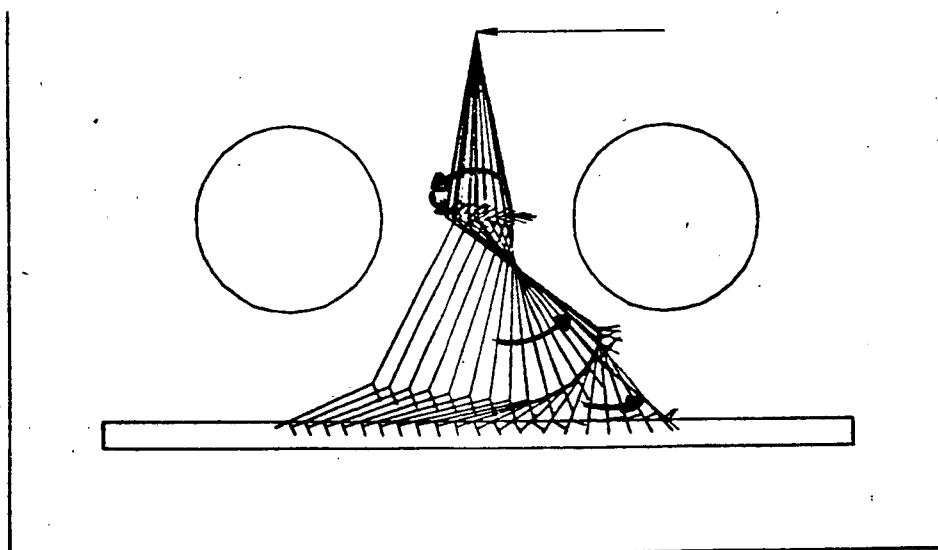


Figure 27

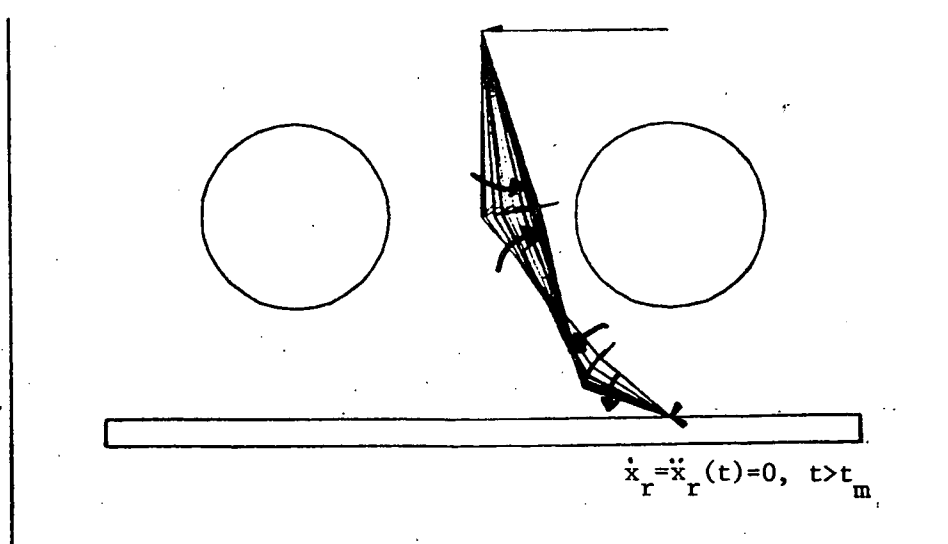


Figure 28

CONCLUSION

Nous avons présenté dans ce rapport une synthèse de la commande des systèmes redondants, en nous plaçant principalement dans un cadre dynamique d'une part, et d'autre part en proposant une méthode d'évitement d'obstacles basée sur l'utilisation de capteurs locaux.

Malgré les bons résultats de simulation présentés, il faut signaler quelques problèmes délicats, principalement de mise en oeuvre.

En premier lieu, et de façon évidente sur le plan pratique, il ne paraît pas réaliste de truffier un robot industriel de proximètres optiques tels ceux utilisés dans ce rapport. Il est donc nécessaire d'envisager l'utilisation de systèmes plus simples mais plus robustes (type tout-ou-rien), ou plus globaux, et de modifier les algorithmes en conséquence.

