POLYPUBLIE
Polytechnique Montréal

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Titre: Title: An optimal post-processing module for five-axis CNC milling machines

Auteur: Author: Hojjat Valipour Jahanpisheh

Date: 2007

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Citation: Valipour Jahanpisheh, H. (2007). An optimal post-processing module for five-axis CNC milling machines [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/8029/

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

URL de PolyPublie: PolyPublie URL: https://publications.polymtl.ca/8029/

Directeurs de recherche: Advisors: Luc Baron

Programme: Program: Non spécifié

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal

https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

AN OPTIMAL POST-PROCESSING MODULE FOR FIVE-AXIS CNC MILLING MACHINES

HOJJAT VALIPOUR JAHANPISHEH
DÉPARTEMENT DE GÉNIE MÉCANIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE MÉCANIQUE)
JUIN 2007

© Hojjat Valipour Jahanpisheh, 2007.

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-35703-3
Our file Notre référence
ISBN: 978-0-494-35703-3

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

AN OPTIMAL POST-PROCESSING MODULE FOR FIVE-AXIS CNC MILLING MACHINES

présenté par: VALIPOUR JAHANPISHEH Hojjat

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. BALAZINSKI, Marek, Ph.D., président

M. BARON, Luc, Ph.D., membre et directeur de recherche

M. MAYER René, Ph.D., membre

ACKNOWLEDGMENTS

I would like to tank my supervisor, Professor Luc Baron, for his precious time and effort, invaluable guidance and intellectual support in my graduate study.

Last but not least, I would like to thank the persons who directly or indirectly contributed to my thesis, but are not mentioned here.

RÉSUMÉ

Cette étude présente une procédure de décomposition orthogonale en vue de résoudre la redondance fonctionnelle de machine-outil à commande numerique par l'utilisation d'un calcul de l'inverse généralisé droit de la matrice Jacobienne en considérant les limites articulaires. Une procédure d'optimisation a été implémentée à l'aide d'un module post-processeur et testée sur une architecture d'un centre d'usinage Huron KX0-Five. Ce module a permis d'obtenir avec succès différentes formes de surfaces sans pourtant dépasser les limites articulaires. Les tâches ont été validées avec VERICUT (un logiciel de simulation commercial) avant de procéder à l'usinage avec un centre d'usinage.

ABSTRACT

This study presents an orthogonal decomposition procedure to resolve the functional redundancy of cnc machine-tool by using a direct calculation of the generalized inverse of the Jacobian matrix considering the joint limits of the machine-tool. An optimization procedure has been implemented with a post-processor module and tested with the architecture of the milling center Huron KX8-Five. Different shapes of free form surface have been successfully machined with the module without exceeding joint limits. The tasks have been validated with VERICUT (a commercial simulator software) before machining them with the Milling center.

CONDENSÉ EN FRANÇAIS

Introduction

Les surfaces sculptées (surfaces en forme libre) peuvent être trouvées dans de nombreuses applications industrielles telles que les pièces d'automobiles, les coques de bateaux et les pièces en aérospatial. En effet, les machines d'usinage à cinq axes permettent d'obtenir avec précision des surfaces sculptées très complexes avec les conditions de coupe optimales. De nombreux problèmes peuvent survenir lors du procède d'usinage, par exemple; une qualité de surface usinée inférieure, ou une probable interférence entre l'outil de coupe et les objets aux alentours. Ainsi, il existe une demande pour un logiciel de fabrication assistée par ordinateur (FAO) en vue d'obtenir l'usinage des pièces avec une bonne qualité sans qu'il ait une interférence entre l'outil de coupe et les surfaces des pièces tout en évitant les limites articulaires de la machine-outil.

D'une façon générale les systèmes de FAO peuvent être divisés en trois phases. La première phase a pour but de générer les points de contact du couteau (points CC) sélectionnés en fonction de la géométrie de la surface sculptée. La deuxième phase a pour objet de générer les données de localisation du couteau (données LC) qui dépendent aussi bien des angles spécifiés que de la surface normale aux points CC. Enfin, la dernière phase consiste à convertir les données LC en code machine, c'est aussi l'interférence qui permet de relier les systèmes de FAO aux machines à commande numériques (CN) appelée également la phase de post-processeur. Les positions et orientations de l'outil relativement au référentiel de la pièce nécessitent d'autres transformations par le post-processeur avant de procéder à leur codage en fichier code G. Cette transformation nécessite une connaissance de l'architecture de la machine-outil en vue de résoudre le modèle géométrique inverse, c'est-à-dire le transfert des positions et orientations de l'outil (espace d'opération) en positions des axes de la machine (espace articulaires).

Les modèles géométriques inverses des machines-outils à cinq axes peuvent être trouvées

soit en solution analytique ou numérique itérative. Le calcul des solutions analytiques sont plus rapide que les méthodes numériques. Aussi, c'est plus facile d'appliquer les règles pour lesquelles une ou plusieurs solutions peuvent être sélectionnées mais trouver la solution optimale dans le cas de la redondance est compliquée. Les méthodes numériques itératives pour le calcul des modèles géométriques inverses impliquent d'importantes opérations numériques. Cependant, elles sont puissantes pour retrouver la solution optimale dans le cas d'opérations d'usinage présentant de la redondance comme cela à été démontré dans le cas des robots.

Ce mémoire présente une méthode d'optimisation pour traiter les situations de redondance pouvant apparaître lors d'usinage réalisées à l'aide d'une machine-outil cinq axes. Le controlleure de commandes numériques (CN) ne peut pas prédire si le parcours de l'outil est réalisable ou non. Il est de loin préférable d'avoir une stratégie d'éviter des limites articulaires, et ce dès le début de la résolution du modèle géométrique inverse, afin d'obtenir un parcoures réalisable. Cette spécification peut devenir possible si la dimension de l'espace opérationnel est inférieure à celle de l'espace articulaire. Ce qui signifie que le nombre de degrés de liberté (DDL) nécessaires pour accomplir le travail est inférieur au DDL de la machine. Cette redondance des opérations d'usinage surgit dans plusieurs situations d'usinage où la dimension de l'espace de la tâche (espace d'opération) est plus petite que la dimension de l'espace articulaire. Cela signifie que le DDL requis par l'outil en vue d'exécuter la tâche est inférieur au nombre d'axe de la machine. Par exemple, une tâche de surfaçage utilisant un outil avec bout demi-sphérique peut être considérée comme un problème redondant. Cette tâche nécessite seulement trois DDL et par conséquent peut être réalisée avec seulement une machine-outil à trois axes avec une orientation adéquate de l'outil. Cependant, si la tâche est réalisée avec une machine à cinq axes, il y a deux DDL de redondance et, par conséquent, des spécifications additionnelles sont permises et souhaitables dans le but d'éviter les limites articulaires.

Définition du problème

La redondance peut être présentée dans plusieurs situations d'usinage. Comme on l'a déjà mentionné, ces situations se présentent lorsque le DDL de la machine-outil dépasse celui nécessaire à l'accomplissement de la tâche. Dans le cadre de ce travail, nous avons décidé d'illustrer cette redondance avec l'usinage une surface en forme libre utilisant un outil avec bout demi-sphérique. L'usinage de cette surface avec un code G généré par un post-processeur ordinaire qui ne considère pas les limites articulaires de la machine-outil n'est pas réalisable. Cette tâche a besoin de trois DDL de la position d'outil et l'orientation d'outil n'est pas effet sur la géométrie de surface d'usiner. Pour pouvoir générer le code d'usinage (code G), le post-processeur doit solutionner le modèle géométrique inverse, afin d'atteindre les positions et orientations d'outil désirées.

Modèle géométrique

Comme le standard ISO-3592 le spécifie, le fichier LC contient deux types d'information :

- Des mots post-processeur, qui rien à voir avec le mouvement de l'outil et qui sont donc directement transmis dans le code G.
- Des positions et orientations d'outil par rapport à la pièce, qui nécessitent des transformations par le post-processeur avant d'être codée dans le fichier de code G.

D'une façon générale ces transformations requièrent la connaissance de la longueur de l'outil par rapport au plan de jauge de la broche et/ou, la position de la pièce par rapport au référentiel attaché à la table d la machine pour résoudre le modèle géométrique inverse et transférer les informations sur la position et l'orientation de l'outil (espace opérationnel), en coordonnées articulaires de la machine (espace articulaire). La figure 1 montre la transformation de l'espace opérationnel à l'espace articulaire par le modèle géométrique inverse (MGI) et vice versa par le modèle géométrique direct (MGD).

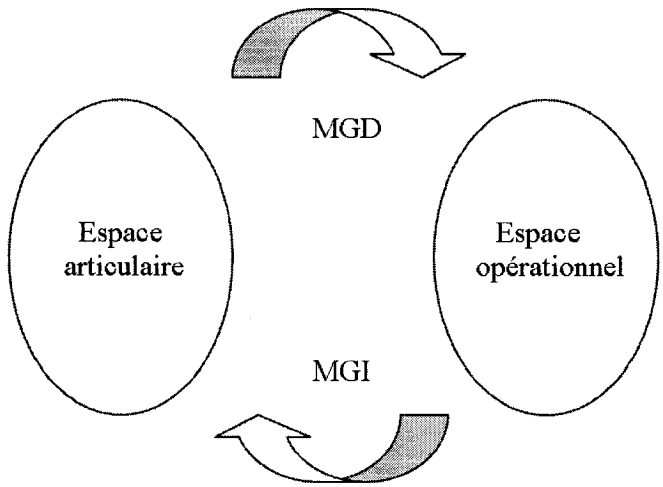


Figure 1: Relation enter les spaces articulaier et opérationnel

Notons que le MGD des machines-outils (qui sont en général des chaînes cinématique sérielles) donne un système d'équations linéaires et donc est facile à résoudre. Par contre, le MGI donne un système modal, non linéaire et beaucoup plus difficile à résoudre.

MGD du Huron KX8-Five

La solution du MGD peut trouver par plusieurs méthodes mathématiques, dans cette étude la méthode Denavit-Hartenberg (D-H) (1955) [13] est choisi pour obtenir la MGD d'une centre usinage comme Huron KX8-Five.

Les directions positives du mouvement des joints articulaires est illustré sur la Fig. 2. Le référentiel P (pièce) est rattaché au centre du plateau et orienté au départ de la même façon que le référentiel global. Le référentiel T (outil) est rattaché au centre du bout demi sphérique de la fraise, son axe z pointant vers le haut de la broche.

La Fig. 3 présenté la chaîne cinématique de la machine ainsi différentes matrices de transformation qui permettent de passer d'un référentiel à un autre. La Fig. 4 schématise les référentiels rattachés à chaque articulation en respectant de la convention D-H. Il faut

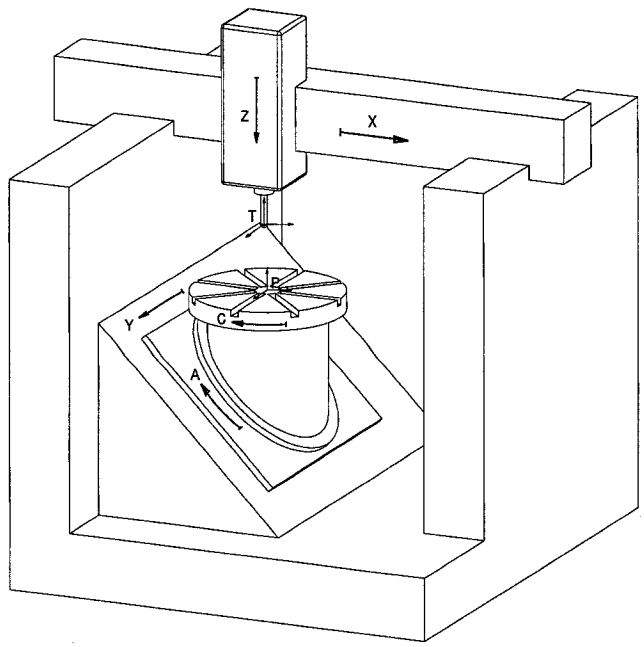


Figure 2: Le Schéma du Huron KX8-Five

rester sur la table de la machine puis regarder les mouvemente des joints articulaires et rattacher les référentiels à chaque articulation en respectant de la convention D-H. Les paramètres D-H sont résumés à la table 3.1. Maintenant les matrices transformations homogenèses entre chaque référentiel peuvent être construire en utilisant eq. 1

$$
{}^{i-1}\mathbf{A}_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & b_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}
$$

Multiplication de ces matrices construit le MGD du Huron KX8-Five. Lest trois premiers rangs des deux dernières colonnes fournissent, la position P et l'orientation N, vecteurs

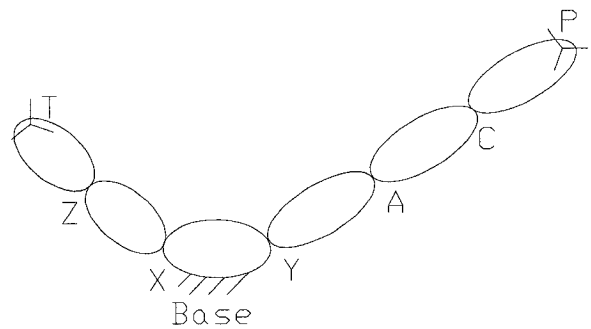


Figure 3: La chaîne cinématique du Huron KX8-Five

Table 1: Les paramètres D-H du Huron KX8-Five

i	θ_i°	a_i	b_i	α_i°	$i+1$
P	(C)-90	0	-d	45	1
1	(A)+90	0	0	90	2
2	45	0	(Y)	-90	3
3	0	0	(X)	-90	4
4	0	0	(Z)+d	0	5
5	90	0	-l	0	T
frame	degree	mm	mm	degree	frame

d'outil par rapport du référentiel de la pièce. Par exemple,

$$\mathbf{P}\begin{bmatrix} x & y & z \end{bmatrix} \text{ où,} \tag{2}$$

$x = Y\sin(C)\cos(A) - 1/2\cos(C)(Z+d-l) - 1/2\,X\cos(C) + \sqrt{2}/2\,X\sin(C)\sin(A)$
$-\sqrt{2}/2\sin(C)\sin(A)(Z+d-l) - 1/2\,X\cos(C)\cos(A) + \sqrt{2}/2\,Y\cos(C)\sin(A)$,
$y = \sqrt{2}/2\cos(C)\sin(A)(Z+d-l) - 1/2\,X\sin(C) + 1/2\sin(C)\cos(A)(Z+d-l)$
$-1/2\sin(C)(Z+d-l) + \sqrt{2}/2\,Y\sin(C)\sin(A) - \sqrt{2}/2\,X\cos(C)\sin(A) - Y\cos(C)$
$\cos(A)$, et
$z = -1/2\,X\cos(A) + 1/2\,X + \sqrt{2}/2\,Y\sin(A) + 1/2\cos(A)(Z+d-l) + 1/2\,(Z-$

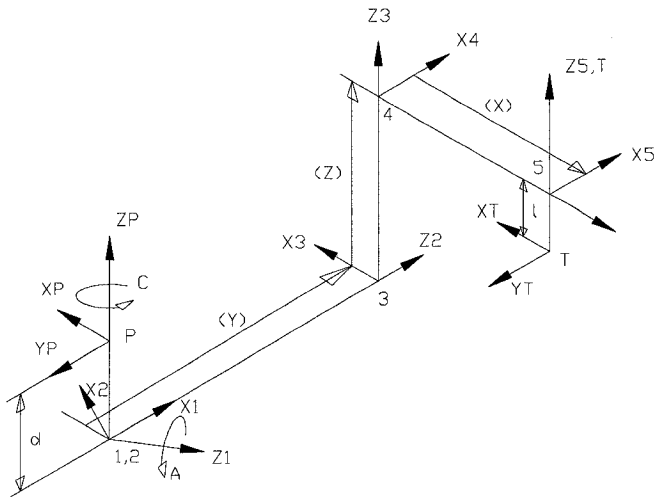


Figure 4: Les référentiels du Huron KX8-Five

$d - l$).

et

$$N = \begin{bmatrix} i & j & k \end{bmatrix} \text{ où,} \tag{3}$$

$i = -1/2 \cos(C) - \sqrt{2}/2 \sin(C) \sin(A) + 1/2 \cos(C) \cos(A)$,

$j = -1/2 \sin(C) + \sqrt{2}/2 \cos(C) \sin(A) + 1/2 \sin(C) \cos(A)$,

et $k = 1/2 \cos(A) + 1/2$,

Ainsi que l est de la longueur de l'outil, i.e., la distance entre au plan de jauge de la broche et le centre d'outil avec bout demi-sphérique. Aussi d est défini la distance entre le plan sur surface de la table et axe rotative A. De plus, les limites articulaires du Huron

sont comme suivant:

$$\begin{aligned} -315\,mm \leq\ & X & \leq 335\,mm \\ -350\,mm \leq\ & Y & \leq 350\,mm \\ 0\,mm \leq\ & Z & \leq 450\,mm \\ -180^\circ \leq\ & A & \leq 45^\circ \\ -99,999.999^\circ \leq\ & C & \leq 99,999.999^\circ \\ & d & = 75\,mm \end{aligned} \tag{4}$$

Critère d'optimisation

Pour pouvoir générer le code d'usinage (code G), le post-processeur doit solutionner le MGI, et ainsi calculer les déplacements et vitesses de chaque articulation, afin d'atteindre les positions et orientations d'outil désirées. Le post-processeur devrait aussi permettre à l'usiner de prévoir et éviter les situations de dépassement articulaire avant de générer le code G. Dans notre cas, la résolution de la redondance va s'appliquer au niveau des vitesses, puisque à ce niveau le système d'équation est linéaire. Comme on l'a déjà mentionné, une tâche de surfaçage utilisant un outil avec bout demi-sphérique a besoin de trois DDL de la position. Donc, les vitesses articulaires sont reliées aux vitesses de déplacement de l'outil par rapport à la pièce par la relation:

$$J\dot{\boldsymbol{\theta}} = \dot{\mathbf{P}} \tag{5}$$

où J est la matrice Jacobianne de la machine-outil, $\dot{\mathbf{P}}$ le vecteur de vitesse de bout de l'outil par rapport à la pièce et $\dot{\boldsymbol{\theta}}$ le vecteur de vitesses articulaires.

Étant donné que notre système est redondant par rapport à la tâche à accomplir, cela

donne un système sous-déterminer. Donc, une infinité solutions existe pour résoudre ce système. Il s'agit alors d'en trouver une qui corresponde à la tâche qui doit être réalisée et qui permet de répondre à un critère secondaire. La réponde du système est bien connue être

$$\dot{\theta} = J^{+}\dot{\mathbf{P}} + (I - J^{+}J)\mathbf{h}, \tag{6}$$

où J^{+}représente l'inverse généralisé de droit de J, i.e., $J^{+} = (J^{T}J)^{-1}J^{T}$ et $\mathbf{h}$ est la tâche secondaire. La première partie de l'équation 6 représente la solution homogène du système, avec la matrice $(I - J^{+}J)$ comme étant le complément orthogonal de J, qui projette le vecteur $\mathbf{h}$ sur l'espace nulle de J. Si on choisit différente valeur pour le vecteur $\mathbf{h}$, on aura différentes solutions pour $\dot{\boldsymbol{\theta}}$. La première partie de l'équation 6 ne contrôlant pas l'orientation de l'outil, $\mathbf{h}$ doit contenir au moins un terme mettant en relation la normale à la surface à usiner et l'axe de l'outil. Avec ce seul terme, il est donc possible d'optimiser l'orientation de l'outil afin de l'approcher de celle de la normale à la surface avec un certain angle d'erreur admissible. Concrètement, l'axe de l'outil serait en tout point de la trajectoire dans un cône de sommet le bout de l'outil (en contact avec la surface), d'axe la normale à la surface et d'angle l'erreur maximale désirée. Cependant, on peut également ajouter un autre terme à $\mathbf{h}$. Dans l'application, une contrainte sur les valeurs articulaires de la machine a été introduite afin de minimiser ses déplacements en utilisant un des deux critères suivants pour ce problème d'optimisation:

- Imposer aux valeurs articulaires de la machine d'être proches de leurs valeurs moyennes (i.e. dans le repère global de la machine),
- Imposer aux valeurs articulaires de la machine d'être proches des valeurs moyennes calculées le long de la trajectoire à parcourir (i.e. dans le repère local de la pièce).

Ces deux techniques sont relativement proches. Cependant, dans la première on s'assure d'être loin des limites articulaires tandis que dans la seconde, l'optimisation de la trajectoire est locale et prend donc en compte les dimensions de la pièce. Par exemple, si

l'on veut usiner un plan dont les dimensions dans le plan XY sont: $X = 200mm$ et $Y = 50mm$, il est clair qu'il vaut mieux se soucier des mouvements de l'axe X plutôt que de ceux de l'axe Y. Dans ces deux approches, la fonction objectif z est définie de la même manière, seuls les poids accordés à chacun des termes changent:

$$z = \frac{1}{2}(\theta - \overline{\theta})^T W^T W(\theta - \overline{\theta}) + W_0(1 - O^T \cdot O_f) \tag{7}$$

où $\bar{\boldsymbol{\theta}}$ est le vecteur des valeurs moyennes de $\bar{\theta} = \frac{1}{2}(\theta_{max} + \theta_{min})$, W est une matrice 5×5 de poids, diagonale et définie positive $W = diag(W_C, W_A, W_Y, W_X, W_Z)$, W_0 est un réel, O est l'orientation de l'outil et O_f est la normale à la surface. Le premier terme de cette équation exprime l'écart des positions articulaires par rapport aux valeurs moyennes. Le second terme caractérise l'écart d'orientation. L'objectif est de minimiser z. On définit donc **h** comme

$$\mathbf{h} = -\nabla z, \tag{8}$$

où ∇ correspond à l'opérateur gradient. Pour vérifier eq. (8), considérons le paraboloïde d'équation: $z = x^2 + y^2$ pour lequel $\nabla z = |\frac{2x}{2y}|$. Il est clair que le vecteur ∇z pointe dans la direction opposée à l'origine o, point correspondant au minimum de z. Pour un problème de minimisation, il faut donc choisir $\mathbf{h} = -\nabla z$.

Donc, on a

$$h = -W(\theta - \overline{\theta}) + W_0 \frac{d}{d\theta}(O^T \cdot O_f) \tag{9}$$

Le module post-processeur

Une solution numérique d'itération est utilisée pour résoudre cinématique inverse, base sur la théorie du " Resolve Motion Rate " de Whitney (1969) [36]. La solution présentée dans eq. (6) et h la tâche secondaire de eq. (9) sont appliquées dans un module post-processeur optimal la quelle qui peut être illustré sur algorithme suivant:

L'algorithme 0.1 d'un module post-processor optimal

Lire $\mathbf{P}_\mathrm{f}$ et $\mathbf{O}_\mathrm{f}$ de chaque ligne de fichier APT

$\boldsymbol{\theta}$, Wet $\mathrm{W}_0 \longleftarrow$ conjecture initial

$\{\mathbf{P},\ \mathbf{O},\ J\} \longleftarrow MGD(\boldsymbol{\theta})$

For i = 1 to 500 step 1 do

$\boldsymbol{\Delta}\mathbf{P} \longleftarrow 0.5\ (\mathbf{P}_f - \mathbf{P})$

$h \longleftarrow -\nabla Z$

$\boldsymbol{\Delta\theta} \longleftarrow J^{+}\boldsymbol{\Delta}\mathbf{P} + (1 - J^{+}J)\ h$

if $\|\boldsymbol{\Delta}\mathbf{P}\| \leq \epsilon \cong 10^{-6}\mathrm{m},$ then stop;

else

$\boldsymbol{\theta} \longleftarrow \boldsymbol{\theta} + 0.8\boldsymbol{\Delta\theta}$

$\{\mathbf{P},\ \mathbf{O},\ J\} \longleftarrow MGD(\boldsymbol{\theta})$

end if

End do

Les vecteurs $\mathbf{P}_f$ et $\mathbf{O}_f$ dénotent la position et l'orientation désirée du référentiel d'outil par rapport au référentiel de la pièce. La fonction MGD calcule la position et l'orientation actuelle du référentiel d'outil par rapport au référentiel de la pièce. Après chaque " Do " itération, les prochains valeurs des articulaires $\boldsymbol{\theta}_{i+1}$ calculées sont considérées comme les valeurs actuelles articulaires $\boldsymbol{\theta}i$ plus un facteur d'amortissement fois des petits déplacements articulaires $\boldsymbol{\Delta\theta}$ afin d'éviter dépassement du $\boldsymbol{\theta}_{i+1}$, au moment du conditionnement de la matrice Jacobienne.

Notre algorithme permette une interpolation linéaire entre la position actuelle et la position finale du centre d'outil avec bout demi-sphérique, pendant qui laisse le contrôle de l'orientation de l'outil à la tâche secondaire. Le deuxième parti de l'équation 6 essaie de garder des valeurs articulaires aux plus proches des valeurs moyennes et de garder d'orientation d'outil au plus proche de la déviation d'orientation désirée d'outil. Une sélection correcte de poids permette un équilibrage entre les deux critères d'optimisation

à la tâche secondaire.

L'assai d'expérimental

Le module post-processeur optimal est appliqué sur d'un centre d'usinage Huron KX0-Five. Le fichier CL est généré à l'aide de CATIA V5, pour une surface en forme libre (voir Fig. 5).

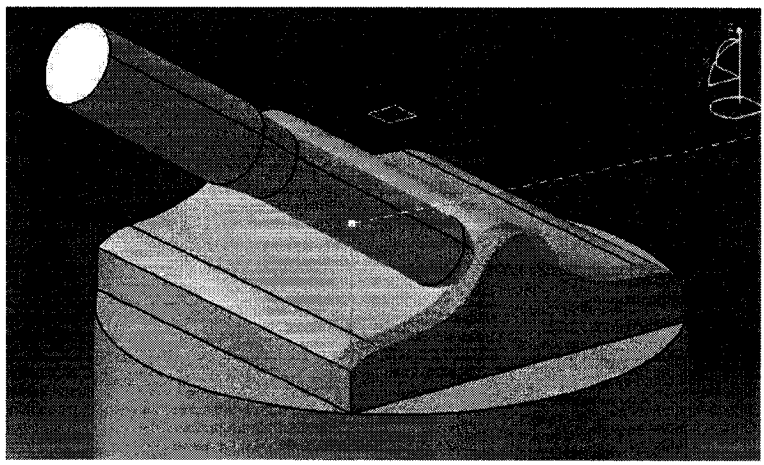


Figure 5: Parcous d'outil généré par CATIA

Une parcourir multi-axes pour un outil avec bout demi-sphérique est choisie comme la tâche surfaçage. L'orientation d'outil est choisie normal à la surface. Le fichier APT est post-procédé pour l'Huron dans quatre approches afin de démontrer la possibilité d'opération et de comparer l'effet d'orientation d'outil sur grossièreté de la surface, i.e.;

1. l'outil toujours parallèle au vertical ;
2. l'outil toujours a une orientation constante différant avec le vertical ;
3. l'outil toujours normal à la surface; et finalement,
4. l'outil presque normal à la surface utilisant le module post-processeur optimal afin d'éviter des limites articulaires de la machine-outil.

La Fig. 6 montre les valeurs articulaires calculées par troisième processus. D'après eq. (4) les limites articulation A sont -180 ¡ A¡ 45, mais la Fig. 6 montre que les valeurs calculées pour A sont -82.5 et +82.5. Donc, l'articulation A est dépassée à ses limites et code G généré pour ce processus n'est pas possible à réaliser. La tâche doit être procédée avec une post-processeur optimal lequel qui peut identifier les limites articulaires.

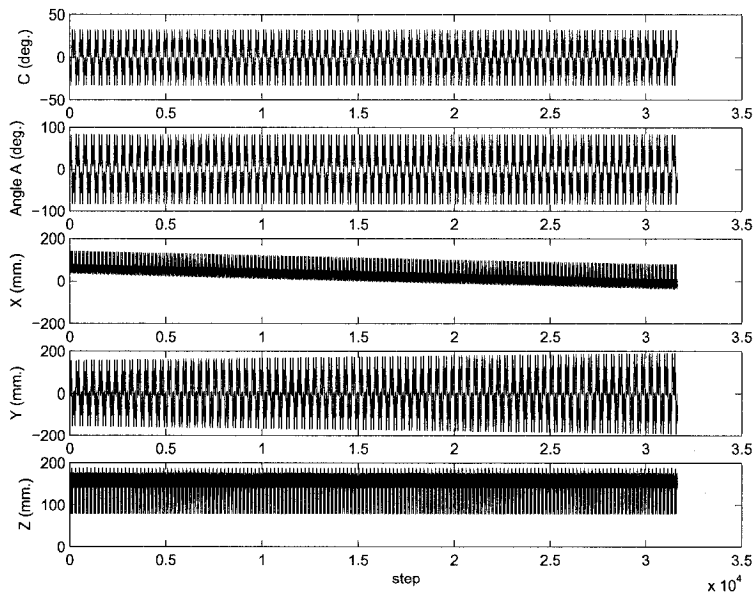


Figure 6: Les valeurs articulaires calculées par troisième processus

Les valeurs articulaires calculées par un module post-processeur optimal sont illustrées sur la Fig. 7, celle-ci montre que les valeurs calculées pour A sont entre -73.5 et 43.5 c'est-à-dire entre ses limites. Donc, le code G généré par ce module post-processeur optimal peut être accomplir par la mchine-outil sans pourtant dépasser les limites de articulaire.

Conclusion

Une procédure de décomposition orthogonale en vue de résoudre la cinématique inverse des machines-outils cinq axes est utilisés pour exécuter une tâche surfaçage. La

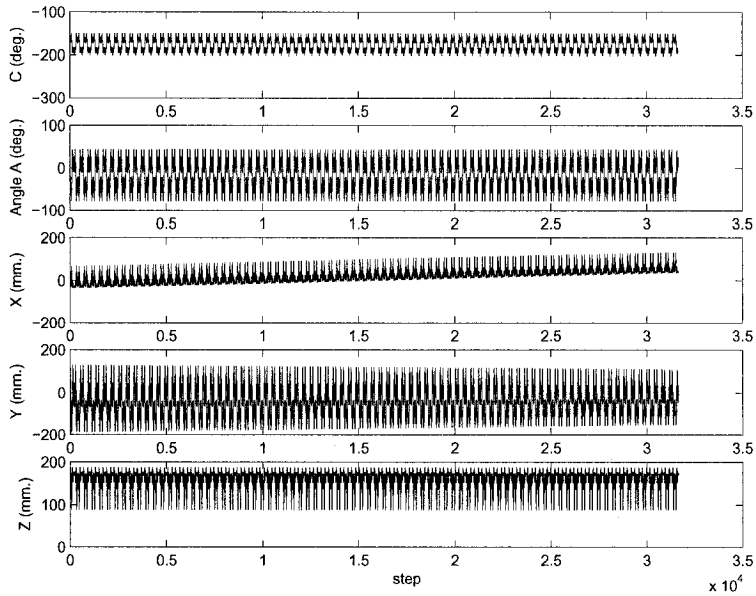


Figure 7: Les valeurs articulaires calculées par un module post-processeur optimal

procédure d'optimisation a été implémentée à l'aide d'un module post-processeur et testée sur une architecture d'un centre d'usinage Huron KX0-Five. Ce module a permis d'obtenir avec succès différentes formes de surfaces sans pourtant dépasser les limites des articulaires et l'outil a parcouru son trajet de façon continue. Les tâches ont été validées avec VERICUT (un logiciel de simulation commercial) avant de procéder à l'usinage avec un centre d'usinage afin d'éviter des n'importe quelles des inconvenantes erreurs. Les assais d'expérimentaux ont démontrés possibilité d'usinage le parcours optimal. Afin de comparer l'effet d'orientation sur grossièreté de la surface. La grossièreté des surfaces est mesurée par Mitutoyo Surftest (une machine à mesurer de la grossièreté de la surface). Tous les résultats démontrent que la grossièreté de la surface qui usinée à l'aide du code G généré par le module post-processeur optimal a amélioré.

Travaux futurs

Une difficulté rencontrée au cours de ces travaux consistait à choisir la bonne valeur de pondération, alors la recherche sur les méthodes automatique pour estimer la bonne valeur de pondération est requise.

Notre module post-processeur optimal ne fournit pas le code G aux machines-outils à grande vitesse. Donc, l'étude sur un module post-processeur optimal pour les machines-outils à grande vitesse est proposée comme travail futur.

TABLE OF CONTENTS

ACKNOWLEDGMENTS iv

RÉSUMÉ v

ABSTRACT vi

CONDENSÉ EN FRANÇAIS vii

TABLE OF CONTENTS xxii

LIST OF FIGURES xxv

LIST OF NOTATIONS AND SYMBOLS xxviii

LIST OF TABLES xxx

LIST OF APPENDICES xxxi

INTRODUCTION 1

CHAPTER 1 KINEMATIC MODELING OF MACHINE-TOOL 3

1.1 The Architecture of Five-Axis Machine-Tools 3

1.2 The Kinematic Chain of Five-Axis Machine-Tools 5

1.3 Direct and Inverse Kinematic Problem 6

1.4 Direct Kinematics (DK) Model 7

1.5 Homogeneous Transformation Matrix 7

1.6 The Denavit-Hartenberg Convention 12

1.7 D-H Modeling of the Matsuura 17

1.8 Different Method of Using D-H 21

1.9 The DK Modeling, Using a New Method 22

1.9.1 D-H Parameters from Base to the Workpiece 22
1.9.2 Define D-H Parameters from Base to Tool 23
1.10 Velocity Modeling 27
1.10.1 Differential-Form of the Jacobian matrix 28
1.10.2 Vector-Form of the Jacobian Matrix 31
1.11 Algorithm to Find the DKM 33
1.12 Inverse Kinematics Model 36
1.12.1 The Closed-Form Solution 36
1.12.2 Iterative Numerical Solution 37

CHAPTER 2 REDUNDANCY AND OPTIMIZATION 41
2.1 Redundancy 41
2.2 Optimization and Redundancy Resolution 41
2.2.1 Generalized Inverse and Pseudoinverse 42
2.2.2 Weighted Generalized Inverse Method 48
2.2.3 Householder Reflection Method 48
2.2.4 Elimination Method 50
2.2.5 Virtual Joint Method 51
2.2.6 Twist Decomposition Method 51
2.3 Secondary Task for Redundant Machining Operation 52
2.4 Conclusion 53

CHAPTER 3 OPTIMAL POST-PROCESSING OF MACHINE-TOOLS . . 54
3.1 Cutter Location File (CL-File) 54
3.2 Linearization 55
3.3 Algorithm for the Optimal Post-Processing 56
3.4 Experimental Test of the Optimal Post-Processor 58
3.5 Example of Operation 64
3.6 Post-Processor Output 64

3.7 Machining Tests 71

CHAPTER 4 CONCLUSION 75

4.1 Future Works 75

REFERENCES 76

APPENDICES 81

LIST OF FIGURES

Figure 1 Relation enter les spaces articulaier et opérationnel x

Figure 2 Le Schéma du Huron KX8-Five xi

Figure 3 La chaîne cinématique du Huron KX8-Five xii

Figure 4 Les référentiels du Huron KX8-Five xiii

Figure 5 Parcous d'outil généré par CATIA xviii

Figure 6 Les valeurs articulaires calculées par troisième processus xix

Figure 7 Les valeurs articulaires calculées par un module post-processeur optimal xx

Figure 1.1 Types of 5-axis milling machines: (a) table rotating type; (b) spindle and table rotating type; (c) spindle rotating type. 4

Figure 1.2 The nomenclature of axis (http://www.mmsonline.com) 5

Figure 1.3 The kinematic chain of a five-axis machine-tool 6

Figure 1.4 Kinematic mapping between joint and Cartesian spaces 7

Figure 1.5 Two frames F_0 and F_1 in space 8

Figure 1.6 The schema of the Matsuura five-axis milling center 9

Figure 1.7 The kinematic chain of the Matsuura five-axis milling center . . 9

Figure 1.8 Definition of X_i when Z_{i-1} and Z_i: (a) are skew; (b) intersect; and (c) are parallel [1] 13

Figure 1.9 Coordinate frames of PUMA robot [1] 14

Figure 1.10 D-H parameters [26] 15

Figure 1.11 (a) The robot Fanuc M16iB in its initial position; (b) The coordinate frames of the Fanuc M16iB 16

Figure 1.12 Geometry of the Matsuura 5-axis milling center without tool at home position 18

Figure 1.13 Matsuura with a spherical end-mill tool of length ℓ at home position 19

Figure 1.14 Coordinate frames of Matsuura from workpiece to tool frames . 20

Figure 1.15 The kinematic chain of a five-axis machine-tool 21

Figure 1.16 The kinematic chain of the Matsuura ; (a) left hand from the base to the Tool; (b) right hand from the base to the workpiece . . . 22

Figure 1.17 Link frames of the Matsuura from the base to the workpiece . . 23

Figure 1.18 Link frames of the Matsuura from the base to the tool 25

Figure 1.19 Link frames of the Matsuura from the base to the workpiece . . 26

Figure 1.20 General n-axis manipulator [1] 31

Figure 1.21 The definition of e_i and r_i for the Matsuura 32

Figure 3.1 Tool path of five-axis milling [17] 55

Figure 3.2 Milling machine center Huron KX8-Five 58

Figure 3.3 Schema of Huron KX8-Five 59

Figure 3.4 The kinematic chain of Huron KX8-Five 59

Figure 3.5 Coordinate frames of Huron KX8-Five from the workpiece to the tool 60

Figure 3.6 Joint limits of the Huron KX8-Five 61

Figure 3.7 Tool path generated by CATIA 64

Figure 3.8 The processed joint positions value when the tool is normal of the surface 65

Figure 3.9 The processed value of joints form optimal post-processing (fourth test) 67

Figure 3.10 Condition number of the Jacobian matrix from optimal post-processing 68

Figure 3.11 Positioning error from optimal post-processing 68

Figure 3.12 Comparing the joint positions A and C from the third and fourth tests. (normal and optimal post-processing) 69

Figure 3.13 Normal post-processing 70

Figure 3.14 Optimal post-processing 70

Figure 3.15 Relative tool orientation between processes 3 and 4 70
Figure 3.16 Simulation of a part 71
Figure 3.17 The Huron KX8-Five in a multi-axis machining process 72
Figure 3.18 The four machining tests 73
Figure 3.19 Location of measurement on the parts 73

LIST OF NOTATIONS AND SYMBOLS

${}^{\mathcal{F}_j}\mathbf{A}_{\mathcal{F}_i}$: Homogenous transformation matrix describing frame $\mathcal{F}_i$ with respect to frame $\mathcal{F}_j$;

A: Revolute axis of machine-tool;

$\mathbf{A}r$: Right hand HTM;

$\mathbf{A}l$: Left hand HTM;

a_i: Position vector between each frame according to D-H;

a_i: D-H notation;

α_i: D-H notation;

B: Revolute axis of machine-tool;

b_i: D-H notation;

C: Revolute axis of machine-tool;

CAM: Computer Aided Manufacturing;

CC points: Cuter Contact points;

CL data: Cuter Location data;

CN: Commande Numérique;

CNC: Computer Numerical Control;

D-H: Denavit-Hartenberg;

DDL: Degrés De Liberté;

DKP: function solving DKP;

DOF: Degrees Of Freedom;

FAO: Fabrication Assistée par Ordinateur;

NC: Numerical Control;

EE: End-Effector;

e_i: Unit vector associated with each revolute axis;

IKP: function solving IKP;

J: Jacobian matrix;

HTM: Homogeneous Transformation Matrix;

ℓ: Tool length;

LC: Localisation du Couteau;

MGD: Modèle Géométrique Direct;

MGI: Modèle Géométrique Inverse;

$\mathbf{N}$: Vector of cuter orientation;

O_f: Desirable tool orientation;

$\mathbf{P}_{1\times3}$: Position vector;

$\mathbf{p}$: Position vector in Cartesian space;

$\mathbf{Q}$: Rotation matrix between two frames;

$\mathbf{R}_{3\times3}$: Rotation matrix;

T: Tool coordinate frame;

$\mathbf{t}$: Twist array expressing the end-effector velocity;

$\boldsymbol{\theta}$: Joint position;

$\boldsymbol{\theta}_r$: Right hand joint position;

$\boldsymbol{\theta}_l$: Left hand joint position;

$\dot{\boldsymbol{\theta}}$: Joint velocity;

r_i: Position vector of EE in base frame;

vect: function transforming a 3×3 rotation matrix into axial vector;

ω: Angular velocity;

$\mathbf{W}$: positive-definite weighting matrix;

W_0: Weighting value;

X: Prismatic axis of machine-tool;

Y: Prismatic axis of machine-tool;

Z: Prismatic axis of machine-tool;

LIST OF TABLES

Table 1	Les paramètres D-H du Huron KX8-Five	xii
Table 1.1	D-H parameters of the Fanuc M16iB	17
Table 1.2	D-H parameters from the workpiece to tool frames	19
Table 1.3	D-H parameter notations from the base to the workpiece	24
Table 1.4	D-H parameter from the base to the workpiece	24
Table 1.5	D-H parameters of the Matsuura (new method)	27
Table 3.1	D-H parameter notations from the workpiece to the tool	60
Table 3.2	Values of Roughness of four parts	74

LIST OF APPENDICES

APPENDIX I THE RESULTS AND FIGURES OF ROUGHNESS MEASUREMENT 81

APPENDIX II MATLAB PROGRAMM BASE ON ALGORITHM 1.1 . . . 102

INTRODUCTION

The sculptured surfaces (free-form surfaces) have found extensive industrial applications, such as in automobile bodies, ship hulls and aerospace parts. In fact, five-axis machining can perform efficient and accurate machining for complicated sculptured surfaces.

There are several problems that may arise during the machining process, such as low quality of machined surface and possible interference between the tool and surrounding objects. Hence, there is a demand for a computer-aided manufacturing (CAM) software to achieve machining with good surface quality and no interference between the tool and the part surfaces, while avoiding the joint limits of the machine-tool.

In general, CAM systems are applied at three stages. The first is to generate cutter contact points (CC points), which are selected based on the geometry of the sculpture surface. The second stage is to generate cutter location data (CL data), which depend both on the specified angles and surface normal at the CC points; Finally, the last stage is to convert CL data to machine code, the interface that links the CAM systems and NC machines, which is called the post-processor stage [14]. The tool positions and orientations relative to the part frame requires further transformations by post-processor before encoding it into G code file. This transformation requires knowledge of the architecture of machine tool in order to solve the inverse kinematics, i.e., transferring the tool positions and orientations (operation space) into machine axes positions (joint space).

The inverse kinematics of five-axis machine-tools can be found either as a closed form solution (analytical solution) or as an iterative numerical solution. Calculation of the closed form solutions is faster than numerical methods. It is also easy to apply the rules for which one of several possible inverse solutions can be selected, but finding an optimal solution in the case of redundancy in this method is complicated. The iterative numerical methods for calculation of the inverse kinematics involve a large number of

numerical operations. However, they are powerful in finding the optimal solution in the case of the redundant machining operations as it has been demonstrated for industrial robots.

The redundancy of machining operations arises in many different machining situations where the dimension of the task space (operation space) is smaller than the dimension of the joint space. It means that the number of degree of freedom (dof) required by the tool to perform the task is less than the number of the machine axes. For example, a surfacing task using a ball-end mill tool can be considered as a redundant problem in this study. This task requires only three dofs, and hence, can be realized with only a 3-axis machine-tool with a proper orientation of the tool. However, if the task is realized with a 5-axis machine-tool, there are two dofs of redundancy, and hence, additional specifications are allowable and desirable in order to avoid joint limits, for example.

CHAPTER 1

KINEMATIC MODELING OF MACHINE-TOOL

1.1 The Architecture of Five-Axis Machine-Tools

The earliest introductions to five-axis milling were given by Baughman (1970) [9], Bezier (1972) [10] and also Boyd (1974) [11]. The five-axis machine-tools are created with combination of three prismatic joints and two revolute joints. Many studies classify 5-axis milling machines into three basic categories [19] as in Figures 1.1. In all categories, the three prismatic joints (P) are called X,Y and Z and the two revolute joints (R) are called A,B or C, according to Figure 1.2. An axis is called A if it rotates around a line parallel to the X-axis, it is called B if it rotates around a line parallel to the Y-axis and it is called C if it rotates around a line parallel to the Z axis. [1]

As shown in Fig. 1.1 (a), type (a) machines have two revolute joins as A and C or B in table and three prismatic joints as X, Y and Z in spindle or table. A five-axis milling machine can be realize with a three-axis machine by adding a rotating/tilting mechanism on its table which is the most economic way for up-to-dating a three axis machine tools. In this type, the load imposed on the spindle is less than other type since the spindle does not tilt and also this type can machine well cylindrical parts with a ring of holes in top face. They are good to use long tools or extensions because tool holder moves with prismatic joint. However, heavy workpieces are not suitable for this type because the rotary table need to power rotating and tilting mechanism. Type (c) machines have two revolute joints as A and B on the cutter spindle and three translation on table or spindle. The major advantage of this type is that heavy parts can be machined easily because

[1] http//www.mmsonline.com

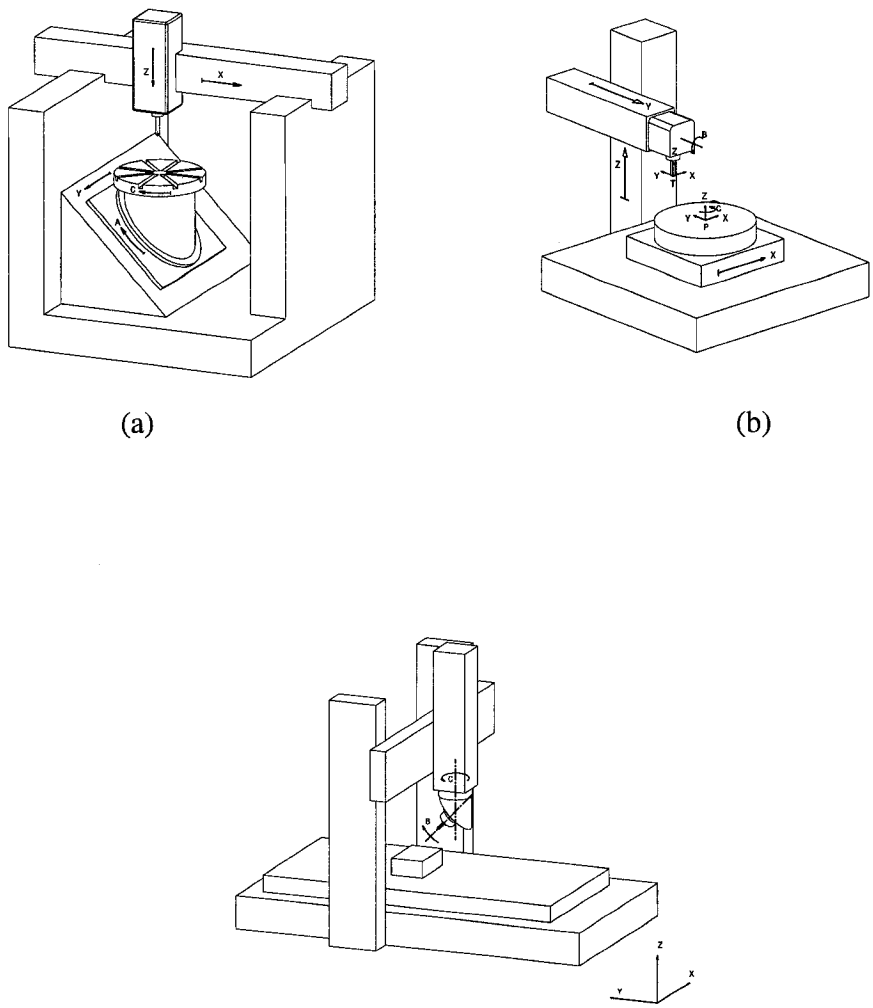


Figure 1.1: Types of 5-axis milling machines: (a) table rotating type; (b) spindle and table rotating type; (c) spindle rotating type.

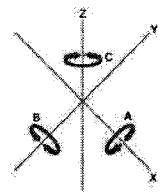


Figure 1.2: The nomenclature of axis (http://www.mmsonline.com)

they do not need rotating and titling. They can machine well such parts as molds of tyre because the molds are heavy and have sculpture shape in their inside that spindle and tool can reach for machining. But because of space limitation, the rotary mechanism for spindle is not enough powerful for heavy machining (i.e., milling a hard material). Type (b) machines are a compound of the other two types the rotary joints as B, A or C are fixed on spindle and table. They are widely used for medium and small part in the die and mold industry, for example.

1.2 The Kinematic Chain of Five-Axis Machine-Tools

The simplest way to model the kinematics of a machine-tool is using the concept of kinematic chain. In a machine-tool or any other mechanisms, each rigid body, called link, are coupled by kinematic pair that constraint the relative motion of the links. There are six possible kinematic pairs, i.e.; revolute (R), prismatic (P), helicoidal (H), planar (E), cylindrical (C) and spherical (S). In the case of machine-tool only R or P joints are used. Figure 1.3 shows the kinematic chain of a five-axis machine-tool of type (b) of Fig. 1.1 .

The coordinate system P (standing for part frame) is attached on top of a rotary table and the coordinate system T (standing for tool frame) is attached on head of the tool, which is connected to the spindle. In this kinematic chain, the base link is fixed and two series

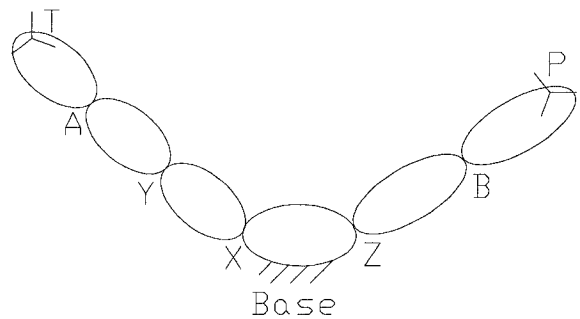


Figure 1.3: The kinematic chain of a five-axis machine-tool

of links are branched to the base. The right branch is connected by the two joints, Z and B to the base, while the left branch is connected by three joints, X, Y and A to the base. This kinematic chain shows that the motion of frames P and T relative to the base are not important to the view of the part to be machined, but rather the motion of T relative to P.

1.3 Direct and Inverse Kinematic Problem

Figure 1.4 shows the mapping between joint space and cartesian space at displacement level. The Direct Kinematic Problem (DKP) or the forward kinematic problem is the mapping from joint space to cartesian space (operation space), i.e., the DKP is to find the position and the orientation (the pose) of the end-effector (EE) or tool from the value of joint position. This problem admits a single solution, which can be determined by simple matrix and vector multiplications. The Inverse Kinematic Problem (IKP) is the mapping from cartesian space to joint position, i.e., the IKP is to find the joint angles from the pose of the tool. The solution of IKP may give a set of solutions or in some cases no solution;

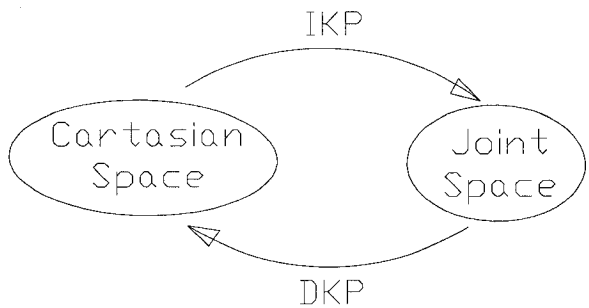


Figure 1.4: Kinematic mapping between joint and Cartesian spaces

1.4 Direct Kinematics (DK) Model

As mentioned before, the DKP is to determine the pose of the EE from a given set of joint positions. In machine-tool, the DKP is to find the pose of the tool from the axis positions. The DK model can be obtained through the several mathematical tools such as Homogeneous Transformation Matrix (HTM), Denavit-Hartenberg (D-H) convention (1955) [13], dual numbers (Bottema and Roth 1979) [8], Screw theory (Sugetomo and Duffy 1982) [35] or Dual Quaternions, (McCarthy 1990) [25]. Each of these tools has its own particular method. In this study, the HTM and D-H convention are chosen as the DK modeling methods.

1.5 Homogeneous Transformation Matrix

An HTM is a 4×4 matrix expressing the pose, position and orientation, of a frame relative to another frame.

Figure 1.5 shows two frames F_0 and F_1 in space.

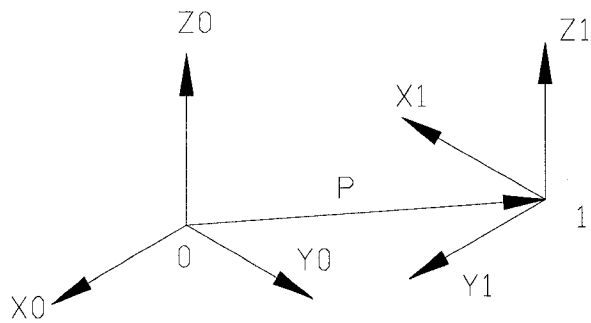


Figure 1.5: Two frames F_0 and F_1 in space

The HTM can be shown as following;

$$ {}^{0}\mathbf{A}_1 = \begin{bmatrix} \mathbf{R}_{3\times3} & \mathbf{P}_{1\times3} \\ 0\,0\,0 & 1 \end{bmatrix} \tag{1.1} $$

consists of the $\mathbf{R}_{3\times3}$, which is the rotation matrix expressing the orientation of frame F_1 in F_0; and the $\mathbf{P}_{1\times3}$, which is the position vector of F_1 in F_0. In other word, the coordinate frame F_1 can be obtained from a translation of F_0 along $\mathbf{P}_{1\times3}$ then a rotation of $\mathbf{R}_{3\times3}$. The multiplication of the HTMs of link by link along its kinematic chain, will defines the DK model. From Fig. 1.3, the ${}^{P}\mathbf{A}_T$ as the DK model is computed from multiplication of the following HTMs: ${}^{P}\mathbf{A}_T = {}^{P}\mathbf{A}_B\ {}^{B}\mathbf{A}_{base}\ {}^{base}\mathbf{A}_X\ {}^{X}\mathbf{A}_Y\ {}^{Y}\mathbf{A}_A\ {}^{A}\mathbf{A}_T$

Figure 1.6 shows the schema of the type (a) Matsuura five-axis milling center. Frame B is a reference frame attached to base of the milling center, frame T is attached to milling tool, while frame P is fixed on top of rotary table. Additionally, frame P1 is attached to the rotary axis A above the table. the kinematic chain of the Matsuura, is shown in Fig. 1.7. This machine is defined with two revolute joints as C and A and three prismatic joints as X,Y and Z. The DK model expresses the position and orientation of the tool frame T relative to the workpiece frame P, when the value of the joints (C,A,X,Y,Z) are known.