Par ailleurs, il faut noter que, bien que les derniers résultats présentés soient effectivement de caractère dynamique, ce qui est original, l'utilisation des variables supplémentaires dans ce contexte n'a pas été testée en simulation, pour la raison signalée dans le texte, c'est à dire la difficulté de calculer J_s à partir de mesures expérimentales approchées. Il reste donc à tester l'approche par variables supplémentaires dans un cas plus classique, par exemple l'évitement de singularités.

Enfin, l'analyse fine de la stabilité de l'approche proposée, en particulier du point de vue de la robustesse lorsque les modèles sont faux, ou lors de l'utilisation de capteurs locaux, reste à faire.

BIBLIOGRAPHIE

- [1] **André, G ; Boulic, R.** Système graphique et capteurs proximétriques pour la programmation de robots. Proceedings of 4th European Conference on CAD/CAM and computer graphics, Paris 1985, Hermes editor.
- [2] **André, G ; Fournier, R.** Generalized end effector control in a computer aided teleoperation system with application to motion coordination of a manipulator arm on a oscillating carrier. ICAR 85, sept. 1985, Tokyo.
- [3] **Baillieul, J ; Hollerbach, J ; Brockett, R.** Programming and control of kinematically redundant manipulators, Proceedings of 23th Conference on decision and control, Las Vegas, dec. 1984
- [4] **Ben-Israel, A ; Greville, T.N.E.** Generalized inverses :theory and applications , Wiley, N.Y, 1974
- [5] **Boulic, R.** CAO de boucles de commande avec capteurs en robotique et en téléopération. Thèse, Université de Rennes, à paraître, février 1985
- [6] **Boullion, T.L ; Odell, P.L.** Generalized inverses matrices, Wiley editor, 1971
- [7] **Campbell, S.L ; Meyer, C.D.** Generalized inverses of linear transformations, Pitman publishers inc. , 1979.
- [8] **Dieudonné, J ; Huet, D.** Eléments d'analyse, tome 1, Gauthier-Villars, 1968
- [9] **Dumas, R.** Simulation et évaluation de commandes dynamiques robustes, Thèse, à paraître, Université de Rennes, février 1985
- [10] **Espiau, B.** Use of optical reflectance sensors, from Recent advances in robotics, Wiley editor, 1985.
- [11] **Espiau, B.** Closed loop control of robots with local environment sensing: principles and applications. 2nd International Symposium on Robotics Research (ISRR), Kyoto, August 1984.
- [12] **Espiau, B. ; Boulic, R.** Collision avoidance for redundant robots with proximity sensors, Third ISRR, Gouvieux, France, octobre 1985.
- [13] **Faverjon, B.** Obstacle avoidance using an octree in the configuration space of a manipulator, IEEE Conference on Robotics, Atlanta, march 1984.
- [14] **Fournier, A.** Génération de mouvements en robotique ; applications des inverses généralisées et des pseudo- inverses, Thèse d'état, Université de Montpellier, avril 1980
- [15] **Freund, E ; Hoyer, H.** Collision avoidance for obstacles and mobile robots in multi-robots systems, Third ISRR, Gouvieux, France , octobre 1985
- [16] **Freund, E ; Hoyer, H.** Collision avoidance in multi-robot systems, 2nd ISRR, Kyoto, august 1984.
- [17] **Gabay, D; Luenberger, D.G.** Efficiently converging minimization methods based on the reduced gradient, SIAM journal of control and optimization, Vol 14 no 1, janvier 1976