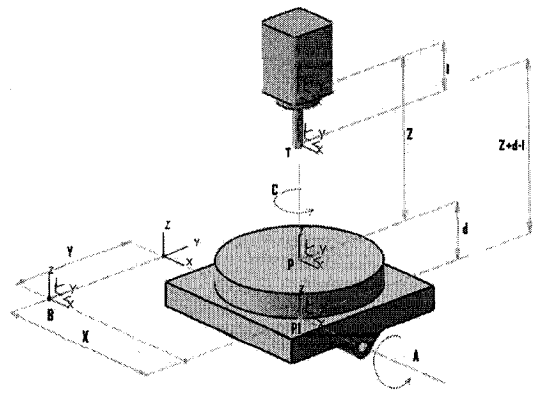


Figure 1.6: The schema of the Matsuura five-axis milling center

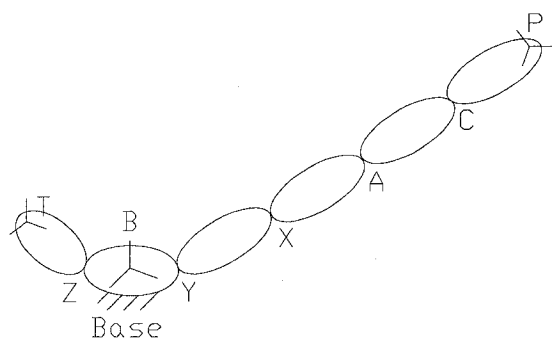


Figure 1.7: The kinematic chain of the Matsuura five-axis milling center

According to Fig. 1.7, we have

$$ {}^{B}\mathbf{A}_{T} = {}^{B}\mathbf{A}_{P}\,{}^{P}\mathbf{A}_{T} \tag{1.2} $$

where, the pose of T relative to P is given as:

$$ {}^{P}\mathbf{A}_{T} = [{}^{B}\mathbf{A}_{P}]^{-1}\,{}^{B}\mathbf{A}_{T} \tag{1.3} $$

Now, the left branch of the kinematic chain of Fig. 1.7 is

$$
{}^{B}\mathbf{A}_{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d+Z-l \\ 0 & 0 & 0 & 1 \end{bmatrix} , \tag{1.4}
$$

while the right branch is given as

$$
{}^{B}\mathbf{A}_{P} = {}^{B}\mathbf{A}_{P1}\,{}^{P1}\mathbf{A}_{P} , \tag{1.5}
$$

where ${}^{B}\mathbf{A}_{P1}$ and ${}^{P1}\mathbf{A}_{P}$ are given as

$$
{}^{B}\mathbf{A}_{P1} = \begin{bmatrix} 1 & 0 & 0 & X \\ 0 & C_A & -S_A & Y \\ 0 & S_A & C_A & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } {}^{P1}\mathbf{A}_{P} = \begin{bmatrix} C_C & -S_C & 0 & 0 \\ S_C & C_C & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.6}
$$

and $C_i \equiv \cos(i)$ and $S_i \equiv \sin(i)$. Upon substituting eqs. (1.6) into (1.5) gives

$$
{}^{B}\mathbf{A}_{P} = \begin{bmatrix} C_C & -S_C & 0 & X \\ C_A S_C & C_A C_C & -S_A & -S_A d + Y \\ S_A S_C & S_A C_C & C_A & C_A d \\ 0 & 0 & 0 & 1 \end{bmatrix} . \tag{1.7}
$$

The inverse of ${}^{B}\mathbf{A}_{P}$, namely $[{}^{B}\mathbf{A}_{P}]^{-1}$, is

$$
[{}^{B}\mathbf{A}_{P}]^{-1} = \begin{bmatrix} [{}^{B}\mathbf{R}_{P}]^{T} & -[{}^{B}\mathbf{R}_{P}]^{T}\,{}^{B}\mathbf{P}_{P} \\ 0 & 1 \end{bmatrix} , \tag{1.8}
$$

we have

$$
{}^{B}\mathbf{R}_{P} = \begin{bmatrix} C_C & -S_C & 0 \\ C_A S_C & C_A C_C & -S_A \\ S_A S_C & S_A C_C & C_A \end{bmatrix} \text{ and } [{}^{B}\mathbf{R}_{P}]^{T} = \begin{bmatrix} C_C & C_A S_C & S_A S_C \\ -S_C & C_A C_C & S_A C_C \\ 0 & -S_A & C_A \end{bmatrix} \tag{1.9}
$$

and hence,

$$
-[{}^{B}\mathbf{R}_{P}]^{T}\,{}^{B}\mathbf{P}_{P} = \begin{bmatrix} C_C & C_A S_C & S_A S_C \\ -S_C & C_A C_C & S_A C_C \\ 0 & -S_A & C_A \end{bmatrix} \begin{bmatrix} X \\ -S_A d + Y \\ C_A d \end{bmatrix} = \begin{bmatrix} -C_C X - C_A S_C Y \\ S_C X - C_A C_C Y \\ -d + S_A Y \end{bmatrix} \tag{1.10}
$$

thus resulting to

$$
[{}^{B}\mathbf{A}_{P}]^{-1} = \begin{bmatrix} C_C & C_A S_C & S_A S_C & -C_C X - C_A S_C Y \\ -S_C & C_A C_C & S_A C_C & S_C X - C_A C_C Y \\ 0 & -S_A & C_A & -d + S_A Y \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.11}
$$

Finally, upon substituting eqs. (1.4) and (1.11) into (1.3), we have

$$
{}^{P}\mathbf{A}_{T} = \begin{bmatrix} C_C & C_A S_C & S_A S_C & -C_C X - C_A S_C Y \\ -S_C & C_A C_C & S_A C_C & S_C X - C_A C_C Y \\ 0 & -S_A & C_A & -d + S_A Y \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d + Z - l \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{P}\mathbf{A}_{T} = \begin{bmatrix} C_C & C_A S_C & S_A S_C & -X C_C - Y C_A S_C + (Z + d - l) S_A S_C \\ -S_A & C_A C_C & S_A C_C & X S_C - Y C_A C_C + (Z + d - l) S_A C_C \\ 0 & -S_A & C_A & Y S_A + (Z + d - l) C_A - d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.12}
$$

The first three rows of the last two columns of eq. (1.12) provide the tool orientation and position, namely the vectors **O** and **P**, respectively. Therefore, the DK model of the Matsuura is:

$$\mathbf{P} = \begin{bmatrix} -XC_C - YC_AS_C + (Z+d-l)S_AS_C \\ XS_C - YC_AC_C + (Z+d-l)S_AC_C \\ YS_A + (Z+d-l)C_A - d \end{bmatrix} \text{ and } \mathbf{O} = \begin{bmatrix} S_AS_C \\ S_AC_C \\ C_A \end{bmatrix} \tag{1.13}$$

1.6 The Denavit-Hartenberg Convention

In general, using the HTM method, for two nonintersecting joint axes of general orientations it would required three translations and three rotations, i.e., six basic HTMs. Denavit and Hartenberg (D-H) (1995) [13] proposed a method which reduces the number of basic HTMs required between two frames of F_i and F_{i-1} to only four HTMs. For the sake of simplicity, the D-H Convention is recalled here based on the summary Angeles's book [1]:

The links are numbered *0,1,...,n*, the *i*th pair being defined as that coupling the (*i*-1)st link with the *i*th link. Hence, the manipulator is assumed to be composed of *n*+1 links and *n* pairs; each of letter can be either R or P, where link 0 is the fixed base, while link n is the EE. Next, a coordinate frame F_i is defined with origin O_i and axes X_i, Y_i and Z_i. This frame is attached to the (*i*-1)st link-**not** to *i*th link- where *i*=1,..., *n*+1. For the first *n* frames, the following rules are given:

1. Z_i is the axis of the *i*th pair. Notice that there are two possibilities of defining the positive direction of this axis, since each pair axis is only a line, not a directed segment. Moreover, the Z_i axis of a prismatic pair can be located arbitrarily, since

only its direction is defined by the axis of this pair.

2. X_i is defined as the common perpendicular to Z_{i-1} and Z_i, directed from the former to the latter, as shown in Figure 1.8a. Notice that if these two axes intersect, the positive direction of X_i is undefined and hence, can be freely assigned. Henceforth, we will follow the *right-hand* rule in this case. This means that if unit vectors $\mathbf{i}_i, \mathbf{k}_{i-1}$, and $\mathbf{k}_i$ are attached to axes X_i, Z_{i-1}, and Z_i, respectively, as indicated in Figure 1.8b, the $\mathbf{i}_i$ is defined as $\mathbf{k}_{i-1} \times \mathbf{k}_i$. Moreover if Z_{i-1} and Z_i are parallel, the location of X_i is undefined. In order to define it uniquely, we specify X_i as passing through the origin of the ((i)-1)st frame, as shown in Figure 1.8c.

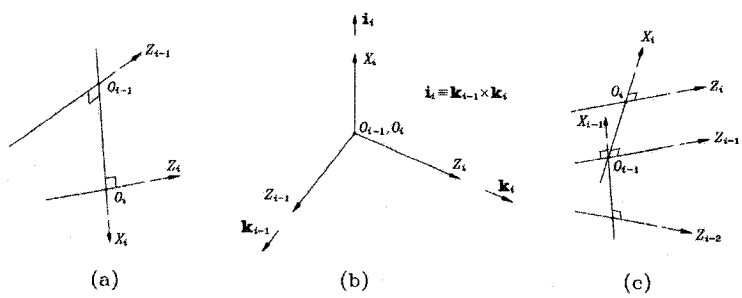


Figure 1.8: Definition of X_i when Z_{i-1} and Z_i: (a) are skew; (b) intersect; and (c) are parallel [1]

3. The distance between Z_i and Z_{i+1} is defined as a_i which is thus nonnegative.

4. The Z_i-coordinate of the intersection O_i' of Z_i with X_{i+1} is denoted by b_i. Since this quantity is coordinate, it can be either positive or negative. Its absolute value is the distance between X_i and X_{i+1}, also called the offset between successive common perpendiculars.

5. The angle between Z_i and Z_{i+1} is defined as α_i and is measured about the positive direction of X_{i+1}. This item is known as the twist angle between successive pair axes.

6. The angle between X_i and X_{i+1} is defined as θ_i and is measured about the positive direction of Z_i.

In summary, a n-axis manipulator is composed of n+1 links and n+1 coordinate frames as F_1, F_2, ..., F_n. These rules are demonstrated with an example. the architecture of PUMA robot is shown in Fig. 1.9. The seven links, numbered from 0 to 6, and seven coordinate frames numbered from 1 to 7. Notice that the last frame is arbitrarily defined, but its origin is placed at specific point of the EE, namely, at the operation point, P. Therefore, according to these rules; each HTM relating to the (i-1)th and ith coordinate

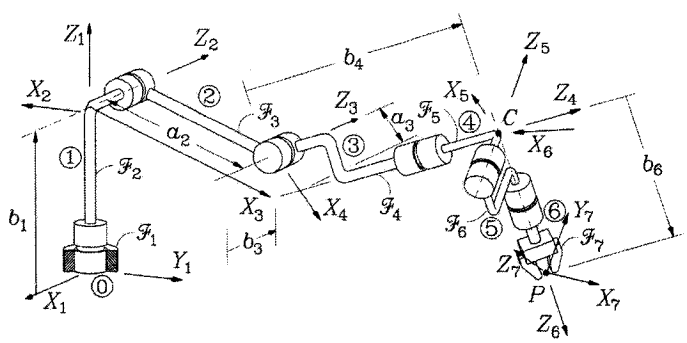


Figure 1.9: Coordinate frames of PUMA robot [1]

frame can be defined from the composition of four basic HTMs as follow:

1. Rotate about the Z_{i-1} axis an angle of θ_i to align the X_{i-1} axis with the X_i axis.
2. Translate along the Z_{i-1} axis a distance of b_i to bring the X_{i-1} and X_i axes into coincidence.
3. Translate along the X_i axis a distance of a_i to bring the two origins as well as the X axis into coincidence.
4. Rotate about the X_i axis an angle of α_i to bring the two coordinate systems into coincidence.

These rotations and translations between two coordinate frames F_{i-1} and F_i is shown in Fig. 1.10. The D-H homogeneous transformation matrix, namely ${}^{i-1}A_i$, expressing the pose of F_i relative to F_{i-1}. Therefore, ${}^{i-1}\mathbf{A}_i$, is given as:

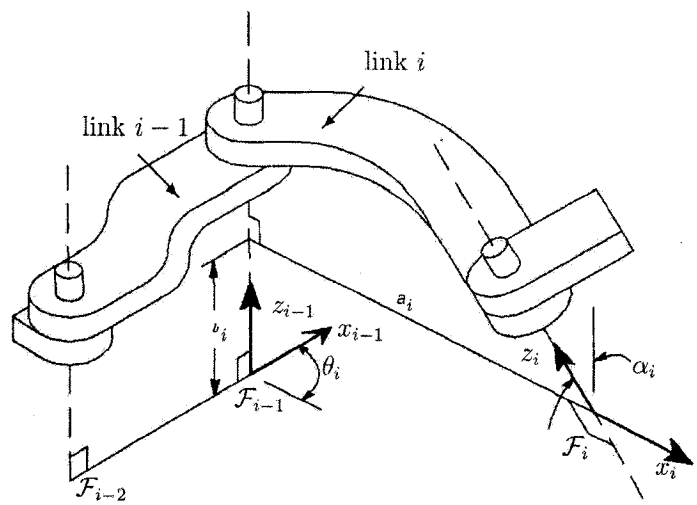


Figure 1.10: D-H parameters [26]

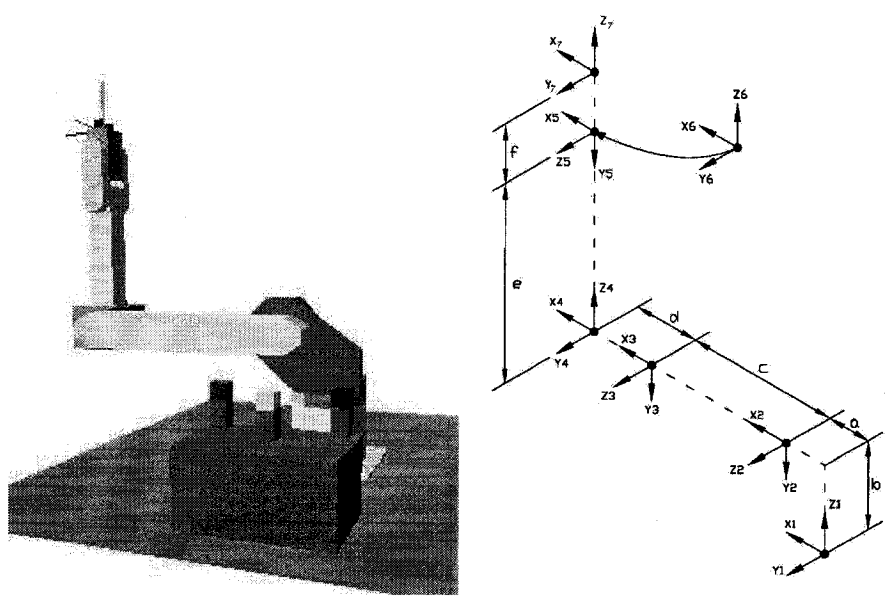
$$
{}^{i-1}\mathbf{A}_i = \mathbf{Rot}(Z,\theta_i)\mathbf{Trans}(Z,b_i)\mathbf{Trans}(X,a_i)\mathbf{Rot}(X,\alpha_i) =
$$

$$
\begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & b_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Hence, from above multiplications, we have the following D-H transformation matrix:

$$
{}^{i-1}\mathbf{A}_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i\sin\theta_i & \sin\alpha_i\sin\theta_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\alpha_i\cos\theta_i & -\sin\alpha_i\cos\theta_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & b_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.14}
$$

The geometry of each link is described by four parameter as: θ_i, a_i, b_i and α_i. These four parameters completely define any revolute and prismatic joints. The a_i *and* α_i are fixed link parameters and determine the structure of the link. θ_i is variable when it is a revolute joint and b_i is variable for prismatic joint. Hence, a n-axis manipulator has n joint variables. The manipulator architecture is fully defined by its D-H parameters, these parameters can be summarize in a table. Figure 1.11a shows the robot Fanuc M16iB in its initial position as an example. Figure 1.11b illustrates its coordinate frames and D-H parameters. The architecture of the Fanuc M16iB is fully defined by its D-H parameters and can be summarized as in table 1.1.



(a) (b)

Figure 1.11: (a) The robot Fanuc M16iB in its initial position; (b) The coordinate frames of the Fanuc M16iB

Table 1.1: D-H parameters of the Fanuc M16iB

i	θ_i°	a_i	b_i	α_i°	$i+1$
1	θ_1	a	b	-90	2
2	θ_2	c	0	0	3
3	θ_3	d	0	90	4
4	θ_4	0	e	-90	5
5	θ_5	0	0	90	6
6	θ_6	0	f	0	7

1.7 D-H Modeling of the Matsuura

In this section, the Direct Kinematics Model of the Matsuura five-axis milling center is found by using the D-H method. Figure 1.7 shows the kinematic chain, while Fig. 1.12 shows the geometry of the Matsuura 5-axis milling center at its initial position, At this configuration, the origin of tool holder frame G is coincident the workpiece frame P on top of center of the table. The table of the machine has two revolute axes C and A, and two prismatic axes X and Y. The column of machine is fixed which has coordinate frame B and tool holder has a vertical prismatic axis Z. When we attach a spherical end mill tool to the tool holder, we need to add the tool length l to Z. For example, when C, A, X, Y are zero then Z=d+l. Figure 1.13 shows the Matsuura with such a tool of length l, when the center of tool is on the top center of the table. If we consider the table of machine tool fixed and the workpiece coordinate frame P parallel to the base coordinate frame, instead of a tree kinematic chain, we will have a serial kinematic chain such as all of machine part move relative to workpiece coordinate frame. We can create D-H parameters directly from workpiece to tool according to table 1.2 as shows in Figure 1.14. The homogeneous transformation matrix between neighboring links F_i to $F_i + 1$

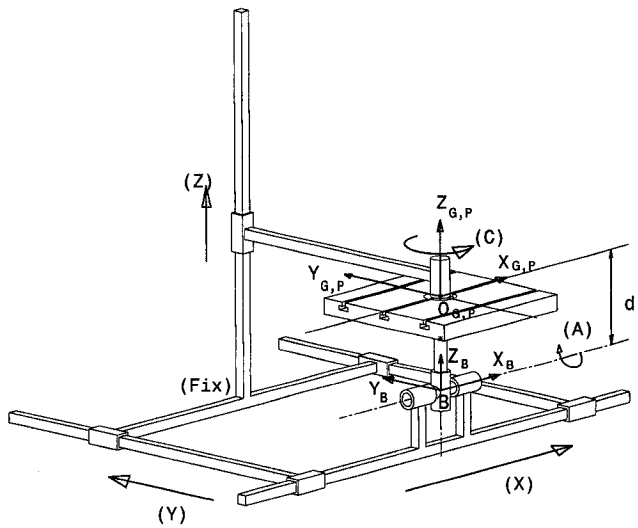


Figure 1.12: Geometry of the Matsuura 5-axis milling center without tool at home position

is calculated from eq (1.14), so ${}^{P}\mathbf{A}_1$ to ${}^{5}\mathbf{A}_T$ can be calculated as follow:

$$
{}^{P}\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{1}\mathbf{A}_2 = \begin{bmatrix} \cos(C) & 0 & \sin(C) & 0 \\ \sin(C) & 0 & -\cos(C) & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{2}\mathbf{A}_3 = \begin{bmatrix} \cos(A) & -\sin(A) & 0 & 0 \\ \sin(A) & \cos(A) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{3}\mathbf{A}_4 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & X & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{4}\mathbf{A}_5 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & Y \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{5}\mathbf{A}_T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & Z+d+l \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

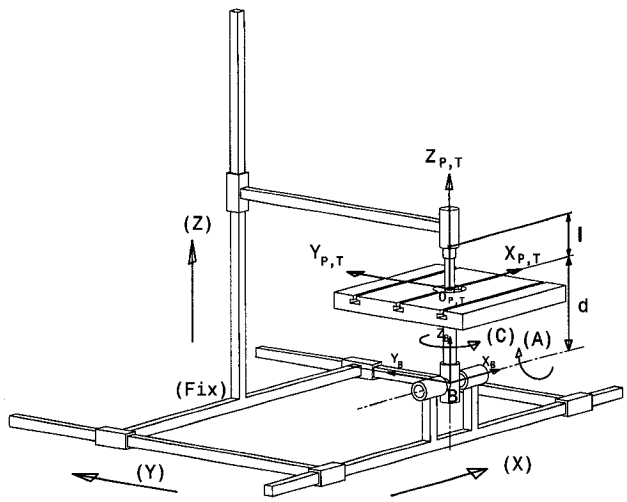


Figure 1.13: Matsuura with a spherical end-mill tool of length ℓ at home position

Table 1.2: D-H parameters from the workpiece to tool frames

i	θ_i°	a_i	b_i	α_i°	$i+1$
P	90	0	0	180	1
1	C	0	d	90	2
2	A	0	0	0	3
3	-90	0	X	90	4
4	90	0	Y	90	5
5	180	0	d+Z-l	0	T

We have:

$$^{P}A_T = ^{P}A_1\,^{1}A_2\,^{2}A_3\,^{4}A_5\,^{5}A_T \tag{1.15}$$

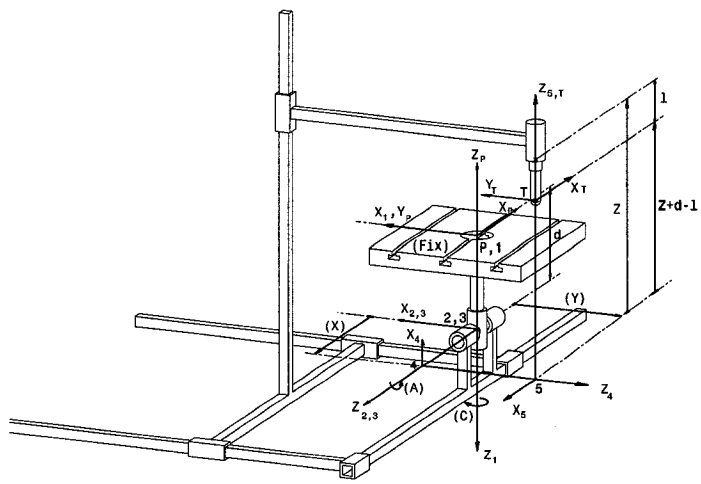


Figure 1.14: Coordinate frames of Matsuura from workpiece to tool frames

So;

$$
{}^{P}A_T = \begin{bmatrix} C_C & C_A S_C & S_A S_C & -XC_C - YC_A S_C + (Z+d-l)S_A S_C \\ -S_A & C_A C_C & S_A C_C & XS_C - YC_A C_C + (Z+d-l)S_A C_C \\ 0 & -S_A & C_A & YS_A + (Z+d-l)C_A - d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.16}
$$

The first three rows of the last two columns of eq. 1.16 provide the tool orientation and position namely O and P, respectively. The DKM of the Matsuura is

$$
P = \begin{bmatrix} -XC_C - YC_A S_C + (Z+d-l)S_A S_C \\ XS_C - YC_A C_C + (Z+d-l)S_A C_C \\ YS_A + (Z+d-l)C_A - d \end{bmatrix} \text{ and } O = \begin{bmatrix} S_A S_C \\ S_A C_C \\ C_A \end{bmatrix} \tag{1.17}
$$

1.8 Different Method of Using D-H Parameters to Procure the DKM of Machine-Tools

As mentioned before, the kinematic chain of machine-tools can be classified as tree-type manipulators. The D-H convention method is suitable only for serial kinematic chain. In the case of machine-tools, the D-H parameters must be created from the table to the tool. This way of modeling machine-tools makes rather difficult to find right direction of motion of each joint. In order to cope with this difficulties, a different way of modeling D-H is shown in Fig. 1.15. The modeling is developed in two steps, one from the base to the table and on other from the base to the tool. The positive direction of joint motion can be the same as the direction of the machine-tools, which is easy to find. According to Fig. 1.15, we have two serial kinematic chain, one in the right hand, and on other in the left hand. Therefore, we have: ${}^{B}\mathbf{A}_T = {}^{B}\mathbf{A}_P \, {}^{P}\mathbf{A}_T$, the position and the orientation of

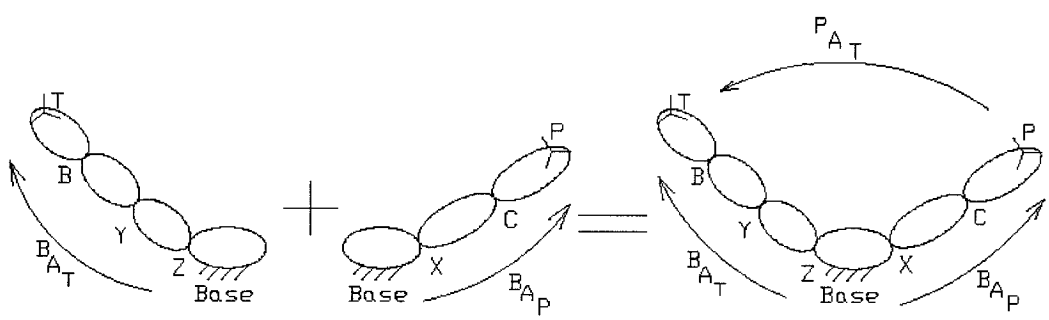


Figure 1.15: The kinematic chain of a five-axis machine-tool

T relative to P is given as ${}^{P}\mathbf{A}_T = [{}^{B}\mathbf{A}_P]^{-1} \, {}^{B}\mathbf{A}_T$ so it is sufficient to develop two D-H parameters one from the base to the tool and other from the base to the table, and put the two series of parameters in one table and calculate the D-H transformation matrices. Notice that the D-H transformation matrices from the base to the part (right hand chain) must be inverse.

As we will see later finding the Jacobian matrix of the five-axis machine-tool using

differential-form makes rather difficult to calculate, we put two D-H parameters for right hand and left hand in kinematic chains in one table and use a matlab program base on algorithm 1.1 to develop the DK model and the Jacobian matrix, which shows advantage of using this method.

In section 1.9 the DK modeling of the Matsuura five-axis milling center is illustrated as an example of using this new method. The kinematic chain of the Matsuura is shown in Fig. 1.16

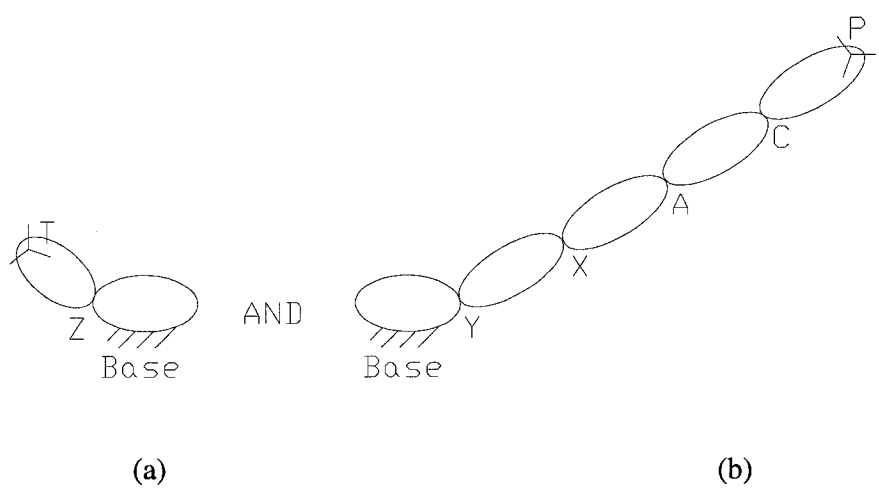


Figure 1.16: The kinematic chain of the Matsuura ; (a) left hand from the base to the Tool; (b) right hand from the base to the workpiece

1.9 The DK Modeling, Using a New Method

1.9.1 D-H Parameters from Base to the Workpiece

According to the D-H parameters, the link frames of the Matsuura from the base to the workpiece are shown in Fig. 1.17. Therefore, the D-H parameter notation from this step

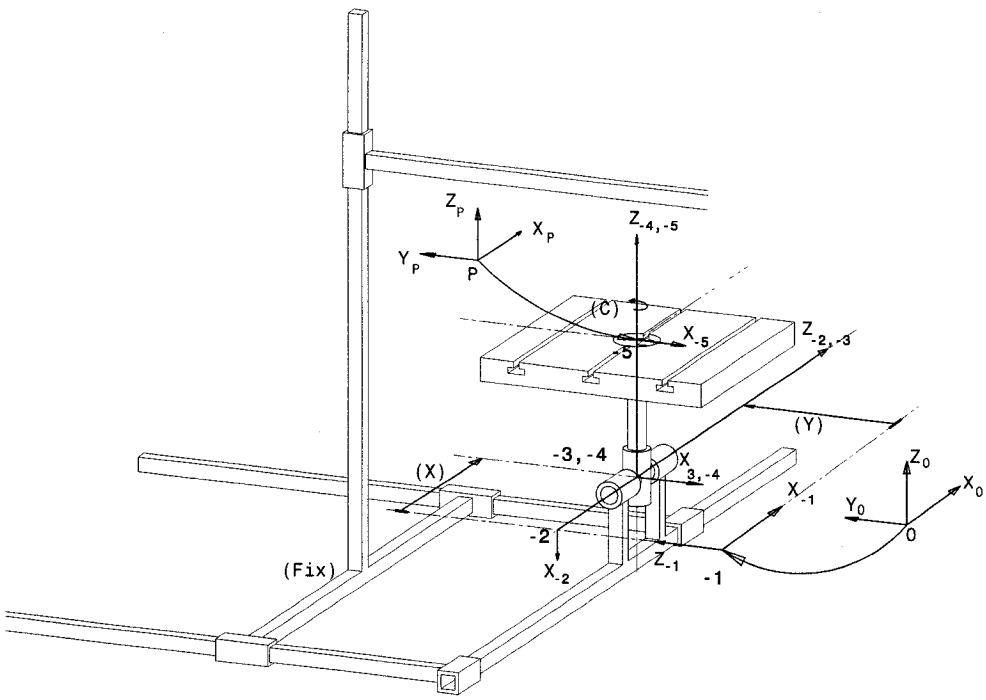


Figure 1.17: Link frames of the Matsuura from the base to the workpiece

is as in table 1.3. Notice that frame 0 is attached to the base and frame numbering are attached in a negative numbering sign are going toward the workpiece.

1.9.2 Define D-H Parameters from Base to Tool

According to the D-H parameters, the link frames of the Matsuura from the the base to the tool shown in Fig. 1.18. Therefore, the D-H parameter notation for this step is as in table 1.4. In summary, the link frames of the Matsuura from the base to workpiece and from the base to the tool can be shown together, while respecting the D-H notations as in Fig. 1.19, and also the two D-H parameter notations, from two steps can be assembled

Table 1.3: D-H parameter notations from the base to the workpiece

i	θ_i°	a_i	b_i	α_i°	$i+1$
-5	90	0	0	0	P
-4	C	0	d	0	-5
-3	A	0	0	90	-4
-2	-90	0	X	0	-3
-1	90	0	Y	90	-2
0	0	0	0	-90	-1

Table 1.4: D-H parameter from the base to the workpiece

i	θ_i°	a_i	b_i	α_i°	$i+1$
0	0	0	d+Z-l	0	T

in one table as in table 1.5. From table 1.5 ${}^{P}\mathbf{A}_{-5}, {}^{-5}\mathbf{A}_{-4} \sim^{0} \mathbf{A}_{T}$ can be found and shown as follow:

$$
{}^{P}\mathbf{A}_{-5} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{-5}\mathbf{A}_{-4} = \begin{bmatrix} C_C & 0 & S_C & 0 \\ -S_C & 0 & C_C & 0 \\ 0 & 1 & 0 & -d \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{-4}\mathbf{A}_{-3} = \begin{bmatrix} C_A & S_A & 0 & 0 \\ 0 & 0 & 1 & 0 \\ S_A & -C_A & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{-3}\mathbf{A}_{-2} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & X \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{-2}\mathbf{A}_{-1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -Y \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{-1}\mathbf{A}_{0} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

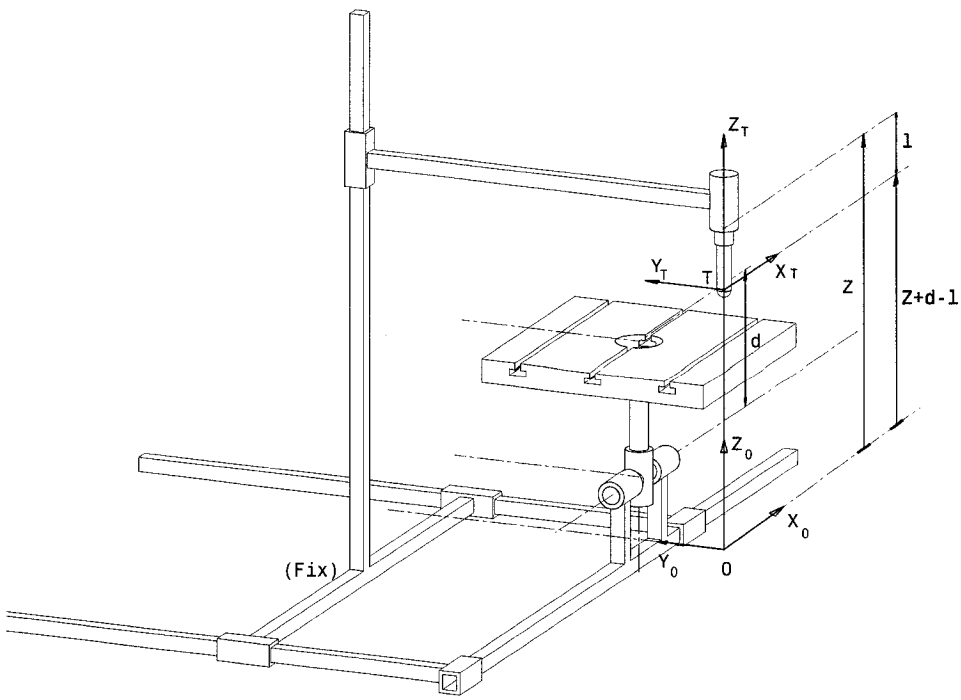


Figure 1.18: Link frames of the Matsuura from the base to the tool

$$
{}^{0}\mathbf{A}_T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & Z+d-l \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

we have:

$$
{}^{P}\mathbf{A}_T = {}^{P}\mathbf{A}_{-5}\,{}^{-5}\mathbf{A}_{-4}\,{}^{-4}\mathbf{A}_{-3}\,{}^{-3}\mathbf{A}_{-2}\,{}^{-2}\mathbf{A}_{-1}\,{}^{-1}\mathbf{A}_0\,{}^{0}\mathbf{A}_T \tag{1.18}
$$

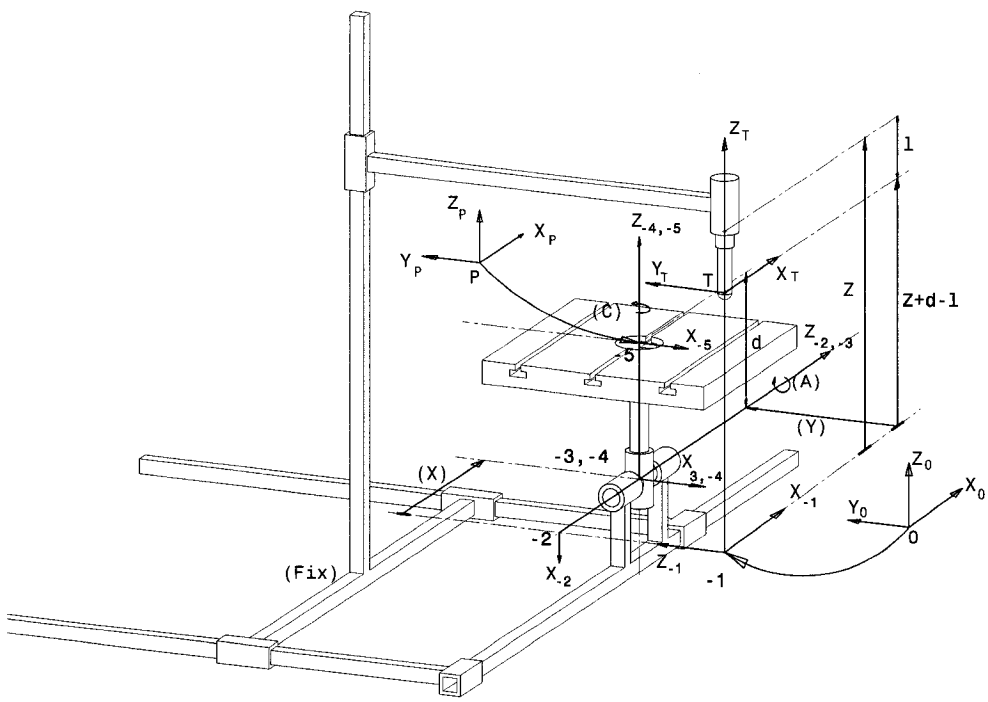


Figure 1.19: Link frames of the Matsuura from the base to the workpiece

So

$$
{}^{P}\mathbf{A}_{T} = \begin{bmatrix} C_C & C_A S_C & S_A S_C & -XC_C - YC_A S_C + (Z+d-l)S_A S_C \\ -S_A & C_A C_C & S_A C_C & XS_C - YC_A C_C + (Z+d-l)S_A C_C \\ 0 & -S_A & C_A & YS_A + (Z+d-l)C_A - d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.19}
$$

Equations 1.19 and 1.16 are identical, showing the identical results of the two methods.

As mentioned before, using our method to develop the D-H parameters of machine-tools simplifies finding the position direction of joint motions. As it has mentioned before another advantage is that it can be easily programmed in a computer. The algorithm

Table 1.5: D-H parameters of the Matsuura (new method)

i	θ_i°	a_i	b_i	α_i°	$i+1$
P	90	0	0	0	-5
-5	C	0	d	0	-4
-4	A	0	0	90	-3
-3	-90	0	X	0	-2
-2	90	0	Y	90	-1
-1	0	0	0	-90	0
0	0	0	d+Z-l	0	T

1.1, using our method will be presented in the following section, which can calculate the DKM and the vector-form of Jacobian matrix of the machine-tools.

1.10 Velocity Modeling

At the displacement level, If X represents the pose and θ represents the joint angles, the equation $X = f(\theta)$ presents the Direct Kinematics Model (DKM) of a manipulator. At the velocity level, the relationship between $\dot{X}$ and $\dot{\theta}$ is:

$$\dot{X} = J(\theta)\dot{\theta}\ , \tag{1.20}$$

where $\dot{X}$ represent the twist of the EE, $\dot{\theta}$ represent the joint velocity along the chain either revolute or prismatic, and $J(\theta)$ is the Jacobian matrix of manipulator. They are defined as

$$\dot{X} \equiv \begin{bmatrix} \omega \\ \dot{p} \end{bmatrix} \in R^6,\ J(\theta) \equiv \frac{\partial f}{\partial \theta} \in R^{6\times n},\ \dot{\theta} \equiv \begin{bmatrix} \dot{\theta}_1 \\ \vdots \\ \dot{\theta}_n \end{bmatrix} \tag{1.21}$$

From eq. (1.21), the twist $\dot{X}$ is a six-dimensional columm which is composed of ω, the angular velocity vector of the origin of F_n in F_0, and $\dot{p}$, the velocity vector of the origin of F_n in F_0. Here, the Jacobian matrix is a function which map the joint rates into the EE twist, and which depends on θ, the joint position. The Jacobian matrix can be determined in two ways differential-form and vector-form.

1.10.1 Differential-Form of the Jacobian matrix

According the eq. (1.21), the Jacobian matrix can be determined as the partial derivatives of the EE pose X with respect to the joint position θ, from DK model of manipulators,

$$J(\theta) = \frac{\partial f}{\partial \theta} = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \cdots & \frac{\partial f_1}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_6}{\partial \theta_6} & \cdots & \frac{\partial f_6}{\partial \theta_n} \end{bmatrix} \in R^{6\times n} \tag{1.22}$$

where n is the number of joints. The DK model of the Matsuura, is given in eq. (1.13). The joint position θ and the tool pose X are defined as follows:

$$\theta \equiv \begin{bmatrix} C \\ A \\ X \\ Y \\ Z \end{bmatrix} \in R^5,\ X \equiv \begin{bmatrix} \mathbf{O} \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} f_O(\theta) \\ f_P(\theta) \end{bmatrix} \in R^6,\ and\ P \in R^3 \tag{1.23}$$

Unfortunately, the total deviation of O with respect to times is not the same as the angular velocity vector, i.e., $\dot{O} \neq \omega$, and hence, we have no other alternative than doing a geometrical interpretation to determine ω.

According to Fig. 1.6, the angular velocity vector of the joint C related to P is $\begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}$.

The angular velocity vector of the joint A is $\begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}$, by multiplying this vector by the rotation matrix $\begin{bmatrix} C_C & S_C & 0 \\ -S_C & C_C & 0 \\ 0 & 0 & 1 \end{bmatrix}$, we have the angular velocity vector of the joint A related to the P $\begin{bmatrix} -C_C \\ S_C \\ 0 \end{bmatrix} = \begin{bmatrix} C_C & S_C & 0 \\ -S_C & C_C & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}$. It is clear that, we have the same angular velocity for three the prismatic joints. Therefore, the angular velocity of the tool expressed in the table frame is given as

$$\omega = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \dot{C} + \begin{bmatrix} -C_C \\ S_C \\ 0 \end{bmatrix} \dot{A} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dot{X} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dot{Y} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dot{Z} \tag{1.24}$$

From eq. (1.22) $\dot{P}$, the velocity of the tool related to P is obtained as follow:
$\dot{P} = \frac{dP}{dt} = \frac{d}{dt} f_P(\theta) = \frac{\partial f_p}{\partial C}\dot{C} + \frac{\partial f_p}{\partial AC}\dot{A} + \frac{\partial f_p}{\partial X}\dot{X} + \frac{\partial f_p}{\partial Y}\dot{Y} + \frac{\partial f_p}{\partial Z}\dot{Z}$

The time derivative of P, given from eq. (1.13), is so,

$$\dot{P} = \begin{bmatrix} S_A C_C (Z - l + d) \\ -S_A S_C (Z - l + d) \\ 0 \end{bmatrix} \dot{C} + \begin{bmatrix} C_A S_C (Z - l + d) \\ C_A C_C (Z - l + d) \\ -S_A (Z - l + d) + C_A Y \end{bmatrix} \dot{A}$$
$$+ \begin{bmatrix} -C_C \\ S_C \\ 0 \end{bmatrix} \dot{X} + \begin{bmatrix} -C_A S_C \\ -C_A C_C \\ S_A \end{bmatrix} \dot{Y} + \begin{bmatrix} -S_A S_C \\ S_A C_C \\ C_A \end{bmatrix} \dot{Z} \qquad (1.25)$$

Therefore, we have

$$\begin{bmatrix} \omega \\ \dot{P} \end{bmatrix} = \begin{bmatrix} 0 & -C_C & 0 & 0 & 0 \\ 0 & S_C & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ S_A C_C (Z - l + d) & C_A S_C (Z - l + d) & -C_C & -C_A S_C & -S_A S_C \\ -S_A S_C (Z - l + d) & C_A C_C (Z - l + d) & S_C & -C_A C_C & S_A C_C \\ 0 & -S_A (Z - l + d) + C_A Y & 0 & S_A & C_A \end{bmatrix} \begin{bmatrix} \dot{C} \\ \dot{A} \\ \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \qquad (1.26)$$

where

$$J(\theta) = \begin{bmatrix} 0 & -C_C & 0 & 0 & 0 \\ 0 & S_C & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ S_A C_C (Z - l + d) & C_A S_C (Z - l + d) & -C_C & -C_A S_C & -S_A S_C \\ -S_A S_C (Z - l + d) & C_A C_C (Z - l + d) & S_C & -C_A C_C & S_A C_C \\ 0 & -S_A (Z - l + d) + C_A Y & 0 & S_A & C_A \end{bmatrix} \qquad (1.27)$$

is the Jacobian matrix of the Matsuura expressed in the workpiece frame.

1.10.2 Vector-Form of the Jacobian Matrix

Whitney (1972) [37] redefined the Jacobian matrix on the vector base. Figure 1.20 shows a general n-axis manipulator, O_i is the origin of ith coordinate frame, attached to the (i-1)st link. A unit vector e_i is associated with each revolute axis. The vector P denotes the position vector of point p attached to the EE and also the vector r_i is defined as the position vector of the origin O_i, with point p, directed from the former to the latter, i.e.,

$$r_i \equiv \mathrm{a}_i + \mathrm{a}_{i+1} + \ldots + \mathrm{a}_n \tag{1.28}$$

and a_i is defined from Fig. 1.10 and eq. (1.14) as:

$$\mathrm{a}_i \equiv \begin{bmatrix} a_i \cos\theta_i \\ a_i \sin\theta_i \\ b_i \end{bmatrix} \tag{1.29}$$

He showed that the Jacobian matrix can be defined as:

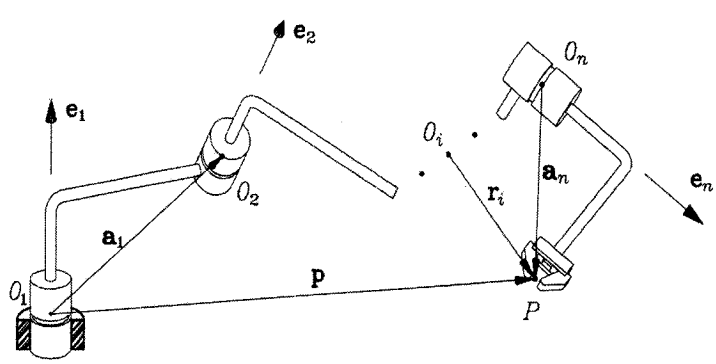


Figure 1.20: General n-axis manipulator [1]

$$J(\theta) = \begin{bmatrix} J_1 & J_2 & \cdots & J_n \end{bmatrix}_{6\times n} \tag{1.30}$$

where $J_i \in R^6$ is ith column of the Jacobian matrix and n is the number manipulator of joints.

J_i is defined as; $\begin{bmatrix} e_i \\ e_i \times r_i \end{bmatrix}$ for the revolute joint and $\begin{bmatrix} 0 \\ e_i \end{bmatrix}$ for the prismatic joint.

Fig. 1.21 illustrates the unit vectors e_i and position vectors r_i of the Matsuura.

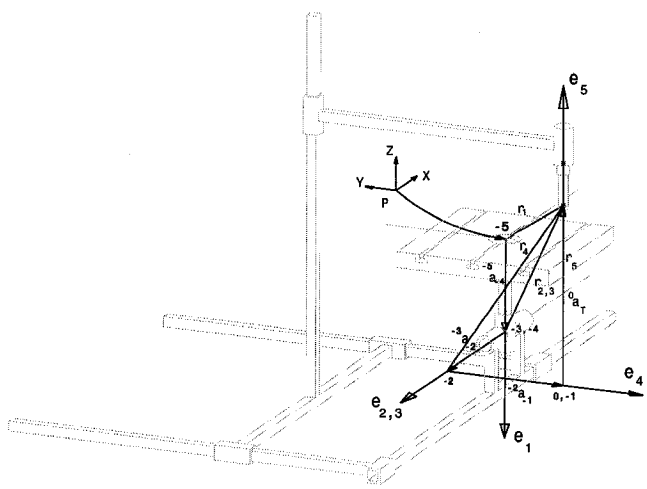


Figure 1.21: The definition of e_i and r_i for the Matsuura

Therefore, the Jacobian matrix is shown as

$$J = \begin{bmatrix} e_1 & e_2 & 0 & 0 & 0 \\ e_1 \times r_1 & e_2 \times r_2 & e_3 \times r_3 & e_4 \times r_4 & e_5 \times r_5 \end{bmatrix}_{6\times 5} \tag{1.31}$$

This matrix is used to find the numerical solution of the IKP and it will be presented in future sections.

1.11 An Algorithm to Find the DKM and the Jacobian Matrix of the Machine-Tools

The solutions of the DK and the Jacobian matrix can be programmed, if the D-H parameters is considered as an input through an algorithm. Here, an algorithm is presented base on new our method of D-H and the vector-form of the Jacobian matrix, i.e.,

Algorithm 1.1 to find the DKM and the Jacobian matrix

Read $\boldsymbol{\theta}_{\mathrm{r}}$, a_{r}, b_{r} and α_{r} from right hand DH parameters

$T = eye(4)$

For m = 1 to size of $\boldsymbol{\theta}_{\mathrm{r}}$ step 1 do

$$^{m}\mathbf{A}r_{m+1} \longleftarrow \begin{bmatrix} Qr_m{}^T & -Qr_m{}^T \mathrm{a}r_m \\ 0 & 1 \end{bmatrix}$$

$Q_m \longleftarrow Qr_m{}^T$

$\mathrm{a}_m \longleftarrow -Qr_m{}^T \mathrm{a}r_m$

$T \longleftarrow T\ ^{m}\mathbf{A}r_{m+1}$

End do

Read $\boldsymbol{\theta}_{\mathrm{l}}$, a_{l}, b_{l} and α_{l} from left hand DH parameters

For n = 1 to size of $\boldsymbol{\theta}_{\mathrm{l}}$ step 1 do

$$^{n}\mathbf{A}l_{n+1} \longleftarrow \begin{bmatrix} Ql_n & \mathrm{a}l_n \\ 0 & 1 \end{bmatrix}$$

$Q_{m+n} \longleftarrow Ql_n$

$\mathrm{a}_{m+n} \longleftarrow \mathrm{a}l_n$

$T \longleftarrow T\ ^{n}\mathbf{A}r_{n+1}$

End do

$^{P}\mathbf{A}_T \longleftarrow$ T; (The DKM)

$R = eye(3)$

Continue of algorithm 1.1

For i = 1 to m + n step 1 do

$R_i \longleftarrow R_{i+1} Q_i$

End do

$k = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}^T$

$j = 1$

For i = 2 to m step 1 do

$e_i \longleftarrow R_j \, k$

$j \longleftarrow j + 1$

End do

For i = 1 to n step 1 do

$e_{i+m} \longleftarrow R_j \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$

$j \longleftarrow j + 1$

End do

$e_1 = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}^T$

$r_{m+n} = a_{m+n}$

For i = m + n − 1 to n step − 1 do

$r_i \longleftarrow \mathrm{a}_i + Q_i \, r_{i+1}$

End do

For i = m + n to 2 step 1 do

$r_i \longleftarrow R_{i-1} \, r_{i+1}$

End do

For i = 1 to m + n step 1 do

$B_i \longleftarrow e_i \times r_i$

End do

Continue of algorithm 1.1

For m = 1 to size of $\boldsymbol{\theta}_{\mathrm{r}}$ step 1 do

if joint is then

$$J_m \longleftarrow \begin{bmatrix} e_{m+1} \\ B_{m+1} \end{bmatrix}$$

else

$$J_m \longleftarrow \begin{bmatrix} 0 \\ e_{m+1} \end{bmatrix}$$

end if

end do

For n = 1 to size of $\boldsymbol{\theta}_{\mathrm{l}}$ step 1 do

if joint is then

$$J_{m+n} \longleftarrow \begin{bmatrix} e_{m+n} \\ B_{m+n} \end{bmatrix}$$

else

$$J_{m+n} \longleftarrow \begin{bmatrix} 0 \\ e_{m+n} \end{bmatrix}$$

end if

end do

$J_{(6)\times(m+n)}$ is the Jacobian matrix

A matlab program base on this algorithm presents in appendix B. It is enough to input D-H parameters according our method in this program. The program calculates automatically the DK model and the Jacobian matrix of the machine-tools which input its D-H parameters.

1.12 Inverse Kinematics Model

The Inverse Kinematic model (IKM) of five-axis machine-tools is to compute all possible sets of joint angles that could be used to attain the given the position and the orientation of the tool. The IKM is not as simple as the DKM, because the kinematic equations are nonlinear for most robot manipulators the closed-form solution (analytical solution) are not easy (or even possible) to solve. The inverse kinematics for five-axis machine-tools most of time can be found as a closed-form solution. Calculation of the closed-form solution is fast and it is also easy to apply the rules to choose one among several possible solutions. Also, the inverse kinematics can be solved as an iterative numerical solution. In the next section, the closed-form solution of the Matsuura is shown, as an example, followed by the iterative numerical solution.

1.12.1 The Closed-Form Solution

The IKM expresses the joint position when position and orientation of the tool coordinate system (T) related to the workpiece coordinate system P is known. From eq. (1.17) of the DKM of the Matsuura, vector O presents the angles A and C, whereas the vector P provides X, Y and Z.

If vector $O = \begin{bmatrix} i \\ j \\ k \end{bmatrix} = \begin{bmatrix} S_A S_C \\ S_A C_C \\ C_A \end{bmatrix}$, we will have:

$$A = \pm \arccos(k) \text{ where } A \neq 0 \,, \qquad (1.32)$$
$$C = \arctan 2(\frac{i}{\sin(A)}, \frac{j}{\sin(A)})$$

If the vector $P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -XC_C - YC_AS_C + (Z+d-l)S_AS_C \\ XS_C - YC_AC_C + (Z+d-l)S_AC_C \\ YS_A + (Z+d-l)C_A - d \end{bmatrix}$, it can be developed as :

$$\begin{bmatrix} -C_C & -C_AS_C & S_AS_C \\ S_C & -C_AC_C & S_AC_C \\ 0 & S_A & C_A \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x - (d-l)S_AS_C \\ y - (d-l)S_AC_C \\ z + d - (d-l)C_A \end{bmatrix}.$$

There is a system of form AX=b where A is an orthogonal matrix and $A^{-1} = A^T$, so $X = A^Tb$. By carrying out and simplifying it, IKM would be:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} -C_Cx + S_Cy \\ -C_AS_Cx - C_AC_Cy + S_A(z+d) \\ S_AS_Cx + S_AC_Cy + C_A(z+d) - (d-l) \end{bmatrix} \tag{1.33}$$

Therefore, equations (1.32) and (1.33) represent the IKM of the Matsuura.

1.12.2 Iterative Numerical Solution

The iterative numerical solution is to find the value of the vector of joint variables $\theta(t_{k+1})$ while the value of $\theta(t_k)$ is given. The Algorithm 1.2 can be used to find this value.