- [18] **Hogan N.** Programmable impedance control of industrial manipulators, Conference on CAD/CAM technology, MIT, Cambridge, 1981
- [19] **Hollerbach, J ; K.C. Suh.** Redundancy resolution of manipulators through torque optimization, IEEE conference on robotics and automation, Saint Louis, Mars 1985.
- [20] **Khatib, O.** Commande dynamique dans l'espace operationnel des robots manipulateurs en presence d'obstacles, Thesis, Toulouse, 1980.
- [21] **Khatib, O.** The operationnal space formulation in the analysis , design and control of robot manipulators, third ISRR, Gouvieux, France, Octobre 1985
- [22] **Khatib, O.** Real time obstacle avoidance for manipulators and mobile robots, IEEE conference on robotics and automation, Saint Louis, mars 1985
- [23] **Kircanski, M; Vukobratovic, M.** Trajectory planning for redundant manipulators in the presence of obstacles, 5th CISM-IFTOMM Symposium "Romansy", Udine , june 1984
- [24] **Klein, C.A.** Use of redundancy in the design of robtic systems, 2 nd ISRR, Kyoto, august 1984.
- [25] **Klein, C.A ; Ching-Hsiang Huang** Review of pseudo-inverse control for use with kinematically redundant manipulators, IEEE trans. on SMC, vol. SMC-13 no 3, 1983.
- [26] **Klema, V.C. ; Laub, A.J.** The singular value decomposition: its computation and some applications, IEEE trans. on AC, vol. AC-25 no 2, 1980.
- [27] **Krogh, B. H.** A generalized potential field approach to obstacle avoidance control, Conference: robotics research, the next five years and beyond, Bethlehem, Penns., august 1984
- [28] **Leborgne, M.** Modélisation des robots manipulateurs rigides, PI 248, IRISA, Rennes, février 1985.
- [29] **Liegeois, A.** Automatic supervisory control of the configuration and behaviour of multibody mechanisms, IEEE trans. on SMC, vol. SMC-7 no 12, 1977.
- [30] **Maciejewski, A.A** Obstacle avoidance for kinematically redundant manipulators, M.S. thesis, The Ohio State University, Dept of engineering, Columbus, 1984.
- [31] **Nakamura, Y. ; Hanafusa, H.** Task priority based redundancy control of robot manipulators, 2nd ISRR, Kyoto, august 1984.
- [32] **Samson, C.** Commande non-linéaire robuste des robots manipulateurs, PI 184, IRISA , Rennes, janvier 1983
- [33] **Yoshikawa, T.** Analysis and design of articulated robot arms from the viewpoint of dynamic manipulability, third ISRR, Gouvieux, France, octobre 1985
- [34] **Yoshikawa, T.** Manipulability of Robotics Mechanisms, International Journal of Robotics Research, Vol 4 no 2, Summer 1985
- [35] **Yoshikawa, T.** Analysis and control of robots manipulators with redundancy, 1st ISRR, Bretton Woods, NH, august 1983.

ANNEXE 1

INVERSES GENERALISEES

Cette annexe donne les définitions des inverses généralisées et en rappelle les propriétés de base. Elle s'inspire des références [6] et [7].

Définition 1 :

A^g est une inverse généralisée d'une matrice $A \in \mathbb{C}^{m \times n}$ si et seulement si :

$$A A^g A = A \quad (A1-1)$$

Cette inverse généralisée existe toujours, car, pour tout A , il existe toujours deux matrices P et Q non singulières telles que :

$$P A Q = \begin{bmatrix} I_\rho & 0 \\ 0 & 0 \end{bmatrix} \quad (A1-2)$$

où ρ est le rang de A . Alors :

$$A^g = Q \begin{bmatrix} I_\rho & U \\ V & W \end{bmatrix} P \text{ pour tout } U, V, W \quad (A1-3)$$

En outre, ce qu'il est convenu d'appeler une *pseudo-inverse* A^+ satisfait les relations suivantes :

$$A^+ A A^+ = A^+ \quad (A1-4)$$

$$(A^+ A)^* = A^+ A \quad (A1-5)$$

$$(A A^+)^* = A A^+ \quad (A1-6)$$

où A^* dénote la transposée conjuguée de A .

Soit :

- A^r une matrice satisfaisant (A1-1) et (A1-4) (appelée "inverse généralisée réflexive")

- A^w une matrice satisfaisant (A1-1), (A1-4), et (A1-5)

- A^n une matrice satisfaisant (A1-1), (A1-4), et (A1-6);

On montre alors les propriétés suivantes :

$$A^r = A^{g_1} A A^{g_2} \quad (A1-7)$$

où A^{g_1} et A^{g_2} sont deux inverses généralisées de A .

$$A^w = A^* (A A^*)^r \quad (A1-8)$$

$$A^n = (A^* A)^r A^* \quad (A1-9)$$

$$A^+ = A^w A A^n \quad (A1-10)$$

A^+ existe donc toujours. Elle correspond à la solution minimale $\hat{x}$ de :

$$A x = y \quad (A1-11)$$

au sens des normes p - et q :

$$||A\hat{x}-y||_p \leq ||Ax-y||_p \quad \forall x \quad (A1-12)$$

$$||\hat{x}||_q \leq ||\tilde{x}||_q \quad (A1-13)$$

$\forall \tilde{x}$ solution de (A1-11) au sens des moindres carrés. Pour un choix de normes donné, A^+ est unique.

Plaçons - nous à présent dans $\mathbb{R}^{m \times n}$ qui sera notre espace de travail dans la suite.

Les propriétés principales de A^+ sont les suivantes:

$$(AB)^+ = B^+ A^+ \quad (A1-14)$$

$$\rho = \text{rang } A = \text{rang } A^+ = \text{rang } (A^+ A) = \text{trace } (A^+ A)$$

$$(A^+)^+ = A \quad (A1-15)$$

En outre, lorsque A est de rang plein, et si la norme est euclidienne, pour les systèmes de classe 3, nous avons :

$$A_3^+ = (A^T A)^{-1} A^T \quad (A1-16)$$

et pour ceux de classe 2 :

$$A_2^+ = A^T (A A^T)^{-1} \quad (A1-17)$$

On rencontre également souvent des versions pondérées de ces expressions: soit W une matrice symétrique définie positive de dimension convenable; alors:

$$A_3^w = (A^T W A)^{-1} A^T W$$

$$A_2^w = W^{-1} A^T (A W^{-1} A^T)^{-1}$$

Les pseudo-inverses sont très liées aux décompositions en valeurs singulières: toute matrice $A \in \mathbb{R}^{m \times n}$, de rang ρ , peut en effet se mettre sous la forme ([26]):

$$A = U \Sigma V^T \quad (A1-18)$$

où $U \in \mathbb{R}^{m \times m}$ et $V \in \mathbb{R}^{n \times n}$ sont des matrices orthogonales et où:

$$\Sigma = \begin{pmatrix} S & 0 \\ 0 & 0 \end{pmatrix} \quad (A1-19)$$

$$S = \text{Diag}(\sigma_1, \dots, \sigma_\rho)$$

$$\sigma_1 \geq \dots \geq \sigma_\rho > 0$$

Les σ_i sont les valeurs singulières de A , et sont aussi les racines carrées des valeurs propres non nulles de $A^T A$.

Nous avons alors :

$$U^T A V = \Sigma \quad (A1-20)$$

et:

$$A^+ = V \Sigma^+ U^T \quad (A1-21)$$

avec:

$$\Sigma^+ = \begin{pmatrix} S^{-1} & 0 \\ 0 & 0 \end{pmatrix} \quad (A1-22)$$

NOTE

Il s'agit là d'un cas particulier de l'expression générale : pour tout A de rang r , il existe P, Q régulières telles que :

$$PAQ = \begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix} \quad (A1-23)$$

B étant une matrice $r \times r$ non singulière. Soit :

$$Q^{-1}A'P^{-1} = \begin{pmatrix} Z & U' \\ V' & W \end{pmatrix} \quad (A1-24)$$

On a alors $A' = A^g$ ssi $Z = B^{-1}$. En partitionnant :

$$P = \begin{pmatrix} P_1 \\ P_2 \end{pmatrix}, \quad Q = (Q_1 \ Q_2)$$

on montre aisément que $A' = A^+$ ssi :

$$\begin{cases} Z = B^{-1} \\ U' = -B^{-1}P_1P_2^+ \\ V' = -Q_2^+Q_1B^{-1} \\ W = Q_2^+Q_1B^{-1}P_1P_2^+ \end{cases} \quad (A1-25)$$

On vérifie que le choix : $P = U^T, Q = V$ donne bien $B = S, U' = V' = W = 0$

Si l'on considère maintenant les divers sous-espaces associés à l'application de matrice A , nous avons :

$$A : E^n \rightarrow E^m \quad m < n$$

$$\text{Im}(A) \oplus \text{Ker}(AA^g) = E^m$$

$$\text{Ker}(A) \oplus \text{Im}(A^gA) = E^n$$

Plus précisément, si $A^g = A^+$, alors :

$\text{Im}(A^+)$ est le complémentaire orthogonal de $\text{Ker}(A)$,

$\text{Ker}(A^+)$ est le complémentaire orthogonal de $\text{Im}(A)$.