Algorithm 1.2 [1]

$\boldsymbol{\theta} \longleftarrow \boldsymbol{\theta}(t_k)$;

1 find correction $\Delta\boldsymbol{\theta}$;

if $\|\Delta\boldsymbol{\theta}\| \leq \epsilon$, then stop;

else

$\boldsymbol{\theta} \longleftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$

go to 1

Various procedures are available to find the correction $\boldsymbol{\Delta\theta}$ of Algorithm 1.2. The convenient one is based on the Newton-Gauss or Newton-Raphson method (Dahlquis and Björck) (1974). In 1969, Whitney proposed a differential kinematic method to solve the IKM of the manipulators and human prostheses [36]. This method use the relationship between the End-Effector (EE) velocity and the joint velocity, which is represented by a linear algebraic equation. The coefficient of the linear equation is the Jacobian matrix, which is a nonlinear function of the joint angles. Whitney named this method "Resolved-Motion-Rate" (RMR). If X represents the position and the orientation and θ represents the joint angles, the equation $X = f(\theta)$ presents the DKM of manipulator. At the velocity level, the eq. (1.20) shows the relationship between $\dot{X}$ (the twist array of the EE) and $\dot{\theta}$ (the joint velocity). Under a common short time interval, the EE and joint velocities can be approximated as EE and joint displacements, i.e. $\dot{X} \equiv \triangle X$ and $\dot{\theta} \equiv \triangle\theta$ and hence, eq. (1.20) can be rewritten as:

$$\triangle X = J(\theta)\triangle\theta \tag{1.34}$$

Therefore, the joint displacements $\triangle\theta$ can be found from $\triangle X$; the small displacement of the EE for the actual position $X_i = f(\theta_i)$:

$$\triangle\theta = J(\theta)^{-1}\triangle X \tag{1.35}$$

The $\triangle X$ is defined as

$$\triangle X \equiv \begin{bmatrix} Q\text{vect}\,(\triangle Q) \\ \triangle p \end{bmatrix} \tag{1.36}$$

and $\triangle p = p_d - P$, $\triangle Q = Q^T Q_d$ where p, Q and p_d, Q_d are, the actual and the desired position vector and rotation matrix of the EE. The iterative numerical solution consists essentially of eqs. (1.35) and (1.36), as first proposed by Pieper (1968) [31].

The function vect $(\triangle\ Q)$ represents the axial vector of a 3×3 rotation matrix $\triangle Q$, and is calculated as follow:

$$\text{vect}(\triangle Q) \equiv \frac{1}{2}\begin{bmatrix} q_{32} - q_{23} \\ q_{13} - q_{31} \\ q_{21} - q_{12} \end{bmatrix} \tag{1.37}$$

Thus, the Resolved-Motion-Rate (RMR) method can be illustrated as the Algorithm 1.3.

Algorithm 1.3 the Resolved-Motion-Rate method

1 $\{\mathbf{p}_d, \mathbf{Q}_d\} \longleftarrow$ desired pose of the EE

$\boldsymbol{\theta} \longleftarrow$ initial guess

$\{\mathbf{p}, \mathbf{Q}\} \longleftarrow \text{f}(\boldsymbol{\theta})$

$\boldsymbol{\Delta}\mathbf{Q} \longleftarrow \mathbf{Q}^T\mathbf{Q}_d$

$\boldsymbol{\Delta}\mathbf{p} \longleftarrow \mathbf{p}_d - \mathbf{p}$

2 $\boldsymbol{\Delta\theta} \longleftarrow \mathbf{J}^{-1}\boldsymbol{\Delta}\mathbf{t}$

if $\|\boldsymbol{\Delta\theta}\| \leq \epsilon$, then stop;

else

$\boldsymbol{\theta} \longleftarrow \boldsymbol{\theta} + \boldsymbol{\Delta\theta}$

$\boldsymbol{\Delta}\mathbf{Q} \longleftarrow \mathbf{Q}^T\mathbf{Q}_d$

$\boldsymbol{\Delta}\mathbf{p} \longleftarrow \mathbf{p}_d - \mathbf{p}$

go to 1

There are many different modified version of this Algorithm 1.3 in the literature. For example, a damping factor $0 < \rho < 1$ is often used in the step $\theta \leftarrow \theta + \triangle\theta$, which becomes $\theta \leftarrow \theta + \rho\triangle\theta$ in order to avoid overshoot of $\triangle\theta$, when the procedure passes closed to a singularity of $J(\theta)$. The J^{-1} is the inverse of the Jacobian matrix, where, $JJ^{-1} = J^{-1}J = I$. The existence of J^{-1} is depended on the condition number of J. The

condition number of J, namely $k(J)$, is defined as the ratio of the largest singular value σ_l of J to the smallest one, σ_s, i.e.

$$k(J) \equiv \frac{\sigma_l}{\sigma_s} \quad (k(J) \geq 1). \tag{1.38}$$

The $k(J)$ can be used as a measure of the amplification of error in differential kinematic method (Salisbury and Craig 1982) [32]. The $k(J)$ can attain values from 1 to infinity. The singular matrices have a smallest singular value that vanishes, and hence, their condition number is infinity. In other word a Jacobian matrix with large condition number is near a singularity, which may involve large computational errors.

CHAPTER 2

REDUNDANCY AND OPTIMIZATION

2.1 Redundancy

If the number of joints of a manipulator (n) is greater than the dimension of the manipulation variable (operation space) of EE (m), i.e. $n > m$, the manipulator is a redundant manipulator. From eq. (1.20), in the frame work of differential kinematics we have:

$$t = J(\theta)\dot{\theta} \tag{2.1}$$

where t is the twist array of EE. Eq. (2.1) is a linear algebraic system with $J(\theta)$ defined as a full-rank $m \times n$ matrix, while $\dot{\theta}$ and t are n- and m- dimensional columns, respectively. In the case of redundant manipulator, as it is mentioned before, we have $n > m$. Therefore, $J(\theta)$ has fewer rows than columns or has a rank less than the number of rows, and hence, the solution of the linear algebraic system of eq. (2.1) is considered as an under-determined case, which has an infinite number of solutions.

2.2 Optimization and Redundancy Resolution

In the field of robotics, the subject of redundancy resolution has been largely studied at both joint position level and joint-rate level. In this study, the joint-rate formulation is adopted because the problem of redundancy resolution is linear at that level. Yashi and ozgoren (1984) [38] proposed the solution by using minimal motion of joints as optimization criteria, with lagrange multiplier method, and they linearized the displacement

equations. Ballieul (1985) [5] proposed a form of the Jacobian matrix. Angeles, Anderson and Gosselin (1987) [2] resolved a system of joints displacement with an orthogonal decomposition algorithm for a non linear problem. They also decomposed the solution into two orthogonal components, one is projected on the null-space of the matrix and another one on the orthogonal space of the same matrix. Angeles and Mathur (1989) [4] resolved the inverse kinematic problem for joint velocity and for a cyclical movement. Most methods worked at the joint-rate level and used the Moore-Penrose Generalize Inverse of the Jacobian matrix. These methods will be presented in more details in the following sections.

2.2.1 Generalized Inverse and Pseudoinverse

For $A \in R^{m\times n}$ and $X \in R^{n\times m}$, the following equations are used to define a generalized inverse, a reflexive generalized inverse, and a pseudoinverse of A (Bollinon and Odell 1971) [27]:

$$AXA = A \tag{2.2}$$

$$XAX = X \tag{2.3}$$

$$(AX)^T = AX \tag{2.4}$$

$$(XA)^T = XA \tag{2.5}$$

Equations (2.2) through (2.5) are called the Penrose conditions (Penrose 1955).

- **Generalized Inverse**

  A generalized inverse of matrix $A \in R^{m\times n}$ is a matrix $X = A^- \in R^{n\times m}$ satisfying eq. (2.2).

- **Reflexive Generalized Inverse**

  A reflexive generalized inverse of matrix $A \in R^{m\times n}$ is a matrix $X = A_r^- \in R^{n\times m}$

satisfying eqs. (2.2) and (2.3).

- **Pseudoinverse**

  A pseudoinverse of matrix $A \in R^{m \times n}$ is a matrix $X = A^{+} \in R^{n \times m}$ satisfying eqs. (2.2) through (2.5).

"A pseudoinverse is somtimes called the Moore-Penrose inverse after the pioneer works by Moore (1920,1935) and Penrose (1955)" (from Nakamura's book page 41 and 42 [27])

The generalized inverse is used for the solution of the linear algebraic system of equations, i.e.,

$$Ax = b\,, \tag{2.6}$$

with A defined as a full-rank $m \times n$ matrix, while x and b are n- and m- dimensional vectors, respectively. Thus, $m \neq n$, we have the following two case:

- over-determined: $m > n$; and
- under-determined $m < n$

The over-determined case does not admit a solution, unless vector b happens to lie in the range of A. Equation (2.6) has fewer unknowns than equations. In this case, the least-square solution is used to find the unknowns. The error in the approximation of eq. (2.6), namely e, is defined as

$$e = b - Ax \tag{2.7}$$

The solution of x minimizes a norm of e, e.g. $\| e \|_2$. The solution is given as

$$x = A^{+}b = (A^{T}A)^{-1}A^{T}b \tag{2.8}$$

where

$$A^{+} \equiv (A^{T}A)^{-1}A^{T} \tag{2.9}$$

is a left generalized inverse of matrix A.

The under-determined case, as mentioned before, has an infinite number of solutions. Thus, any matrix of the form

$$x = A^{+}b + (I - A^{+}A)y \tag{2.10}$$

where $y \in R^{n\times 1}$ is an arbitrary vector, is a solution of eq (2.6). Here, the A^{+} is the right generalized inverse of matrix A, which is defined as

$$A^{+} \equiv A^{T}(AA^{T})^{-1} \tag{2.11}$$

If $y \equiv 0$, the smallest Euclidean norm, usually called the minimum-norm solution is computed. Moreover, the $(I - A^{+}A)$ in eq. (2.10) is an orthogonal complement of A projecting y onto the null-space of matrix A. Thus, we have:

$$\begin{aligned} (I - A^{+}A)A^{+} &= 0 \\ (I - A^{+}A)(I - A^{+}A) &= I - A^{+}A \\ (I - A^{+}A)^{+} &= (I - A^{+}A) \end{aligned} \tag{2.12}$$

Most Redundancy Resolution algorithms used the solution of eq. (2.10) to find joint-rate vector. Therefore eq. (2.1) becomes

$$\dot{\theta} = J^{+}t, \tag{2.13}$$

where, $J^{+} = J^{T}(JJ^{T})^{-1}$, is the right generalized inverse of J.

Klein and Huang (1983) [22] showed that this solution can lead to noncyclic motions in joint space. An orthogonal complement of J projecting h onto the null-space of J, i.e.,

$(I - J^+J)h$ can be added to the minimum-norm solution, i.e., eq. (2.13). Therefore, this non-minimum-norm general inverse kinematic solution can be written as:

$$\dot{\theta} = J^+t + (I - J^+J)h, \tag{2.14}$$

where, the $(I - J^+J)h$ is an homogenous solution of eq. (2.13) and h is an arbitrary vector to be a secondary task. Equation (2.14) is widely used by many researchers such as Angeles (1998) [3] and Siciliano (1992) [33] in order to solve redundant tasks. Here, vector h is an optimized performance criterion. Different value of h will give different performances.

Liégeois (1997) [18] proposed to minimize the index of position-dependent scalar performance, i.e., $p(\theta)$ and developed the *Gradient projection method* and defined takes its gradient as vector h:

$$h = k\frac{\partial p}{\partial \theta} = k\begin{bmatrix} \frac{\partial p}{\partial \theta_1} & \frac{\partial p}{\partial \theta_2} & \cdots & \frac{\partial p}{\partial \theta_n} \end{bmatrix}^T \tag{2.15}$$

where k is a positive scalar coefficient. Liégeois introduced a p that helps the joint-limits avoidance as:

$$p = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\theta_i - \theta_i^{mid}}{\theta_i^{mid} - \theta_i^{max}}\right)^2 \tag{2.16}$$

where $\theta_i^{mid} = (\theta_i^{min} + \theta_i^{max})/2$, θ_i^{min} and θ_i^{max} are, respectively, lower and upper joint limits. Yoshikawa (1984) [38] proposed a scaler value μ to measure the manipulability, as:

$$\mu = \sqrt{det(JJ^T)} \tag{2.17}$$

in order to avoid singularities. He also introduced a performance index for obstacle avoidance, namely,

$$p = \frac{1}{2}(\theta - \theta_r)^T H(\theta - \theta_r) \tag{2.18}$$

where H is a diagonal matrix and θ_r is a given arm posture.

Maciejewski and Klein (1985) [23] also resolved the redundancy to avoid obstacles. They named the nearest manipulator point to the obstacles "obstacle-avoidance point". They developed the equation that avoids the obstacle as

$$J_o\dot{\theta} = \dot{p}_o \tag{2.19}$$

where J_o and $\dot{p}_o$ are respectively the Jacobian matrix and displacement velocity vector, in the obstacle-avoidance point. The solution uses the eq. (2.14) while h is defined as:

$$h = [J_o(I - J^+J)]^+(\dot{p}_o - J_oJ^+\dot{p}) \tag{2.20}$$

putting eq. (2.20) as h in eq. (2.14) and simplify, the result gives us the solution as

$$\dot{\theta} = J^+\dot{p} + [J_o(I - J^+J)]^+(\dot{p}_o - J_oJ^+\dot{p}) \tag{2.21}$$

Klein (1984) [21] resolved redundancy in order to get a maximal dexterity. He suggested to use a performance μ_r, which is the smallest singular value of the Jacobian matrix, then, the value defined as singularities. So, μ_r can be used as maximal joint velocity;

$$\| \dot{\theta} \| = \frac{1}{\mu_r} \| \dot{p} \| \tag{2.22}$$

However, a problem is raised because the norm of the Cartesian velocity vector can be physically complicated to define.

Yoshikawa's manipulability index has been used extensively. However, the determinant cannot be used to measure how close a matrix is to singularity, as Golub and Van (1989) [15] pointed out. Therefore, Kosuge and Furuta (1985) [24] suggested to take the reciprocal of the condition number of the Jacobian matrix as a controllability measure.

Hanafusa, Yoshikawa and Nakamura (1987) [29] decomposed a task into several sub-

tasks and assigned priorities to each subtask. They considered these in the inverse kinematics model, then, each sub task is solved by using the residual of the degree of freedom from the former resolution.

The damped least-square methods introduce a damping factor ρ in the solution, i.e.,

$$\dot{\theta} = J_{\tau}^{+}\dot{p} \tag{2.23}$$

where

$$J_{\tau}^{+} = J^{T}(JJ^{T} + \rho I)^{-1} \tag{2.24}$$

Nakamura and Hanafusa (1987) [28] named this inverse of in eq. (2.24), the "SR-inverse" (Singularity-Robust-Inverse). They suggested an automatic adjustment of the damping factor, which affects the manipulator performance near of singularities, as forms

$$\rho = \begin{cases} \rho_0(1 - \mu/\mu_0)^2, & \text{if } \mu < \mu_0; \\ 0, & \text{otherwise,} \end{cases} \tag{2.25}$$

where μ is the manipulability, defined in eq. (2.17), μ_0 represents the near boundary of the singular points and ρ_0 is a constant which represents the scale at the singularity.

Kelmar and Khosla (1990) [20] proposed to use the ratio of μ between two iterations, instead of determining the μ and testing the $\mu < \mu_0$. The idea was based on while the manipulator approaches a singularity, the ratio will drop dramatically, and they suggested the damping factor as following;

$$\rho = \begin{cases} \rho_0(1 - \mu^{i+1}/\mu^{i})^2, & \text{if } \mu^{i+1}/\mu^{i} < k; \\ 0, & \text{otherwise,} \end{cases} \tag{2.26}$$

where k is the near boundary of the singular points as μ_0 that they determined experimentally.

2.2.2 Weighted Generalized Inverse Method

The weighted generalized inverse solution of eq. (2.1) can be express as follow;

$$\dot{\theta} = J_w^+ \dot{p} \tag{2.27}$$

where J_w^+ is;

$$J_w^+ = W^{-1} J^T (J W^{-1} J^T)^{-1} \tag{2.28}$$

and W is a positive definite weighting matrix. Chan and Dubey (1993) [12] compared the weighted generalized inverse method with the gradient projection method in the case of joint limits avoidance problem, and they found the weighted generalized inverse method reached more accessible joint trajectory than gradient projection method. Whitney (1969) [36] applied priorities through the weighted matrix. Konstantinov, Markov and Nechev (1981) introduced a method which avoids exceeding the joint limits. Park ,Chung and Youm (1996) [30] used the weight not only in the generalized inverse, but also in the Jacobian matrix and the joint velocity and they named them, weighted Jacobian and weighted joint velocity. Therefore, they solved eq. (2.1) as

$$J_w \dot{\theta} = t, \tag{2.29}$$

where J_w and $\dot{\theta}$ are the weighted Jacobian and the weighted joint velocity.

2.2.3 Householder Reflection Method

In order to solve eq. (2.14), Angeles, Arenson and Slutski (1998) [3] introduced the Householder Reflection procedure instead of using a direct computation of the generalized inverse of J. Under the assumption of full rankness of J, eq. (2.14) can be rewritten

as

$$\dot{\theta} = w + h, \tag{2.30}$$

where w is the minimum norm solution of the following linear algebraic system

$$Jw = r \tag{2.31}$$

while r is defined as

$$r \equiv t - Jh \tag{2.32}$$

So, eq. (2.31) can be rewritten as

$$Jw = t - Jh \tag{2.33}$$

an Householder orthogonal matrix H that triangularizes J^T in the form of

$$HJ^T = \begin{bmatrix} U \\ O \end{bmatrix} \tag{2.34}$$

is used for solving eq. (2.33), where U is a upper-triangular matrix and O is the zero matrix. Inserting the identity matrix $H^T H$ into the eq. (2.33) as follow

$$JH^T Hw = t - Jh, \tag{2.35}$$

yields;

$$\begin{bmatrix} H^T & O^T \end{bmatrix} Hw = t - Jh \tag{2.36}$$

because of

$$JH^T = (HJ^T)^T = \begin{bmatrix} H^T & O^T \end{bmatrix} \tag{2.37}$$

Therefore, w can be defined as follows

$$w = H^T \begin{bmatrix} (U^T)^{-1}(t - Jh) \\ O \end{bmatrix} \tag{2.38}$$

Putting eq. (2.38) into eq. (2.30) yields

$$\dot{\theta} = H^T \begin{bmatrix} (U^T)^{-1}(t - Jh) \\ O \end{bmatrix} + h \tag{2.39}$$

Therefore, eq. 2.39 can be rewritten as

$$\dot{\theta} = H^T y + h \tag{2.40}$$

where $y \equiv \begin{bmatrix} (U^T)^{-1}(t - Jh) \\ O \end{bmatrix}$.

Using this solution avoids a direct computation of the generalized inverse of the Jacobian matrix, therefore, the squaring of the condition number of the Jacobian matrix is avoided and the round-off error of the algorithm is not amplified.

2.2.4 Elimination Method

Baron (1999) [6] proposed an elimination method to optimize the post processing of the surface milling operation using a ball-end mill tool with a five-axis machine-tools. The surface is milled with a ball-end mill tool requiring only three dofs, and hence, can be realized with a three-axis machine-tools. However, if the task is realized with a five-axis machine-tools, the orientation of the tool can be chosen freely. Therefore, the angular velocity can be eliminated from the twist. so, eq. (2.14) can be rewritten as

$$\dot{\theta} = B^+ \dot{p} + (I - B^+ B)h, \tag{2.41}$$

where B is the lower part of the Jacobian matrix, i.e., a 3×5 matrix and $\dot{p} \in R^{3\times 1}$ is the displacement velocity.

2.2.5 Virtual Joint Method

Baron (2000) [7] proposed a joint limits avoidance strategy for arc-welding robots. A virtual joint around the symmetry axis of the electrode is added in order to obtain an under-determined linear algebraic system with at least one dof of redundancy. Equation (2.14) can be rewritten as

$$\dot{\theta}_v = J_v^+ t + (I - J_v^+ J_v)h, \tag{2.42}$$

where J_v is an augmented Jacobian matrix by the virtual joint-rate $\dot{\theta}_{n+1}$ and $\dot{\theta}_v \equiv \begin{bmatrix} \dot{\theta} \\ \dot{\theta}_{n+1} \end{bmatrix}$.

2.2.6 Twist Decomposition Method

Huo and Baron (2005) [16] decomposed the twist in eq. (2.13) into two orthogonal subspace, one lying into the task subspace and another lying into the redundant subspace. They found a new solution for the inverse kinematics of functionally redundant serial manipulators, then they applied this method to arc welding task with the PUMA 500 manipulator. The idea is, in the eq. (2.13), t (Δt) can be decomposed into two orthogonal part

$$t = t_\tau + t_{\tau\perp} = Tt + (1 - T)t \tag{2.43}$$

where T is a twist projector. Therefore, eq. (2.13) can be rewritten as

$$\Delta\theta = (J^+T)\Delta t + J^+(1 - T)\Delta t \tag{2.44}$$

The first part of eq. (2.44) represents the task displacement and the second part represents the redundant displacement. The Δt can be considered as Jh, i.e., $\Delta t \equiv Jh$, where h is an arbitrary vector of joint spaces allowing to satisfy a secondary task. Hence, eq. (2.44) can be rewritten as follow

$$\Delta\theta = (J^{+}T)\Delta t + J^{+}(1 - T)Jh \tag{2.45}$$

In general arc welding operations, the electrode has a symmetry axis. The electrode can be rotated around this axis without affecting the task. This axis describes the geometry of the functional redundancy. The unit vector e denote the orientation of the symmetry axis along the electrode. The twist projector for a general arc welding task is defined as

$$T \equiv \begin{bmatrix} (1 - ee^{T}) & 0 \\ 0 & 1 \end{bmatrix}, \quad T^{\perp} \equiv 1 - T = \begin{bmatrix} ee^{T} & 0 \\ 0 & 0 \end{bmatrix} \tag{2.46}$$

2.3 Secondary Task for Redundant Machining Operation

The definition of a secondary task in the case of redundant machining depends on the operation. In this study, the case of a surface milling operation with a ball-end tool is considered as a redundant task requiring only the three dofs of position. The tool orientation could be chosen with criterion optimization. Therefore, the first three rows of the Jacobian matrix can be eliminated in order to keep only the last three rows concerning the translational velocity that maps the joint-rates $\dot{\theta}$ into the tool velocity vector $\dot{p}$ as $J\dot{\theta} = \dot{p}$, where the solution is well-know to be

$$\dot{\theta} = J^{+}\dot{P} + (I - J^{+}J)h, \tag{2.47}$$

with J^{+} being the right generalized inverse of J, i.e., $J^{+} = (J^{T}J)^{-1}J^{T}$, and h a secondary task that is projected onto the null-space of J. The first part of eq. (2.47) presents

the minimum-norm solution, while the second part of eq. (2.47) presents the homogenous solution. Clearly, the first part of eq. (2.47) does not control the tool orientation, and hence, h in the second part must include at least a term of cutting tool orientation, i.e., the normal of the milling surface O_f. With this term, it is possible to optimize the tool orientation O, such that to approach O_f with some acceptable angle error. In fact, the axis of the tool at any point of the tool path must be within a cone of axis of desired maximum error angle. Here, the first part of eq. (2.47) cannot control the joint limits of the milling machine, therefore, keeping θ as close as possible to the mid-joint position $(\overline{\theta})$ can be required. The secondary task h can be chosen as ∇z, where z is the following objective function

$$z = \frac{1}{2}(\theta - \overline{\theta})^T W^T W(\theta - \overline{\theta}) + W_0(1 - O^T \cdot O_f) \tag{2.48}$$

where W is a diagonal positive definite weighting matrix and $\overline{\theta}$ is the mid-joint position, while, O_f and W_0 are the unit vector along the desired final tool orientation and positive weighting, respectively. The objective is minimizing z so h could be $-\nabla z$ hence,

$$h = -W(\theta - \overline{\theta}) + W_0 \frac{d}{d\theta}(O^T \cdot O_f) \tag{2.49}$$

2.4 Conclusion

In this chapter, we have review the literature on the inverse kinematics of serial manipulator in the case of redundancy. A special form has been placed on the solution of these problem at the velocity level, where an iterative numerical procedure is considered.
In the next chapter, we will transpose these solution procedure into the post-processor of a CNC five-axis milling center for surfacing milling operations.

CHAPTER 3

OPTIMAL POST-PROCESSING OF MACHINE-TOOLS

As it was mentioned before, the last stage of CAD/CAM systems is to convert CL data to machine code, the interface that links the CAM systems and CNC machines, which is called post-processor. In this chapter, an optimal post-processor for five-axis CNC milling machines is studied.

3.1 Cutter Location File (CL-File)

In order to calculate the milling of a shape with a tool in a CAM system, it is often necessary to approximate a space curve by a few thousands straight line segments. To accomplish this, the CAM system must discrete the curve into thousands of small straight lines within the machining tolerance of the tool path and for each of these end-points a cutter-radial offset must be calculated to determine the cutter location. These locations are saved into the recalled cutter location file namely CL-file under an ISO standard. One of these standards is the APT format (Automatically Programmed Tools). In 1955, a prototype APT system was coded for the Whirlwind computer at MIT to demonstrate its feasibility.

Figure 3.1 shows an example of tool path of five-axis machining with a ball-end mill tool from CAM system. The vectors, $P = [x\ y\ z]$ and $N = [i\ j\ k]$ are used to define the position and the orientation of center of ball-end mill tool related to the workpiece coordinate system. The APT of this CL data can be shown as:

"GOTO / x , y , z , i , j , k" in an APT-file, where, GOTO is a standard command in APT system, the three first columns represent P and last three columns represent N.

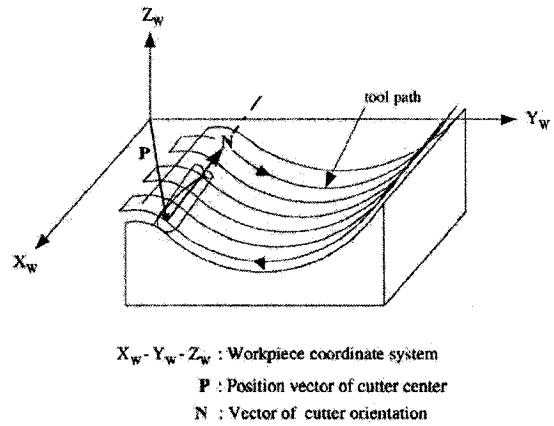


Figure 3.1: Tool path of five-axis milling [17]

It is worth mentioning that each line starting with a GOTO in an APT-file represents the pose of tool related to the workpiece coordinate system, which must be translated trough the inverse kinematics by the post-processor module into "G01 X Y Z A C" in G-code system. Notice that x, y, z, A and C are the joint position of the milling machine.

3.2 Linearization

When CL data are generated by the CAM system, it is assumed that the tool path between two CL points is a straight line relative to the workpiece. However, due to the rotary axes of the machine, the tool path between two blocks in the NC program will be non-linear relative to the workpiece, reducing the accuracy of the tool path. Linearization of the tool path in the post-processing will solve this problem. In an unlinearized NC program each motion command is a direct transformation of one motion command in the CL data file, i.e., a G01 command is translated from each GOTO command. The number of lines in the NC program is approximately equal to the number of lines in the CL file. Linearization of the tool path is performed in the post-processor by interpolat-

ing new CL data point along the ideal tool path and thereby adding new blocks to the NC program. One command in the CL file may result in several lines in the NC file, so the size of the NC program increases. The final tool path will consist of position that originate directly from CL data file, and new positions that are generated by CL data interpolation in the post-processor. Linearization will not provide perfect motion of the tool, but allow deviation from the ideal tool path within an acceptable level. Linearization algorithms must use calculations of both the forward and the inverse kinematics to calculate deviations from ideal tool path and to interpolate new CL data. A recursive method for linearization, taking into consideration both the position and the orientation deviation of the tool, is described by K. Sorby (1999) [34].

3.3 Algorithm for the Optimal Post-Processing

In this section, an iterative numerical solution method for the calculation of the inverse kinematics is illustrated in algorithm 3.1, the solution presented in eq. (2.47) and secondary task h from eq. (2.49) are implemented as optimal post-processing in this algorithm. The algorithm can be summarized as following:

Algorithm 1.1 optimal post-processor module

Read $\mathbf{P}_f$ and $\mathbf{O}_f$ from each line of APT − file

$\boldsymbol{\theta}$, $W$and W_0 $\longleftarrow$ initial guess

$\{\mathbf{P},\ \mathbf{O},\ J\} \longleftarrow DKM(\boldsymbol{\theta})$

For i = 1 to 500 step 1 do

$\boldsymbol{\Delta}\mathbf{P} \longleftarrow 0.5\ (\mathbf{P}_f - \mathbf{P})$

$h \longleftarrow -\nabla Z$

$\boldsymbol{\Delta\theta} \longleftarrow J^{+}\boldsymbol{\Delta}\mathbf{P} + (1 - J^{+}J)\ h$

if $\|\boldsymbol{\Delta}\mathbf{P}\| \leq \epsilon \cong 10^{-6}$m, then stop;

else

$\boldsymbol{\theta} \longleftarrow \boldsymbol{\theta} + 0.8\boldsymbol{\Delta\theta}$

$\{\mathbf{P},\ \mathbf{O},\ J\} \longleftarrow DKM(\boldsymbol{\theta})$

end if

End do

Vectors $\mathbf{P}_f$ and Of denote the desired final position and orientation of the tool frame relative to the part frame. The function DKM calculates the actual pose of the tool frame relative to the part frame. After each "Do" iteration, the next joint position $\boldsymbol{\theta}_{i+1}$ is computed as the actual joint position $\boldsymbol{\theta}_i$ plus a damping factor 0.8 times the joint displacement $\boldsymbol{\Delta\theta}$, in order to avoid overshooting of $\boldsymbol{\theta}_{i+1}$, for ill-condition Jacobian matrix.

Our algorithm allows a linear interpolation between the actual and final positions of the center of the ball-end mill, while leaving the control of the tool orientation to the secondary task. The latter tries to keep both the joint positions and the tool orientation at the closest deviation from the preferred tool orientation. A proper selection of the weight allows a good balancing between the two contradictory objectives.

3.4 Experimental Test of the Optimal Post-Processor

The forward and the inverse kinematics described in this study, along with the algorithm 3.1 and lineariztion algorithm, have been implemented in a post-processor module for the Milling machine center Huron KX8-Five. The post-processor was programmed using the Matlab language. The CL data was generated by CATIA V5.

Figure 3.2 shows a picture of Huron KX8-Five. This machine is a five-axis high speed

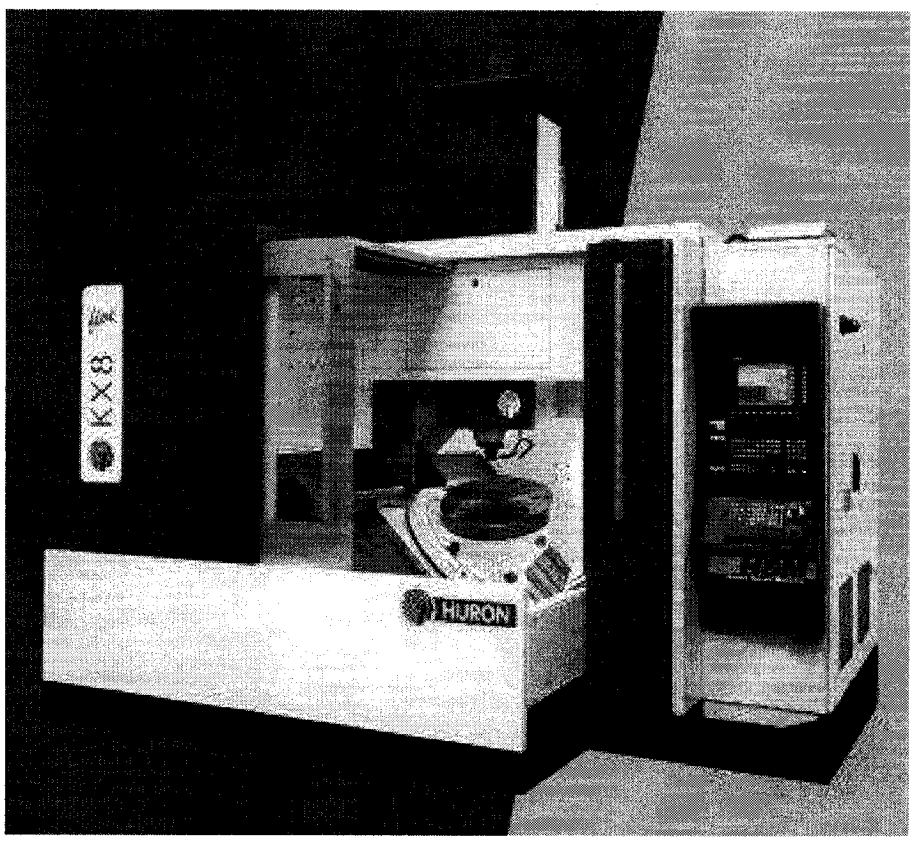


Figure 3.2: Milling machine center Huron KX8-Five

Machine (HSM) that use powerful motor and can machine a part with acceleration of 0.5 g (g=9.8 m/s^2). The positive direction of joints is shown on Fig. 3.3. The coordinate systems P and T are associated to top of rotary table C and center of sphere of ball-end

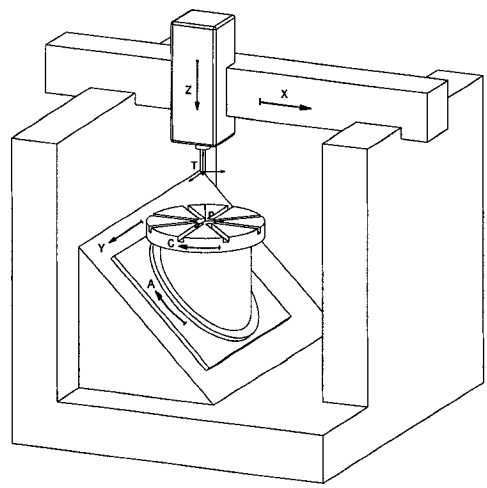


Figure 3.3: Schema of Huron KX8-Five

mill tool, respectively. The kinematic chain of the Huron can be represented as in Fig. 3.4. Associating the coordinate frames from the workpiece to the tool, according to our

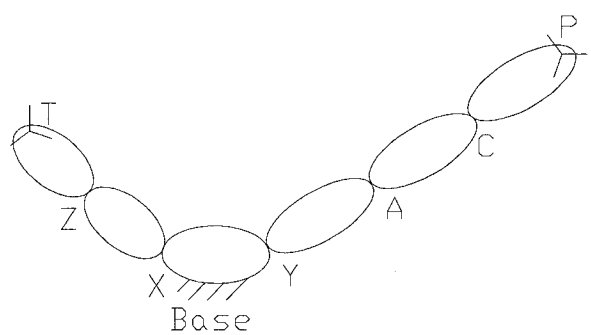


Figure 3.4: The kinematic chain of Huron KX8-Five

new method of D-H parameters introduced in chapter one, is shown in Fig. 3.5.Finally, the D-H parameters can be summarized as in table 3.1 From table 1.5, ${}^{P}A_{-3}, {}^{-3}A_{-2} \sim^{3} A_{T}$ can be found and therefore, the homogeneous transformation matrix of the Huron

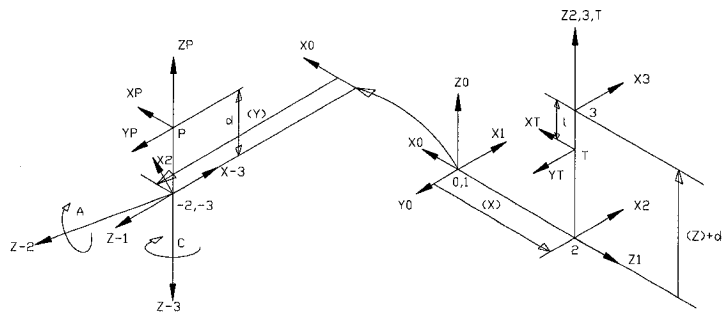


Figure 3.5: Coordinate frames of Huron KX8-Five from the workpiece to the tool

Table 3.1: D-H parameter notations from the workpiece to the tool

i	θ_i°	a_i	b_i	α_i°	$i+1$
P	(C)-90	0	-d	180	-3
-3	(A)+90	0	0	-45	-2
-2	-45	0	(Y)	-90	-1
-1	0	0	0	-90	0
0	-90	0	0	90	1
1	0	0	(X)	-90	2
2	0	0	(Z)+d	0	3
3	90	0	-l	0	T

KX8-Five is

$$ {}^{P}\mathbf{A}_T = \begin{bmatrix} R1 & R4 & i & x \\ R2 & R5 & j & y \\ R3 & R6 & k & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1} $$

where,

$R1 = 1/2\ \cos(C) - \sqrt{2}/2\ \sin(C)\sin(A) + 1/2\ \cos(C)\cos(A),$

$R2 = \sqrt{2}/2\ \cos(C)\sin(A) + 1/2\ \sin(C) + 1/2\ \sin(C)\cos(A),$

$R3 = 1/2\ \cos(A) - 1/2,$

$R4 = -\sqrt{2}/2\ \sqrt{2}\cos(C)\sin(A) - \sin(C)\cos(A),$

$R5 = \cos(C)\cos(A) - \sqrt{2}/2\ \sin(C)\sin(A),$

$R6 = -\sqrt{2}/2\ \sin(A),$

$i = -1/2\ \cos(C) - \sqrt{2}/2\ \sin(C)\sin(A) + 1/2\ \cos(C)\cos(A),$

$j = -1/2\ \sin(C) + \sqrt{2}/2\ \cos(C)\sin(A) + 1/2\ \sin(C)\cos(A),$

$k = 1/2\ \cos(A) + 1/2,$

$x = Y\sin(C)\cos(A) - 1/2\ \cos(C)\,(Z + d - l) - 1/2\,X\cos(C) + \sqrt{2}/2\,X\sin(C)\sin(A)$
$- \sqrt{2}/2\sin(C)\sin(A)\,(Z + d - l) - 1/2\,X\cos(C)\cos(A) + \sqrt{2}/2\,Y\cos(C)\sin(A),$

$y = \sqrt{2}/2\ \cos(C)\sin(A)\,(Z + d - l) - 1/2\,X\sin(C) + 1/2\ \sin(C)\cos(A)\,(Z + d - l)$
$-1/2\ \sin(C)\,(Z + d - l) + \sqrt{2}/2\,Y\sin(C)\sin(A) - \sqrt{2}/2\,X\cos(C)\sin(A) - Y\cos(C)$
$\cos(A)$, and

$z = -1/2\,X\cos(A) + 1/2\,X + \sqrt{2}/2\,Y\sin(A) + 1/2\ \cos(A)\,(Z + d - l) + 1/2\,(Z -$
$d - l)$. According to Fig. 3.6, the joints limits are as follow:

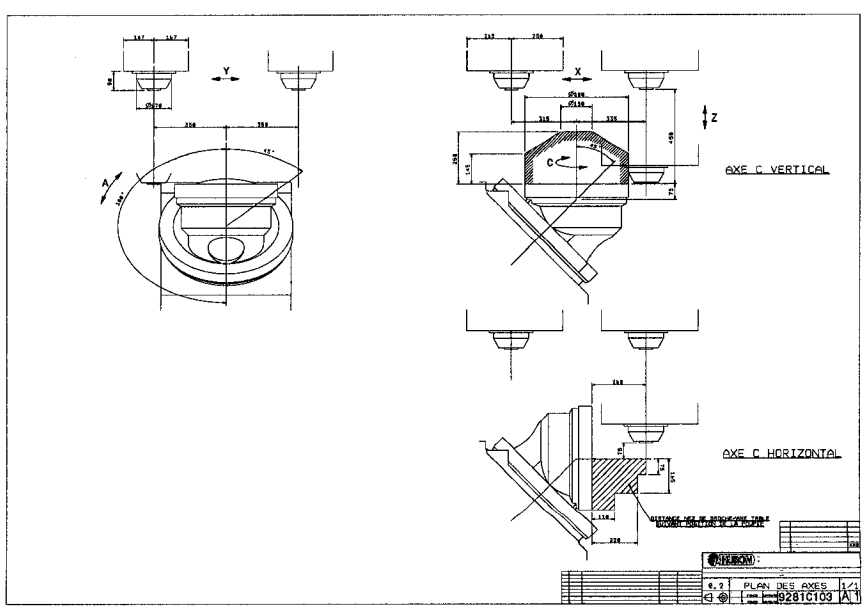


Figure 3.6: Joint limits of the Huron KX8-Five

$$\begin{aligned} -315\,mm \leq\ & X \leq 335\,mm \\ -350\,mm \leq\ & Y \leq 350\,mm \\ 0\,mm \leq\ & Z \leq 450\,mm \\ -180^\circ \leq\ & A \leq 45^\circ \\ -99,999.999^\circ \leq\ & C \leq 99,999.999^\circ \\ d\ =\ & 75\,mm \end{aligned} \tag{3.2}$$

Therefore, the forward kinematics of Huron KX8-Five is defined from eqs. (3.1) and joints limits (3.2). Using a Matlab programm base on the algorithm 1.1 presented in appendix II gives us the Jacobian matrix of the Huron KX8-Five automatically as following

$$J(\theta) = \begin{bmatrix} J_{1,1} & J_{2,1} & J_{3,1} & J_{4,1} & J_{5,1} \\ J_{1,2} & J_{2,2} & J_{3,2} & J_{4,2} & J_{5,2} \\ J_{1,3} & J_{2,3} & J_{3,3} & J_{4,3} & J_{5,3} \\ J_{1,4} & J_{2,4} & J_{3,4} & J_{4,4} & J_{5,4} \\ J_{1,5} & J_{2,5} & J_{3,5} & J_{4,5} & J_{5,5} \\ J_{1,6} & J_{2,6} & J_{3,6} & J_{4,6} & J_{5,6} \end{bmatrix} \tag{3.3}$$

where,

$J_{1,1} = 0,$

$J_{1,2} = 0,$

$J_{1,3} = 1,$

$J_{1,4} = -\sqrt{2}/2\cos(C)\sin(A)(Z+d-l)+1/2\sin(C)(Z+d-l)+\cos(C)\cos(A)Y-1/2\sin(C)\cos(A)(Z+d-l)+1/2\sin(C)X+\sqrt{2}/2\cos(C)\sin(A)X-\sqrt{2}/2\sin(C)\sin(A)Y+1/2\sin(C)\cos(A)X,$

$J_{1,5} = -\sqrt{2}/2\sin(C)\sin(A)(Z+d-l)-1/2\cos(C)(Z+d-l)+\sqrt{2}/2\sin(C)\sin(A)X+\sqrt{2}/2\cos(C)\sin(A)Y-1/2\cos(C)\cos(A)X+1/2\cos(C)\cos(A)(Z+d-l)+\sin(C)\cos(A)Y-1/2\cos(C)X,$

$J_{1,5} = 0,$

$J_{2,1} = -\sqrt{2}/2 \cos(C),$

$J_{2,2} = -\sqrt{2}/2 \sin(C),$

$J_{2,3} = \sqrt{2}/2,$

$J_{2,4} = -1/2 \cos(C) \sin(A)(Z+d-l) + \sqrt{2}/2 \cos(C) \cos(A) Y - \sqrt{2}/2 \sin(C) \cos(A)$
$(Z+d-l) - \sin(C) \sin(A) Y + 1/2 \cos(C) \sin(A) X + \sqrt{2}/2 \sin(C) \cos(A) X,$

$J_{2,5} = -\sqrt{2}/2 \cos(C) \cos(A) X + \sqrt{2}/2 \sin(C) \cos(A) Y + 1/2 \sin(C) \sin(A) X +$
$\cos(C) \sin(A) Y + \sqrt{2}/2 \cos(C) \cos(A)(Z+d-l) - 1/2 \sin(C) \sin(A)(Z+d-l),$

$J_{2,6} = 1/2 \sin(A) X - 1/2 \sin(A)(Z+d-l) + \sqrt{2}/2 \cos(A) Y,$

$J_{3,1} = 0,$

$J_{3,2} = 0,$

$J_{3,3} = 0,$

$J_{3,4} = \sqrt{2}/2 \sin(C) \sin(A) - 1/2 \cos(C) \cos(A) - 1/2 \cos(C),$

$J_{3,5} = -\cos(C) \cos(A) + \sqrt{2}/2 \sin(C) \sin(A),$

$J_{3,6} = \sqrt{2}/2 \sin(A),$

$J_{4,1} = 0,$

$J_{4,2} = 0,$

$J_{4,3} = 0,$

$J_{4,4} = \sqrt{2}/2 \sin(C) \sin(A) - 1/2 \cos(C) \cos(A) - 1/2 \cos(C),$

$J_{4,5} = -\sqrt{2}/2 \cos(C) \sin(A) - 1/2 \sin(C) \cos(A) - 1/2\, sin\,(C),$

$J_{4,6} = -1/2 \cos(A) + 1/2,$

$J_{5,1} = 0,$

$J_{5,2} = 0,$

$J_{5,3} = 0,$

$J_{5,4} = 1/2 \cos(C) \cos(A) - \sqrt{2}/2 \sin(C) \sin(A) - 1/2 \cos(C),$

$J_{5,5} = \sqrt{2}/2 \cos(C) \sin(A) + 1/2 \sin(C) \cos(A) - 1/2 \sin(C),$ and

$J_{5,6} = 1/2 \cos(A) + 1/2.$

3.5 Example of Operation

The example of machining operation is a free form surface as shown in Fig. 3.7. The

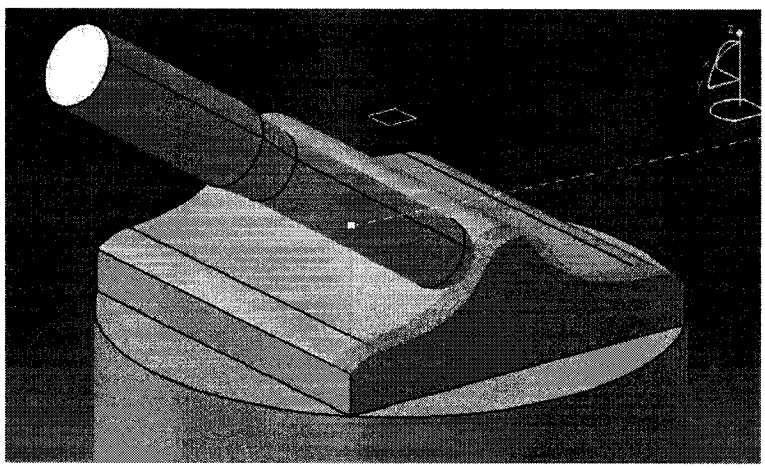


Figure 3.7: Tool path generated by CATIA

surface is inscribed in a 4 inch dia. cylindric block. A multi-axis zigzag strategy with a 0.5 inch ball-end mill tool is chosen as a finish milling operation. The scallop height of the tool path is 0.005 mm and the machining tolerance is 0.01 mm. The orientation of tool has been chosen normal to surface. An APT-file is generated for this operation from center of sphere of ball-end mill tool by CATIA V5.

3.6 Post-Processor Output

The five-axis APT-file from CATIA after linearization has been post-processed for the milling machine center Huron KX8-Five in four way in order to demonstrate the feasibility of operation and also to compare the effect of tool orientation on surface quality. 1) CL data processed always parallel to vertical direction; 2) tool always at a constant orientation different than the vertical direction; 3) tool always normal to the surface as

it is desirable, and finally, 4) tool almost normal to the surface using the optimal post-processor module in order to avoid exceeding joint limits.

The processed joint positions value from the third test are shown in Fig. 3.8. According

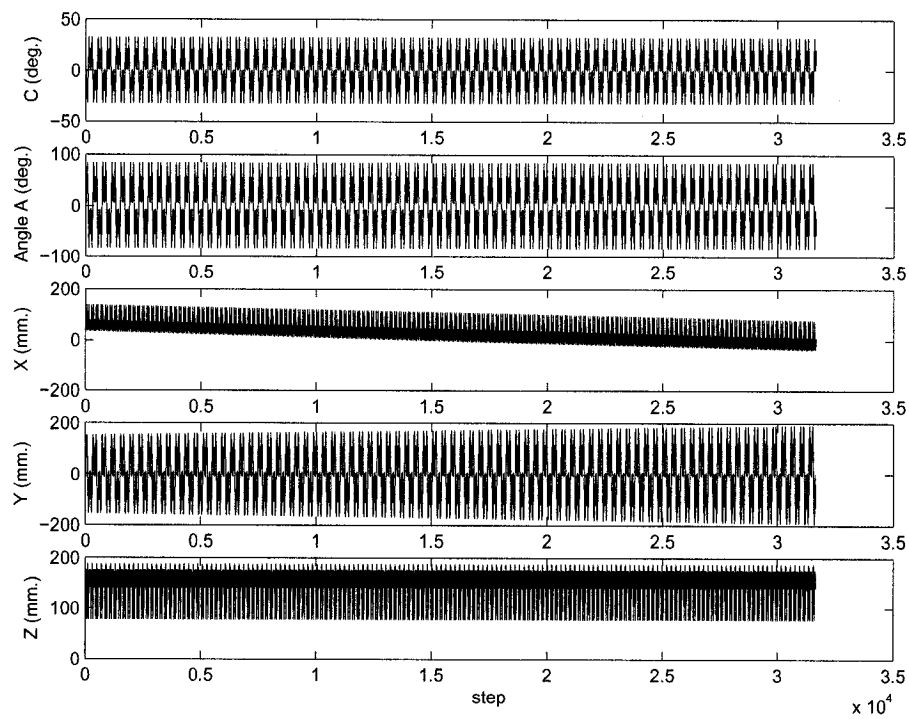


Figure 3.8: The processed joint positions value when the tool is normal of the surface

the eq. (3.2) limit of joint limit A is; $-180° \leq A \leq 45°$ but from Fig. 3.8, the processed joint A is between $-82.8°$ and $82.8°$, Therefore, The joint A exceed its limit and the NC-file generated from this process can not be accomplished . In order to accomplish this operation, the CL-data must be processed by an optimal post-processor module which can identify the joint limits.

In order to have a optimal post-processing the secondary task h from eq. (2.49) for the Huron KX8-Five must be defined. The calculation of this secondary task can be shown as follow

From the DKM of the Huron KX8-Five, eq. (3.1), we have:

$$O = \begin{bmatrix} i = -1/2\cos(C) - \sqrt{2}/2\sin(C)\sin(A) + 1/2\cos(C)\cos(A) \\ j = -1/2\sin(C) + \sqrt{2}/2\cos(C)\sin(A) + 1/2\sin(C)\cos(A) \\ k = 1/2\cos(A) + 1/2 \end{bmatrix},$$

If $O_f = \begin{bmatrix} O_{fx} \\ O_{fy} \\ O_{fz} \end{bmatrix}$;

$$\begin{aligned} O^T \cdot O_f &= -1/2\cos(C)\,O_{fx} - \sqrt{2}/2\sin(C)\sin(A)\,O_{fx} + 1/2\cos(C)\cos(A)\,O_{fx} \\ &\quad -1/2\sin(C)\,O_{fy} + \sqrt{2}/2\cos(C)\sin(A)\,O_{fy} + 1/2\sin(C)\cos(A)\,O_{fy} \\ &\quad +1/2\cos(A)\,O_{fz} + 1/2O_{fz} \end{aligned} \tag{3.4}$$

$$\text{and} \quad \frac{d}{d\theta}(O^T \cdot O_f) = \begin{bmatrix} \frac{d}{dC}(O^T \cdot O_f) \\ \frac{d}{dA}(O^T \cdot O_f) \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.5}$$

Finally,

$$h = -W \begin{bmatrix} C - \bar{C} \\ A - \bar{A} \\ Y - \bar{Y} \\ X - \bar{X} \\ Z - \bar{Z} \end{bmatrix} + W_0 \begin{bmatrix} \frac{d}{dC}(O^T \cdot O_f) \\ \frac{d}{dA}(O^T \cdot O_f) \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where,

$\frac{d}{dC}(O^T \cdot O_f) = -\frac{1}{2}S_C C_A O_{fx} + \frac{1}{2}S_C O_{fx} - \frac{\sqrt{2}}{2}C_C S_A O_{fx} - \frac{1}{2}S_C C_A O_{fy} + \frac{1}{2}S_C O_{fy} - \frac{\sqrt{2}}{2}C_C S_A O_{fy}$,

$\frac{d}{dA}(O^T \cdot O_f) = -\frac{1}{2}C_C S_A O_{fx} - \frac{\sqrt{2}}{2}S_C C_A O_{fx} - \frac{1}{2}S_C S_A O_{fy} + \frac{\sqrt{2}}{2}C_C C_A O_{fy} - \frac{1}{2}S_A O_{fz}$,

$W = diag(W_C, W_A, W_Y, W_X, W_Z)$ and $\bar{\theta} = \frac{1}{2}(\theta_{max} + \theta_{min})$

The mid-joint of $\bar{\theta}$ can be calculated from eq. (3.2), i.e.,

$$\bar{\theta} = \begin{bmatrix} \bar{C} = 0^\circ \\ \bar{A} = (45 - 180)/2 = -67.5^\circ \\ \bar{Y} = 0mm \\ \bar{X} = 10mm \\ \bar{Z} = 225mm \end{bmatrix} \tag{3.6}$$

The weighting values W and W_0 can be determined after some tried. The objective of this optimal post-processing is to try to approach to the mid-joint of A in the first part of h and also to try to approach the normal to the surface O_f in the second part of h. The results of successful optimal post-processing are shown on the following figures by considering the weighting value $W = diag(0, 0.135, 0, 0, 0)$ in the first part and $W_0 = 0.85$ in second part. The joint positions from the optimal post-processing module are illustrated in Fig. 3.9. This figure shows that the joint A values are much

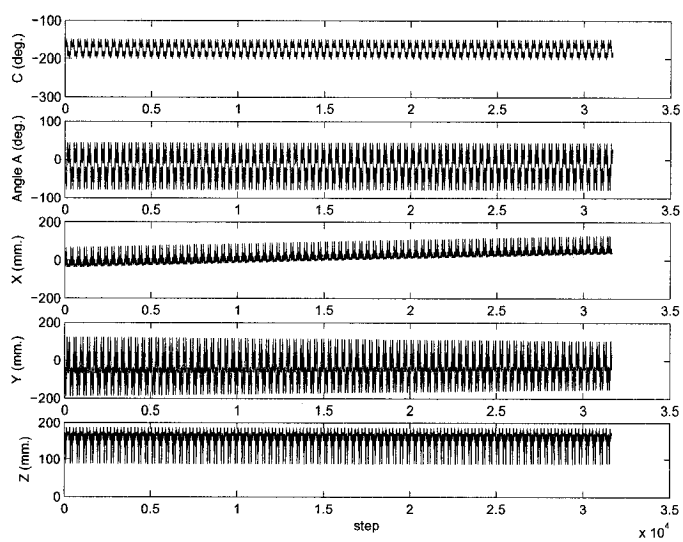


Figure 3.9: The processed value of joints form optimal post-processing (fourth test)

lower, i.e., between $-78.5°$ and $43.5°$ that is within the limit of this joint. Therefore, the NC-file generated from this optimal process can be accomplished by the machine-tool without exceeding the joint limits. In order to verify the numerical stability of our solution procedure the condition number of the Jacobian matrix is shown in Fig. 3.10. According to Fig. 3.10 the Jacobian matrix has acceptable condition number, while