Les projections orthogonales associées sont :

$$AA^+ = \text{Proj. sur Im}(A) = U_1U_1^T$$

$$A^+A = \text{Proj. sur Im}(A^T) = \text{Ker}(A) = V_1V_1^T$$

$$I - AA^+ = \text{Proj. sur Ker}(A^T) = U_2U_2^T$$

$$I - A^+A = \text{Proj. sur Ker}(A) = V_2V_2^T$$

$$\text{Où } U = (U_1 \ U_2), \quad V = \begin{pmatrix} V_1 \\ V_2 \end{pmatrix}$$

De plus, pour les systèmes de classe 2, nous avons :

$$AA^+ = I \quad (A1-26)$$

et pour ceux de classe 3 :

$$Q = Q' = \text{Im}(A^T) \Rightarrow A^+A = I \quad (A1-27)$$

Soit P le projecteur sur le noyau de A (opérateur idempotent, c'est-à-dire : $P^2 = P$):

$$P = I - A^+ A \quad (A1-28)$$

Enfin, comme

$$E^n = \text{lin}(A^+) \quad (29)$$

on peut écrire :

pour tout $v \in E^n$ tel que $u \in E^m = Av$, alors :

$$u = A^+ u + P u, \quad \forall u \quad (A1-30)$$

avec les propriétés évidentes:

$$PA^+ = AP = 0 \quad (A1-31)$$

NOTE

Certaines des propriétés de A^+ sont également applicables à A^T , par exemple (A1-28) à (A1-31).

Enfin, en notant $\dot{M} = \frac{dM}{dt}$ nous pouvons établir les relations suivantes, dans le cas (A1-17):

$$P\dot{A}^+ = -\dot{P}A^+ \quad (A1-32)$$

$$A\dot{A}^+ = -\dot{A}A^+ \quad (A1-33)$$

$$(\dot{A}^+) = -P\dot{A}^T(AA^T)^{-1} - A^T\dot{A}A^+ \quad (A1-34)$$

$$A\dot{P} = -\dot{A}P \quad (A1-35)$$

$$P\dot{P}P = 0 \quad (A1-36)$$

$$\dot{P} = [P(A\dot{A})^T + (A^+\dot{A})P] \quad (A1-37)$$

$$(\dot{A}^+)A = -(\dot{P} + A^+\dot{A}) \quad (A1-38)$$

ANNEXE 2

CALCUL DES INVERSES GENERALISEES

1. FORME GENERALE

Les inverses généralisées et pseudo-inverses présentées dans la littérature peuvent être interprétées comme autant de cas particuliers de la solution générale d'un problème de la forme:

$$\text{minimiser } ||AY - b||^2 \quad (\text{A2-1})$$

sous la contrainte:

$$CY = f \quad (\text{A2-2})$$

avec:

$$A \in \mathbb{R}^{p \times n} ; b \in \mathbb{R}^p$$

et:

$$C \in \mathbb{R}^{m \times n} ; f \in \mathbb{R}^m ; m < n$$

Cette solution, lorsqu'elle est unique, est donnée par ([7]):

$$Y^* = A_c^+ b + (I - A_c^+ A) C^+ f \quad (\text{A2-3})$$

où:

$$A_c^+ = (A(I - C^+ C))^+ \quad (\text{A2-4})$$

De plus, l'ensemble des solutions possibles (si par exemple A n'est pas de rang plein) est donné par :

$$Y^* = A_c^+ b + (I - A_c^+ A) (C^+ f + z) \quad (\text{A2-5})$$

pour tout $z \in \text{Ker } C$, ou pour tout $z' \in \mathbb{R}^n$ et :

$$z = (I - C^+ C) z'$$

Revenons au problème des robots redondants et considérons alors le cas général suivant:

minimiser:

$$p(q, \dot{q}) = \frac{1}{2} (y - X\dot{q})^T W (y - X\dot{q}) \quad (\text{A2-5})$$

sous la contrainte :

$$T = U\dot{q} \quad (\text{A2-6})$$

avec:

$$X \in \mathbb{R}^{p \times n} ; y \in \mathbb{R}^p ; p \geq n$$

et:

$$U \in \mathbb{R}^{r \times n} ; T \in \mathbb{R}^r$$

W est une matrice de pondération $n \times n$ symétrique définie positive, et X et U

sont supposées de rang plein.