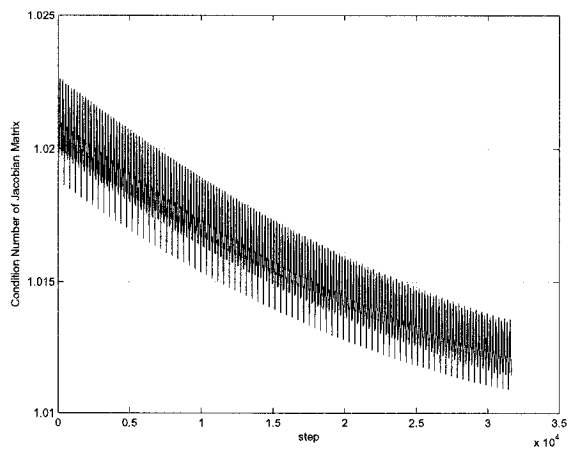


Figure 3.10: Condition number of the Jacobian matrix from optimal post-processing

processing. Also, the positioning error of this process is shown in Fig. 3.11, according

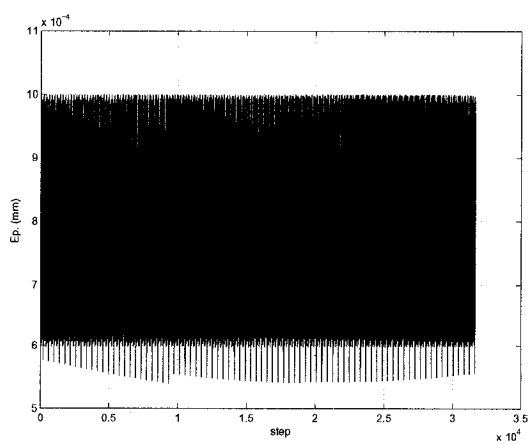


Figure 3.11: Positioning error from optimal post-processing

this figure, the positioning error is always less than $0.0001mm$, that is negligible with comparing to machining tolerance. For simplicity of observation, the processed result of one cycle of CL-data is illustrated in the following figures. Figure 3.12 shows, the joint positions A and C from two different post-processes, normal and optimal post-processing (third and fourth tests). The joint position A in optimal post-processing does

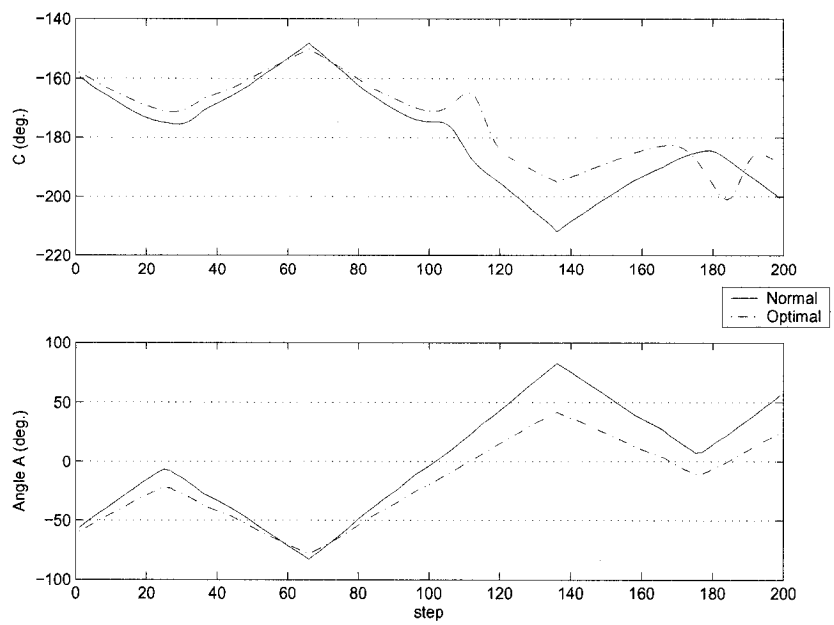


Figure 3.12: Comparing the joint positions A and C from the third and fourth tests. (normal and optimal post-processing)

not exceed its joint limit. Whereas, the tool orientation from the two processes has three dimension geometry and difficult to illustrate, hence, in order to illustrate that, the CL-data file from two process is created by using forward kinematics and shown in the following figures. Figure 3.13 illustrates the tool path from normal post-processing, when the tool is always normal to the surface. Figure 3.14 illustrates the optimal tool path from optimal post-processing. The relative tool orientation between these two processes is shown in Fig. 3.15. According to the figures the joint position A is optimized in order to stay in its joint limits.

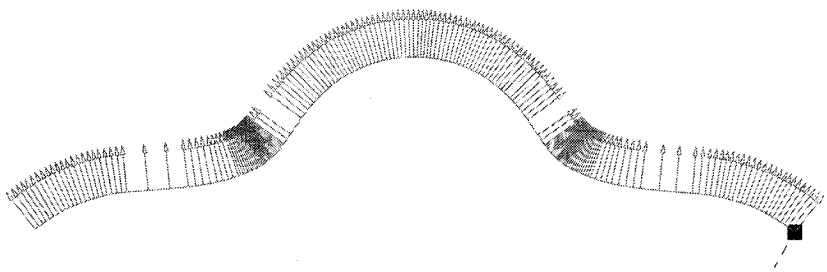


Figure 3.13: Normal post-processing

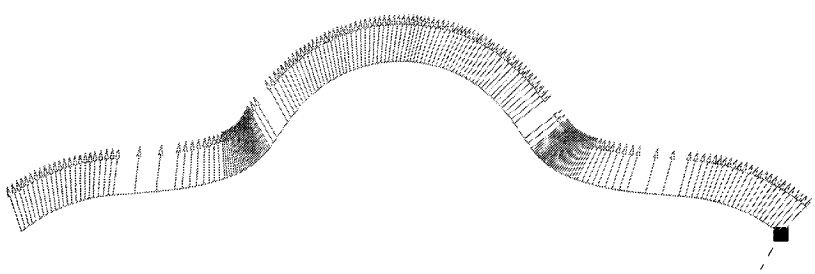


Figure 3.14: Optimal post-processing

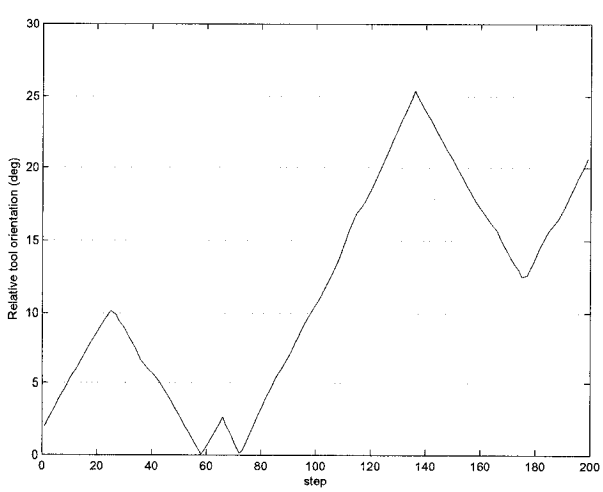


Figure 3.15: Relative tool orientation between processes 3 and 4

3.7 Machining Tests

The machining tests was conducted to demonstrate the feasibility of the optimal tool path and to compare machining quality with our four ways of machining. In order to avoid the risk of human or system error, the generated tool path should be verified before actual machining. Figure 3.16 shows the process of multi-axis machining of a part simulated by VERICUT software. The Huron KX8-Five use a SIEMENS 840D controller. Therefore,

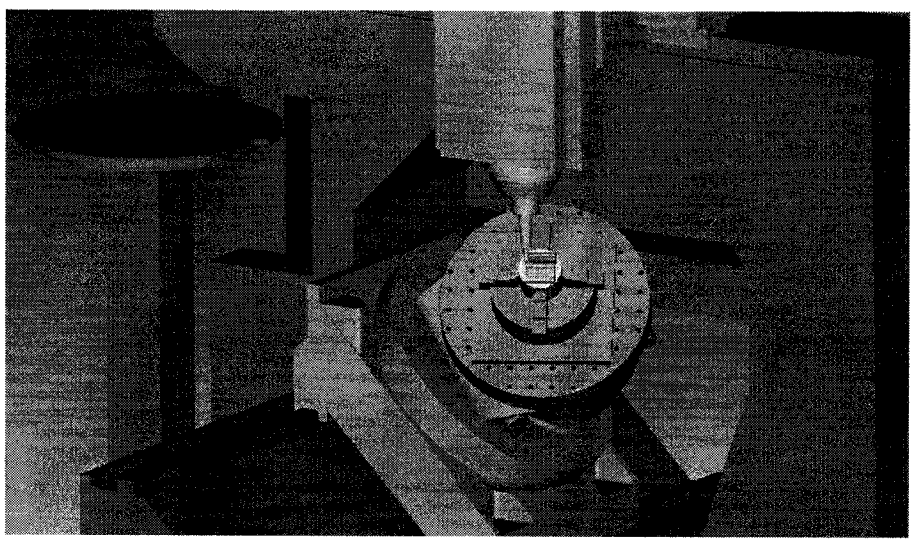


Figure 3.16: Simulation of a part

some operation commands must be added to our processed Gcode-file in order to adjust the feed rate and other parameters to have a suitable an accurate machining procedure. Figure 3.17 shows the Huron KX8-Five in multi-axis machining process of one of our test parts. As mentioned before, the task with tool orientation normal to the free form surface, cannot be accomplished with this machine, Hence, the CL-data of this file is cut and the part was machined until $A = 43°$, before it reaches the limit. The result after machining is shown in Fig. 3.18. The part number 1 is machined by 0.5" ball-end mill tool always parallel to Z axe of machine tool, i.e., $A = 0$ and $C = 0$. The part

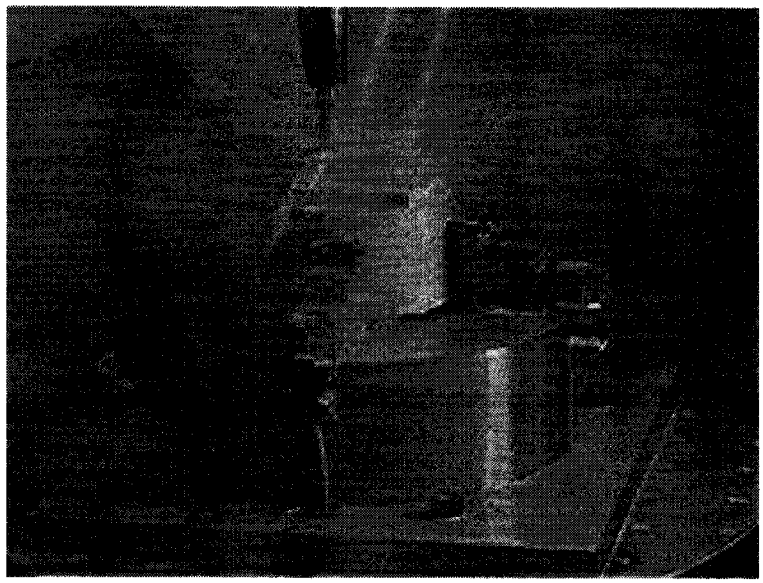


Figure 3.17: The Huron KX8-Five in a multi-axis machining process

number 2 is machined by a 0.5" ball-end mill tool with constant angle for joint A, i.e., $A = 14°$ for one side and $A = -14°$ for the other side. The part number 3 is machined by 0.5" ball-end mill tool always normal to the free form surface before joint limit A is exceeded. Finally, the part number 4 is machined successfully by the same tool with consideration of joint limit A, it means the optimal orientation of tool, using the optimal post-processor module.

Roughness of the final surface of the four machined part is measured by the Mitutoyo Surftest surface roughness measuring machine. The way of measurement is illustrated in Fig. 3.19, the probe of measuring machine has measured three line, length 12.5 mm, in one direction on two side of each part. The results and figures of measurement are shown in appendix I and the values of Ra for each measurement summarized in table 3.2. The values of Ra the third part is more than the other parts because of reaching to joint limit A-axis, only front face of this part is machined and tool path for machining is shorter than the other part. Hence, the machining can not be accomplished with a suitable feed rate as be accomplished on the other parts. The results demonstrate that the

Figure 3.18: The four machining tests

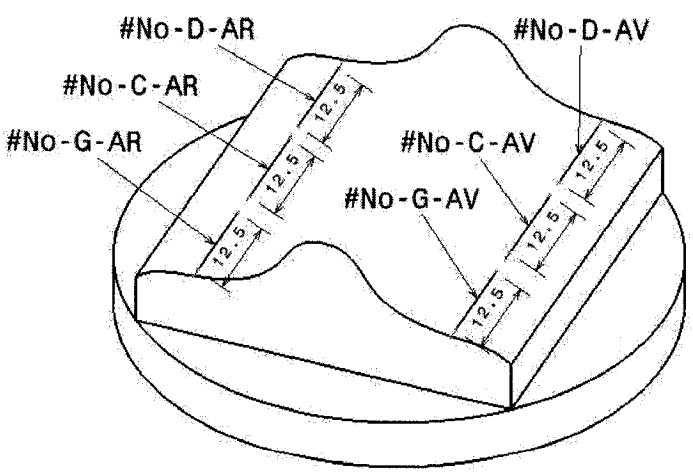


Figure 3.19: Location of measurement on the parts

Table 3.2: Values of Roughness of four parts

Number of part	Rear	Average of $R_a(\mu m)$	Front	Average of $R_a(\mu m)$
1	1-D-AR	1.4	1-D-AV	1.3
1	1-C-AR	1.394	1-C-AV	1.444
1	1-G-AR	1.826	1-G-AV	1.802
2	2-D-AR	1.211	2-D-AV	1.468
2	2-C-AR	1.589	2-C-AV	1.254
2	2-G-AR	1.79	2-G-AV	1.423
3	3-D-AR	-	3-D-AV	1.557
3	3-C-AR	-	3-C-AV	1.572
3	3-G-AR	-	3-G-AV	1.404
4	4-D-AR	1.396	4-D-AV	1.2
4	4-C-AR	1.408	4-D-AV	1.18
4	4-G-AR	1.642	4-G-AV	1.223

part 4, which is machined by our optimal tool orientation has the less Ra.

CHAPTER 4

CONCLUSION

In this thesis, an orthogonalization procedure was used to compute the inverse kinematic of five-axis CNC milling machines performing a surfacing task. The solution procedure has been implemented within a post-processor module and tested with the architecture of the Huron KX8-Five milling center. Different shapes of surfaces have been successfully machined with the module without exceeding joint limits. It was shown that the procedure allows the addition of a secondary task to the primary one, which improves the quality of the task performed. The experimental tests was conducted to demonstrate the feasibility of the optimal post-processing tool path and to compare machining quality with different way of machining. The part 4, which is machined without exceeding the joint limits by our optimal tool orientation has acceptable surface quality.

4.1 Future Works

The weighting values W and W_0 which is used in our optimal post-processor depend on the shape of free form surface and strategy of machining. Different way of machining the surface need different weighting values. These values find after some try and error in process. Hence, our optimal post processor need an automatical method to find the weighting values.

Our optimal post-processor can not supply the new codes which are used for high speed machining, these codes work base on spline or nurbs curve instead of small line. The movement of machine-tool with these codes is faster than G01. Therefore, study in two cases is recommended as future works.

REFERENCES

[1] Angeles, J., *Fundamentals of robotic mechanical systems: theory, methods and algorithms*, Springer, New York, 107, 390 pages, 2003.

[2] Angeles, J., Anderson, K. and Gosselin, C., *An Orthogonal-Decomposition Algorithm for Constrained Least-Square Optimization,* ASME Robotics, Mechanisms and Machine Systems, Design Eng. Division, Vol. 2, pp. 215–220, 1987.

[3] Arenson, N., Angeles, J. and Slutski, L., *Redundancy-resolution algorithms for isotropic robots,* Advances in Robot Kinematics: Analysis and Control, pp. 425–434, 1998.

[4] Angeles, J. and Mathur, S., *Resolved-rate control of redundant manipulators with elimination of nonconservative effects* Robotics Research. Fifth International Symposium, pp. 209–216, 1990.

[5] Baillieul, J., *Kinematic programming alternative for redundant manipulator,* Proc. International Conference on Robotics and Automation, St-Louis, pp. 722–728, 1985.

[6] Baron, L., *An Optimal Surfacing Post-Processor Module for 5-Axes CNC Milling Machines,* Third International Conf. on Industrial Automation, Montréal, Canada, June 1999.

[7] Baron, L., *A joint-limits avoidance strategy for arc-welding robots,* International Conference on Integrated Design and Manufacturing in Mechanical Engineering, Montreal, Canada, May 2000.

[8] Baron, L., Mayer, R., Arghavani, J. and Fortin C., *Literature review of the kinematic theories Pertaining to NC Post-Processor,* CDT Research contract No P1981, Ecole Polytechnique de Montreal,75 pages, 1995.

[9] Baugman, J.A., *Multi-axis machining with APT,* in: W.H.P. Leslie (ED), Numerical Control User's Handbook, McGraw-Hill, New York, pp. 271-298, 1970.

[10] Bézier, P., *Numerical control mathematics and applications,* Wiley, New York, 1972.

[11] Boyd, B.K., *Five-axis machining'* in: Mechanical Design, pp. 134-138, May 16 1974.

[12] Chang, T.-F. and Dubey, R.-V., *A weighted least-norm solution based scheme for avoiding joints limits for redundant manipulators,* IEEE Trans. Robot. Automat.,vol. 11, pp. 286–292, Apr. 1995.

[13] Denavit, J. and Hartenberg, R.S., *A kinematics notation for lower-pair mechanism based on matrices,* ASME J. Appl. Mech. 22 (2) pp. 215–221, 1955.

[14] Shareghi, F. and Baron, L., *An optimal redundancy-resolution scheme for post-processing of 5-axes milling machine,* International Conference on Integrated Design and Manufacturing in Mechanical Engineering, Montréal, Canada, 16-19 may 2000.

[15] Golub, G.H. and Van Loan, C.F., *Matrix computations,* The Johns Hopkins University Press, Baltimore and London, 1989.

[16] Huo, L., and Baron, L., Kinematic inversion of functionally-redundant serial manipulators: application to arc-welding, *Transactions of the Canadian Society for Mechanical Engineering, 2005*, Canada.

[17] Jung Y.H., et al., *Nc post-processor for 5-axis milling machine of table-rotating/tilting type,* Journal of Materials Processing Technology, vol. 130-131, pp. 641-646, Dec 20, 2002.

[18] Liégeois, A., *Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms,* IEEE Trans. Syst., Man, Cybern., vol. SMC-7, pp. 245–250, Mar. 1977.

[19] Lee, R.S. She, C.H., *Developing a post-processor for three types of five-axis machine tools,* International Journal of Advanced manufacture 13 pp. 658-665, 1997.

[20] Kelmar, L. and Khosla, P.K., *Automatic generation of forward and inverse kinematics for a reconfigurable modular manipulator system,* Journal of Robotic Systems, Vol. 7, No. 4, pp. 599-619, 1990.

[21] Klein, C. A., *Use of redundancy in the design of robotic systems,* 2nd, Int. Symp. of Robotics Research, MIT Press, pp. 207–214, August 1984.

[22] Klein, C. and Huang, C.H., *Review of pseudoinverse control for use with kinematically redundant manipulators,* IEEE Trans. on Systems, Man, and Cybernetics, Vol. SMC-13, No. 3, pp. 245–250, 1983.

[23] Klein, C. A. and Maciejewski, A. A., em Obstacle avoidance for kinemaically redundant manipulators in dynamically varying enviroments, International Journal of Robotics Research, Vol. 4, No. 3, pp. 109–117, 1985.

[24] Kosuge, K. and Furuta, K., *Kinematic and dynamic analysis of robot arm,* IEEE Int. Conf. on Robotics and Automation, pp. 1039–1044, 1985.

[25] McCarthy, J.M., *An introduction to theoretical kinematics,* the MIT Press, Cambridge, Mass. London, England, 1990.

[26] Mooring, W. B., Roth, Z. S. and Driels, M. R., *Fundamentals of manipulator calibration,* A Wiley-interscience Publication, pp. 329, 1991.

[27] Nakamura, Y., *Advanced robotics: redundancy and optimization*, Addison-Wesley Pub. Co., Massachusetts, 337 pages, 1991.

[28] Nakamura, Y. and Hanafusa, H., *Optimal redundancy control of robot manipulators*, International Journal of Robotics Research, Vol. 6, No. 1, Spring, pp. 32–42, 1987.

[29] Nakamura, Y., Hanafusa, H. and Yoshikawa, T., *Task-priority based redundancy control of robot manipulators*, Int. J. Robotics Research, Vol. 6, No. 2, 1987.

[30] Park, J., Chung, W. and Youm, Y., *Weighted decomposition of kinematics and dynamaics of kinematically redundant manipulators*, IEEE International Conference on Robotics and Automation, Vol. 1, pp. 480–486, 1996.

[31] Pieper, D.L., *The Kinematics of Manipulators under Computer Control,* Ph.D. thesis, Stanford University, 1968.

[32] Salisbury, J.K. and Craig, J.J., *Articulated hands: force control and kinematic issus,* The Int. J. of Robatics Research, Vol. 1, No. 1, pp. 4-17, 1982.

[33] Siciliano, B., *Solving manipulator redundancy with the augmented task space methode using the constraint Jacobian transpose,* IEEE Intern. Conf. on Robotics and Automation, Tutorial M1, pp. 5.1–5.8, 1992.

[34] Sorby K., *Inverse kinematics of five-axis machines near singular configurations,* International Journal of Machine Tools and Manufacture may 2006.

[35] Sugimoto, K. and Duffy, J., *Application of linear algebra to screw system,* Mechanism and machine theory, vol. 17, No. 1, pp. 73-83, 1982.

[36] Whitney, D.E., *Resolved motion rate control of manipulators and human prostheses.* IEEE Trans. Man-Machine Syst., vol. 10, no. 2, pp. 47–53. 1969.

[37] Whitney, D.E., *The mathematics of coordinated control of prosthetic arms and manipulator,* ASME J. Dynamics Systems, Measurement and Control, Vol. 94, No. 4, pp. 303–309, 1972.

[38] Yashi, O.S., and Ozgoren, K., *Minimal joint motion optimization of manipulators with extra degrees of freedom,* Mechanism and Machine Theory, Vol. 19, No. 3, pp. 325–330, 1984.

[39] Yoshikawa, T., *Analyais and control of robot manipulators with redundancy,* Robotics Research: The First International Symposium, pp. 735–747, 1984.

APPENDIX I

THE RESULTS AND FIGURES OF ROUGHNESS MEASUREMENT

Rugosite Piece: #1-C-AR 06/12/13

Parameter Sum Table

	Profile=R_ISO - Sections=[1]	Average Value
Ra (um)	1.393	1.393
Rq (um)	1.655	1.655
Rsk	0.130	0.130
Rku	2.179	2.179
Rp (um)	3.669	3.669
Rv (um)	3.097	3.097
Rz (um)	6.766	6.766
Rt (um)	7.920	7.920
Rc (um)	4.916	4.916
RSm (mm)	0.677	0.677
Rdq	0.022	0.022
Rk (um)	4.948	4.948
Rpk (um)	0.869	0.869
Rvk (um)	1.256	1.256
Mr1 (%)	7.789	7.789
Mr2 (%)	94.282	94.282

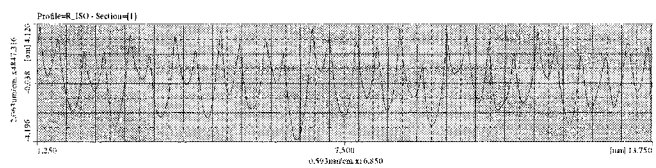
Evaluate Condition List<<Profile=R_ISO - Sections=[1]>>

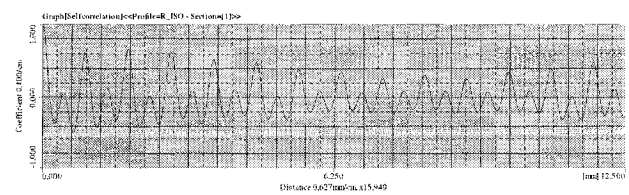
Kind of Profile:	R_ISO
Sampling Length(lc):	2.5 mm
No of Sampling(n):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evlun Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 mm
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Pitch(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AA(731-12AA8355)
Point Reversal	Off	Straightness Compensation	Off
Autocompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #1-C-AR 06/12/13





Rugosite Piece: #1-C-AV

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.444	1.444
Rq (um)	1.741	1.741
Rsk	-0.023	-0.023
Rku	2.332	2.332
Rp (um)	3.804	3.804
Rv (um)	3.453	3.453
Rz (um)	7.257	7.257
Rt (um)	8.467	8.467
Rc (um)	5.402	5.402
RSm (mm)	0.684	0.684
Rdq	0.023	0.023
Rk (um)	4.937	4.937
Rpk (um)	1.095	1.095
Rvk (um)	1.485	1.485
Mr1 (%)	7.862	7.862
Mr2 (%)	91.578	91.578

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

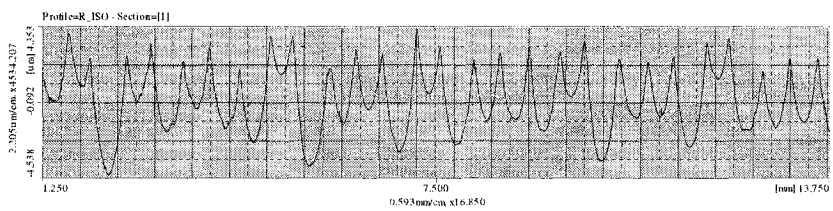
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

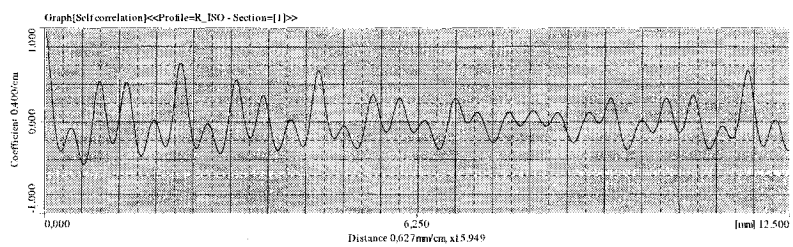
Rugosite Piece: #1-C-AV

06/12/13

Profile=R_ISO - Section=[1]



Graph[Self correlation]<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #1-D-AR

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.400	1.400
Rq (um)	1.667	1.667
Rsk	0.113	0.113
Rku	2.319	2.319
Rp (um)	3.980	3.980
Rv (um)	3.055	3.055
Rz (um)	7.035	7.035
Rt (um)	8.651	8.651
Rc (um)	5.193	5.193
RSm(mm)	0.702	0.702
Rdq	0.024	0.024
Rk (um)	4.961	4.961
Rpk (um)	1.175	1.175
Rvk (um)	1.006	1.006
Mr1 (%)	7.218	7.218
Mr2 (%)	93.692	93.692