En annulant le gradient du Lagrangien:

$$\Lambda = p(q, \dot{q}) + \lambda^T (T - U\dot{q}) \quad (\text{A2-7})$$

il vient:

$$\dot{q}^* = U_g T + (I - U_g U) X^g y \quad (\text{A2-8})$$

où:

$$U_g = B^{-1} U^T (U B^{-1} U^T)^{-1} \quad (\text{A2-9})$$

(pseudo-inverse de U pour la métrique B)

$$B = X^T W X \quad (\text{A2-10})$$

$$X^g = B^{-1} X^T W \quad (\text{A2-11})$$

(solution minimale au sens des moindres carrés pour $c(q, \dot{q})$)

2. FORMES PARTICULIERES

2.1. Pseudo-inverse de Moore-Penrose

C'est la version la plus simple ([3]) ; avec le choix :

$$\begin{cases} y = 0 ; X = W = I \\ U = J_x ; T = \dot{x} \end{cases} \quad (\text{A2-12})$$

on obtient alors (39) et :

$$\dot{q}^* = J_x^+ \dot{x} \quad (\text{A2-13})$$

Cette approche, qui minimise $\|\dot{q}\|^2$ est utilisée en [25].

2.2. Pseudo-inverse "étendue"

Le deuxième terme de (A2-8) n'est dans ce cas pas nul, car on choisit alors de réaliser le mieux possible l'algorithme de gradient:

$$\dot{q} = -\lambda \frac{dc}{dq} = -\lambda g \quad (\text{A2-14})$$

Alors:

$$\begin{cases} y = -\lambda g ; X = W = I \\ U = J_x ; T = \dot{x} \end{cases} \quad (\text{A2-15})$$

et l'on obtient (38).

C'est la méthode proposée originellement par Liégeois et utilisée par de nombreux auteurs.

2.3. Pseudo-inverses pondérées.

Dans les deux algorithmes précédents, on peut introduire une matrice W non identité. Ainsi, Kircanski ([23]) utilise une approche du type précédent, mais avec cette fois:

$$W = \Delta$$

ou:

$$W = J_k^T \Delta J_k$$

Δ étant une matrice diagonale dépendant de la distance entre un point donné du k ème corps C_k et un obstacle. Hollerbach ([19]), et d'autres auteurs cités par lui-même, utilisent également des pseudo-inverses pondérées.

Un cas particulier important est mentionné, entre autres, par Khatib ([21]): $W = M(q)$, matrice d'inertie. Alors, $p(q, \dot{q})$ est l'énergie cinétique totale du manipulateur, et l'on obtient l'équation (31).

2.4.

Choix d'une vitesse intermédiaire généralisée

C'est l'approche de Klein ([24]) et Maciejewski ([30]). L'algorithme proposé consiste en effet à choisir d'imposer en un point donné x_k d'un corps donné C_k du robot une vitesse d'éloignement désirée T_k . Alors, (A2-1) et (A2-2) s'appliquent en choisissant:

$$\begin{cases} C = J_x ; A = J_k \\ Y = \dot{q} ; b = T_k ; f = \dot{x} \end{cases} \quad (A2-16)$$

Remarquons que, suivant les valeurs de m, n, p , la dimension de l'espace des solutions peut varier. Lorsque T_k et $\dot{x}$ sont tous deux *parfaitement* réalisables de façon *unique*, on retrouve les expressions du paragraphe 2.2 du rapport, avec : $J_s = J_k$.

2.5. Choix de plusieurs vitesses intermédiaires

C'est une extension du cas précédent obtenue en prenant : $W = I$, et:

$$\begin{cases} Y^T = (S_{k_1}^T, \dots, S_{k_r}^T) \\ X^T = (J_{k_1}^T, \dots, J_{k_r}^T) \end{cases} \quad (A2-17)$$

où les S_{k_i} , $i=1, \dots, r$ sont cette fois les torseurs cinématiques complets désirés pour chaque corps C_{k_i} .

Alors:

$$B = \sum_{i=1}^r J_{k_i}^T J_{k_i} \quad (A2-18)$$

Si le robot est une chaîne cinématique ouverte, alors chacun des jacobiens J_{k_i} , de dimensions $6 \times n$, est de rang $\rho_i \leq \min(6, l)$. Or :

$$B = \sum_{i=1}^r B_{k_i} \quad (A2-19)$$

avec:

$$B_{k_i} = \begin{pmatrix} J_{k_i}^T J_{k_i} & 0 \\ 0 & 0 \end{pmatrix} \quad (A2-20)$$

où:

$$J_{k_i} = (J_{k_i}' \mid 0) \quad (A2-21)$$

$$\dim J_{k_i}' = 6 \times k_i$$

Nous avons donc:

$$\min_i \rho_{k_i} \leq \text{rang}(B) \leq k_r$$

et une condition nécessaire, mais non suffisante pour l'inversibilité de B est : $k_r = n$.

Si cette condition est respectée, et si B est régulière, alors les expressions (A2-8) à (A2-11) s'appliquent, avec en particulier:

$$X^g y = B^{-1} \left(\sum_{i=1}^r J_{k_i}^T S_{k_i} \right) \quad (A2-22)$$

Un cas particulier intéressant est celui où:

$$k_i = 1, 2, \dots, n-1$$

Alors, la solution est obtenue à partir d'un autre lagrangien:

$$\Lambda' = \frac{1}{2} \|y - B'^T \dot{q}'\|^2 + \lambda^T (T - (U' \dot{q}' + u \dot{q}_n)) \quad (A2-23)$$

où:

$$\dot{q} = \begin{pmatrix} \dot{q}' \\ \dot{q}_n \end{pmatrix}; \quad \dim(\dot{q}') = n-1 \quad (A2-24)$$

Alors:

$$\dot{q}'^* = U'_g (I - u u_g) T + ((I - U'_g U') + U'_g u u_g U') B'^+ y \quad (A2-25)$$

et:

$$\dot{q}_n^* = u_g T - u_g U' B'^+ y \quad (A2-26)$$

avec:

$$U'_g = H^{-1} U'^T S^{-1} \quad (A2-27)$$

$$H = B'^T B' = \sum_{k=1}^{n-1} J_k^T J_k, \quad (A2-28)$$

supposée régulière,

$$U = (U' \mid u), \quad \dim u = n \times 1 \quad (A2-29)$$

$$S = U' H^{-1} U'^T \quad (A2-30)$$

$$B'^+ = H^{-1} B'^T \quad (A2-31)$$

$$u_g = (u^T S^{-1} u)^{-1} u^T S^{-1} \quad (A2-32)$$

$$y = \sum_{k=1}^{n-1} J_k^T S_k \quad (A2-33)$$

Remarque

On peut également obtenir une autre version de ces équations, si l'on ne désire pas travailler sur toutes les 6 composantes de chaque torseur : alors, chaque J_k est remplacé par J_k , qui ne comporte que les lignes désirées de J_k .

2.6. Algorithmes itératifs

Enfin, signalons l'existence d'algorithmes itératifs (ou séquentiels) de calcul des inverses généralisées. Les plus connus sont la méthode des moindres carrés récursifs pour les systèmes surdimensionnés, et l'algorithme de Gréville ([4]) pour les pseudo-inverses, décrit par exemple en [5].

Bernard ESPIAU

**COMMANDE
DE SYSTEMES REDONDANTS
ET EVITEMENT
D'OBSTACLES**

Bernard ESPIAU

**COMMANDE
DE SYSTEMES REDONDANTS
ET EVITEMENT
D'OBSTACLES**

Publication Interne
n° 280

Janvier 1986

- PI 274** **Détection de pannes et reconfiguration automatique**
Michèle Basseville - 26 pages ; Novembre 85.
- PI 275** **Estimation de l'ordre d'un processus Arma à l'aide de résultats
de perturbations de matrices**
Jean-Jacques Fuchs - 40 pages ; Décembre 85.
- PI 276** **Detection and diagnosis of changes in the eigenstructure of
nonstationary multivariable systems**
Michèle Basseville, Albert Benveniste, Georges Moustakides,
Anne Rougée - 44 pages ; Décembre 85.
- PI 277** **Optimum robust detection of changes in the Ar Part of a
multivariable Arma process**
Anne Rougée, Michèle Basseville, Albert Benveniste, Georges
Moustakides - 48 pages ; Décembre 85.
- PI 278** **Controlling knowledge transfers in distributed algorithms -
Application to deadlock detection**
Jean-Michel Hélyar, Aomar Maddi, Michel Raynal - 32 pages ;
Janvier 86.
- PI 279** **Une méthode de conception de programmes fonctionnels**
Raymond Durand, Martine Vergne - 16 pages ; Janvier 86.
- PI 280** **Commande de systèmes redondants et évitement d'obstacles**
Bernard Espiau - 52 pages ; Janvier 86.