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

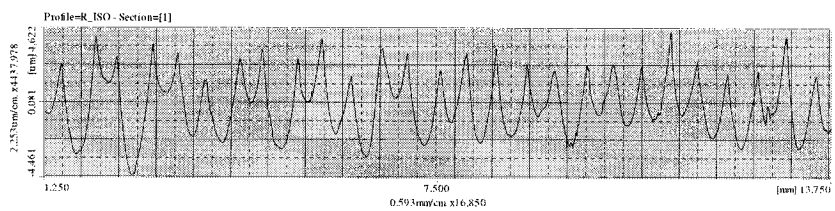
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

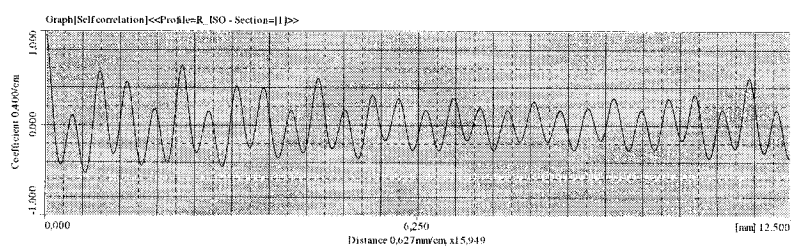
Rugosite Piece: #1-D-AR

06/12/13

Profile=R_ISO - Section=[1]



Graph[Selfcorrelation]<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #1-D-AV

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.300	1.300
Rq (um)	1.557	1.557
Rsk	0.164	0.164
Rku	2.303	2.303
Rp (um)	3.716	3.716
Rv (um)	2.834	2.834
Rz (um)	6.550	6.550
Rt (um)	7.429	7.429
Rc (um)	4.453	4.453
RSm (mm)	0.551	0.551
Rdq	0.023	0.023
Rk (um)	4.399	4.399
Rpk (um)	1.126	1.126
Rvk (um)	0.887	0.887
Mr1 (%)	9.140	9.140
Mr2 (%)	91.606	91.606

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

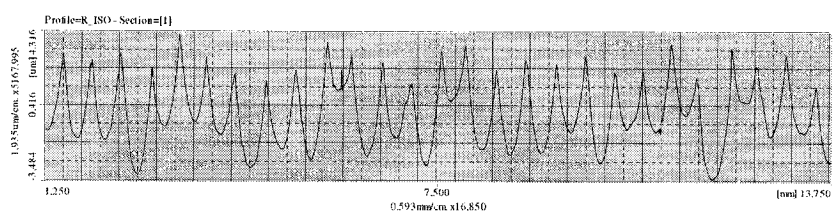
Kind of Profile:	R_ISO
Smplg Length(lc):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

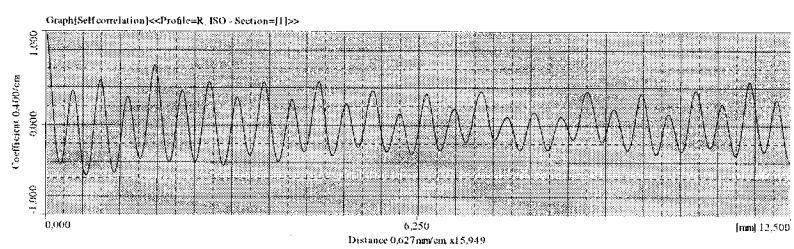
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AA(731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #1-D-AV

06/12/13





Rugosite Piece: #1-G-AR

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.826	1.826
Rq (um)	2.148	2.148
Rsk	-0.211	-0.211
Rku	1.976	1.976
Rp (um)	4.038	4.038
Rv (um)	3.962	3.962
Rz (um)	8.000	8.000
Rt (um)	9.643	9.643
Rc (um)	6.966	6.966
RSm (mm)	0.875	0.875
Rdq	0.023	0.023
Rk (um)	5.827	5.827
Rpk (um)	0.857	0.857
Rvk (um)	1.562	1.562
Mr1 (%)	6.129	6.129
Mr2 (%)	85.224	85.224

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

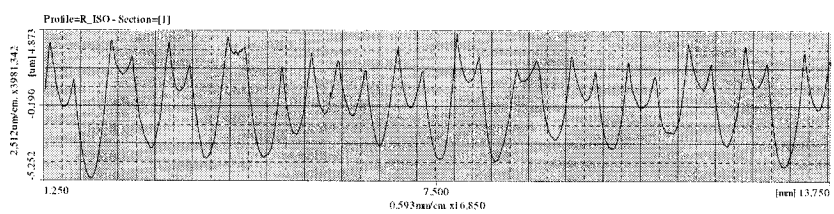
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

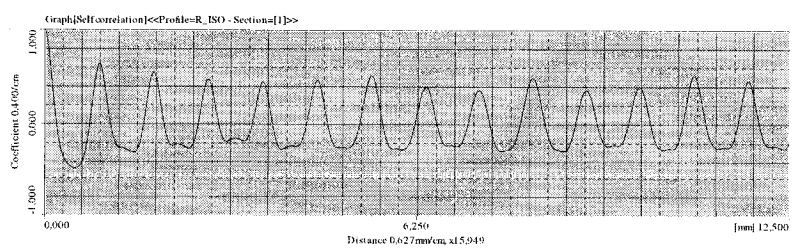
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #1-G-AR

06/12/13





Rugosite Piece: #1-G-AV

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.802	1.802
Rq (um)	2.150	2.150
Rsk	-0.257	-0.257
Rku	2.181	2.181
Rp (um)	3.959	3.959
Rv (um)	3.983	3.983
Rz (um)	7.942	7.942
Rt (um)	9.949	9.949
Rc (um)	6.093	6.093
RSm (mm)	0.852	0.852
Rdq	0.024	0.024
Rk (um)	6.025	6.025
Rpk (um)	1.012	1.012
Rvk (um)	2.379	2.379
Mr1 (%)	6.475	6.475
Mr2 (%)	90.412	90.412

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

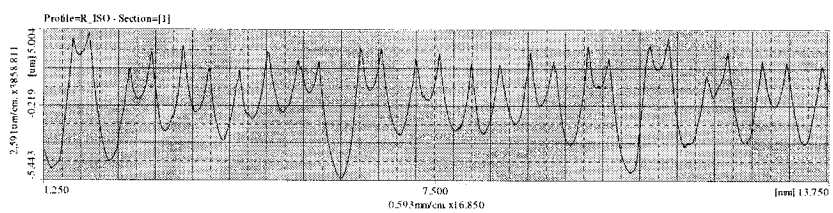
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

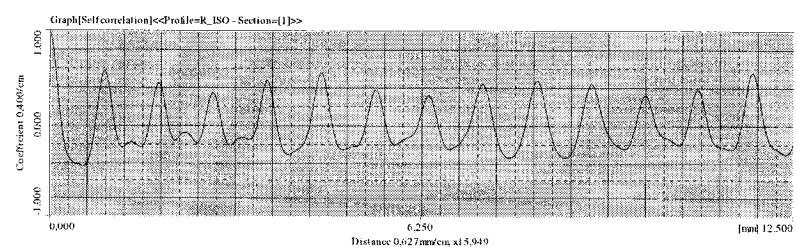
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #1-G-AV

06/12/13





Rugosite Piece: #2-G-AV

06/12/13

Parameter Sum/Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.421	1.421
Rq (um)	1.682	1.682
Rsk	-0.059	-0.059
Rku	2.006	2.006
Rp (um)	3.311	3.311
Rv (um)	3.067	3.067
Rz (um)	6.378	6.378
Rt (um)	7.321	7.321
Rc (um)	5.461	5.461
RSm (mm)	0.816	0.816
Rdq	0.026	0.026
Rk (um)	4.994	4.994
Rpk (um)	0.625	0.625
Rvk (um)	0.919	0.919
Mr1 (%)	5.570	5.570
Mr2 (%)	90.143	90.143

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

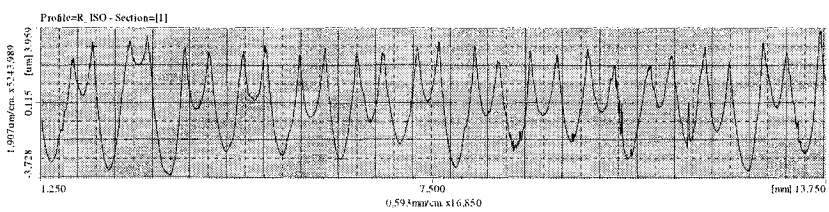
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

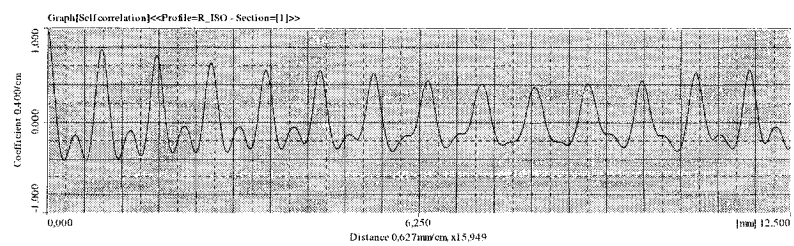
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.00005 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #2-G-AV

06/12/13





Rugosite Piece: #2-G-AR

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.790	1.790
Rq (um)	2.058	2.058
Rsk	-0.230	-0.230
Rku	1.788	1.788
Rp (um)	3.700	3.700
Rv (um)	3.418	3.418
Rz (um)	7.118	7.118
Rt (um)	8.181	8.181
Rc (um)	6.724	6.724
RSm (mm)	0.923	0.923
Rdq	0.025	0.025
Rk (um)	5.445	5.445
Rpk (um)	0.650	0.650
Rvk (um)	1.215	1.215
Mr1 (%)	4.328	4.328
Mr2 (%)	79.794	79.794

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

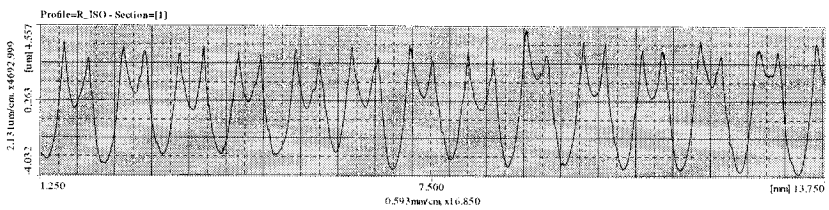
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100nm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

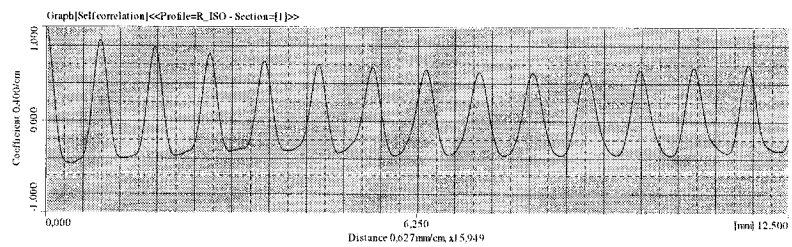
Rugosite Piece: #2-G-AR

06/12/13

Profile=R_ISO - Section=[1]



Graph|SelfCorrelation|<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #2-D-AV

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.468	1.468
Rq (um)	1.742	1.742
Rsk	-0.135	-0.135
Rku	2.186	2.186
Rp (um)	3.384	3.384
Rv (um)	3.495	3.495
Rz (um)	6.879	6.879
Rt (um)	7.635	7.635
Rc (um)	4.920	4.920
RSm(mm)	0.652	0.652
Rdq	0.025	0.025
Rk (um)	5.172	5.172
Rpk (um)	0.559	0.559
Rvk (um)	1.358	1.358
Mr1 (%)	5.625	5.625
Mr2 (%)	90.418	90.418

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

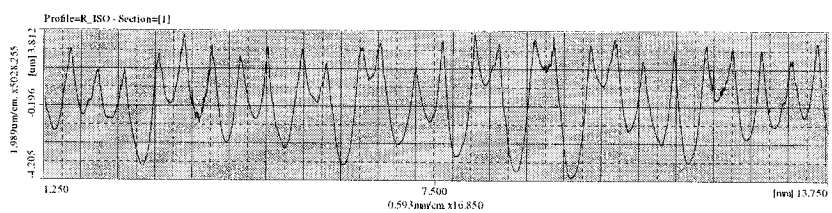
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc:	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

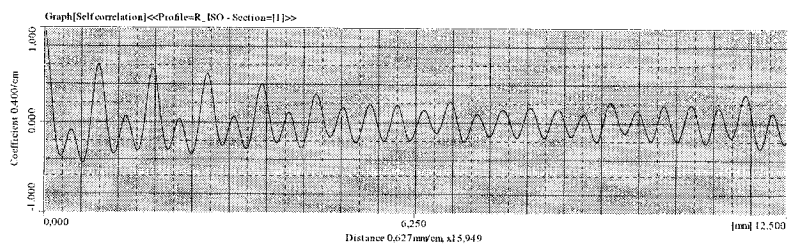
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000J14	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #2-D-AV

06/12/13





Rugosite Piece: #2-D-AR

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.211	1.211
Rq (um)	1.449	1.449
Rsk	0.220	0.220
Rku	2.135	2.135
Rp (um)	3.076	3.076
Rv (um)	2.665	2.665
Rz (um)	5.741	5.741
Rt (um)	6.543	6.543
Rc (um)	4.264	4.264
RSm (mm)	0.544	0.544
Rdq	0.022	0.022
Rk (um)	4.026	4.026
Rpk (um)	0.900	0.900
Rvk (um)	0.614	0.614
Mr1 (%)	13.348	13.348
Mr2 (%)	93.049	93.049

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

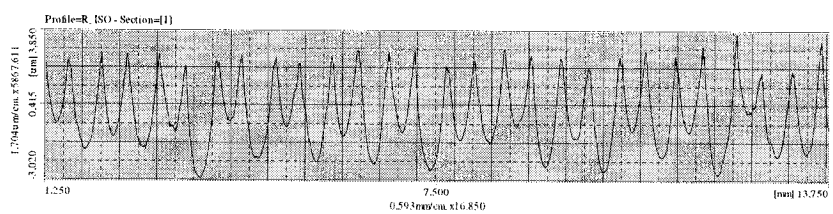
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evlin Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

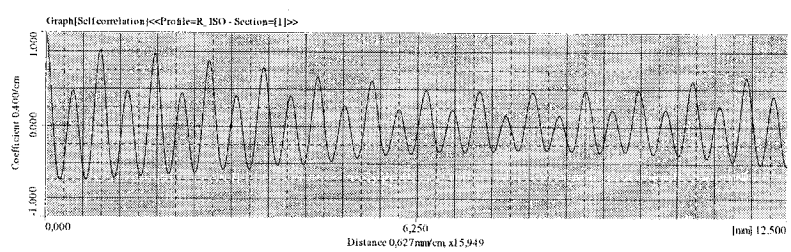
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4 mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #2-D-AR

06/12/13





Rugosite Piece: #2-C-AV

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1,254	1,254
Rq (um)	1,546	1,546
Rsk	0,387	0,387
Rku	3,023	3,023
Rp (um)	4,164	4,164
Rv (um)	2,965	2,965
Rz (um)	7,129	7,129
Rt (um)	11,364	11,364
Rc (um)	4,500	4,500
RSm (mm)	0,552	0,552
Rdq	0,031	0,031
Rk (um)	3,963	3,963
Rpk (um)	1,553	1,553
Rvk (um)	1,090	1,090
Mr1 (%)	13,462	13,462
Mr2 (%)	93,469	93,469

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

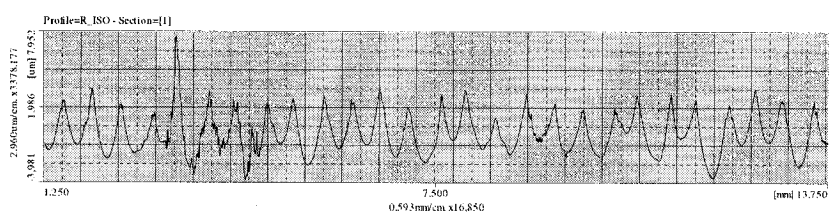
Kind of Profile:	R_ISO
Smplg Length(lle):	2,5 mm
No of Smplg(nle):	5
Lc	2,5 mm
Ls	0,008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12,5 mm
Pre-Travel:	1,25 mm
Post-Travel:	1,25 mm
Smooth Connection	Off

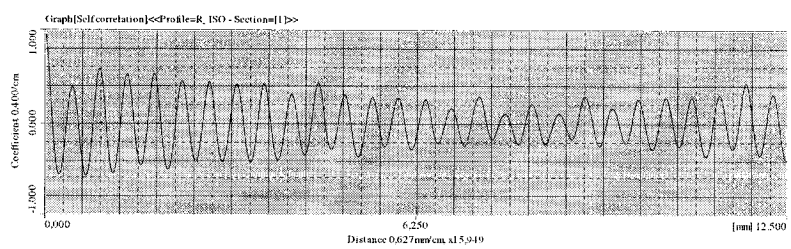
Measurement Condition

Measurement Length	15,0 mm	Measurement Start Position	0,0 mm
Column Escape	5,0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800,0 um
Speed	1,0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0,0 um
Pitch	1,0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #2-C-AV

06/12/13





Rugosite Piece: #2-C-AR

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.589	1.589
Rq (um)	1.873	1.873
Rsk	-0.234	-0.234
Rku	1.931	1.931
Rp (um)	3.534	3.534
Rv (um)	3.441	3.441
Rz (um)	6.975	6.975
Rt (um)	7.911	7.911
Rc (um)	6.075	6.075
RSm (mm)	0.865	0.865
Rdq	0.025	0.025
Rk (um)	5.172	5.172
Rpk (um)	0.669	0.669
Rvk (um)	1.094	1.094
Mr1 (%)	4.217	4.217
Mr2 (%)	83.799	83.799

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

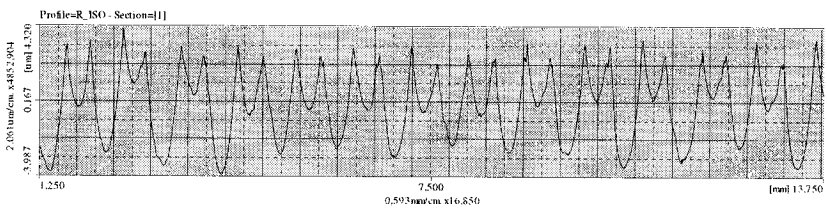
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

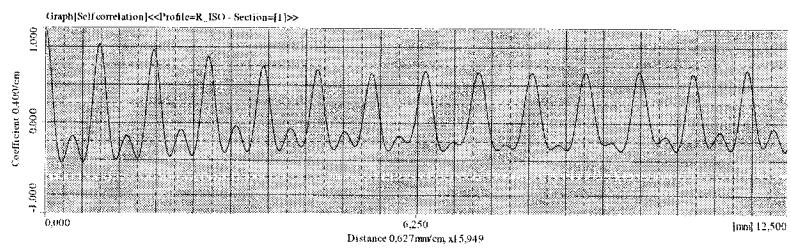
Rugosite Piece: #2-C-AR

06/12/13

Profile=R_ISO - Section=[1]



Graph[Selfcorrelation]<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #3-G-AV

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.404	1.404
Rq (um)	1.697	1.697
Rsk	0.526	0.526
Rku	2.583	2.583
Rp (um)	4.409	4.409
Rv (um)	2.995	2.995
Rz (um)	7.404	7.404
Rt (um)	8.805	8.805
Rc (um)	5.679	5.679
RSm (mm)	0.408	0.408
Rdq	0.044	0.044
Rk (um)	3.644	3.644
Rpk (um)	2.573	2.573
Rvk (um)	0.948	0.948
Mr1 (%)	20.925	20.925
Mr2 (%)	94.174	94.174

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evlun Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

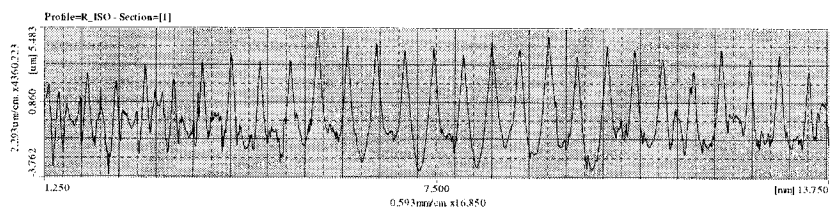
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

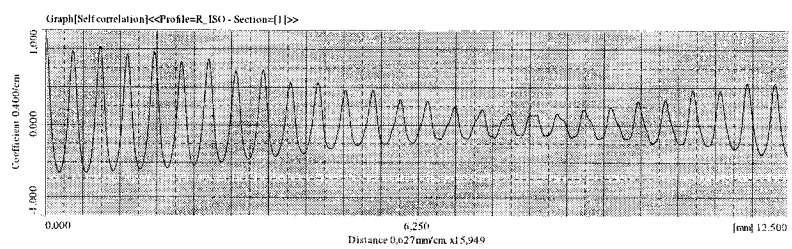
Rugosite Piece: #3-G-AV

06/12/13

Profile=R_ISO - Section=[1]



Graph[Self correlation]<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #3-D-AV

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.557	1.557
Rq (um)	1.851	1.851
Rsk	0.445	0.445
Rku	2.274	2.274
Rp (um)	4.478	4.478
Rv (um)	2.893	2.893
Rz (um)	7.371	7.371
Rt (um)	8.510	8.510
Rc (um)	5.662	5.662
RSm (mm)	0.410	0.410
Rdq	0.040	0.040
Rk (um)	4.441	4.441
Rpk (um)	2.410	2.410
Rvk (um)	0.532	0.532
Mr1 (%)	19.098	19.098
Mr2 (%)	97.218	97.218

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(lc):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

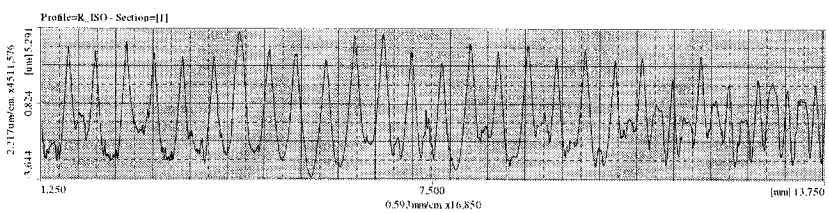
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	-0.00005 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AA(731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

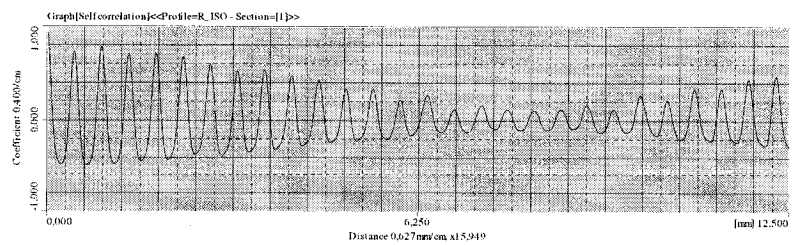
Rugosite Piece: #3-D-AV

06/12/13

Profile=R_ISO - Section=[1]



Graph[Self correlation]<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #3-C-AV

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.572	1.572
Rq (um)	1.846	1.846
Rsk	0.586	0.586
Rku	2.276	2.276
Rp (um)	4.211	4.211
Rv (um)	2.733	2.733
Rz (um)	6.943	6.943
Rt (um)	7.690	7.690
Rc (um)	5.813	5.813
RSm(mm)	0.431	0.431
Rdq	0.036	0.036
Rk (um)	3.862	3.862
Rpk (um)	2.807	2.807
Rvk (um)	0.276	0.276
Mr1 (%)	23.999	23.999
Mr2 (%)	96.199	96.199

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

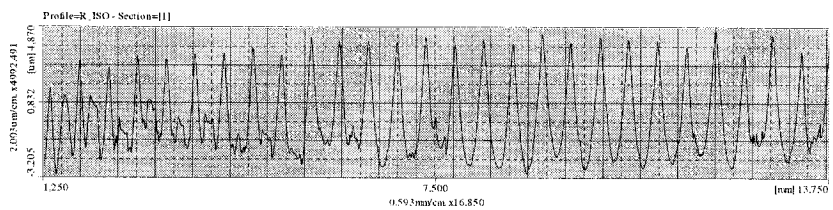
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4nN)	Stylus	Standard (12AA(731-12AA(355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

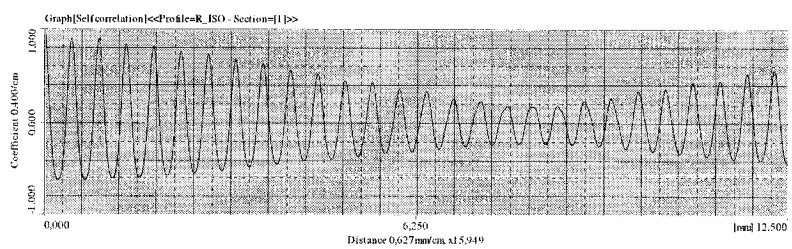
Rugosite Piece: #3-C-AV

06/12/13

Profile=R_ISO - Section=[1]



Graph[Selfcorrelation]<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #4-G-AV

06/12/13

Parameter Sum Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.223	1.223
Rq (um)	1.478	1.478
Rsk	0.301	0.301
Rku	2.468	2.468
Rp (um)	3.814	3.814
Rv (um)	2.751	2.751
Rz (um)	6.564	6.564
Rt (um)	9.544	9.544
Rc (um)	4.743	4.743
RSm (mm)	0.538	0.538
Rdq	0.029	0.029
Rk (um)	3.806	3.806
Rpk (um)	1.416	1.416
Rvk (um)	0.908	0.908
Mr1 (%)	15.582	15.582
Mr2 (%)	94.327	94.327

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

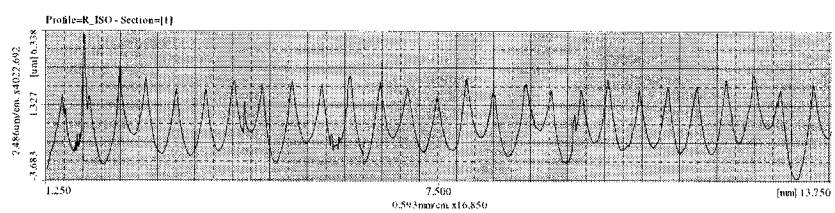
Kind of Profile:	R_ISO
Smplg Length(lc):	2.5 mm
No of Smplg(nle):	5
Lc:	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

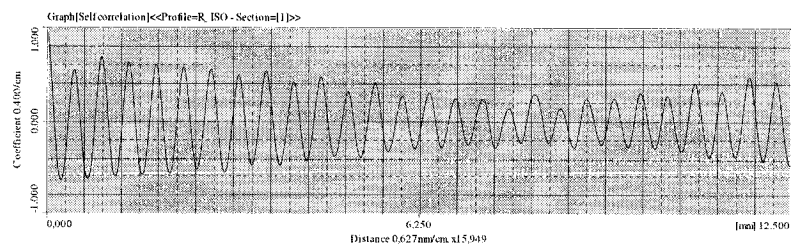
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #4-G-AV

06/12/13





Rugosite Piece: #4-G-AR

06/12/13

Parameter Sum/Table

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.642	1.642
Rq (um)	1.990	1.990
Rsk	0.522	0.522
Rku	2.505	2.505
Rp (um)	5.277	5.277
Rv (um)	3.320	3.320
Rz (um)	8.597	8.597
Rt (um)	11.153	11.153
Rc (um)	6.606	6.606
RSm(mm)	0.477	0.477
Rdq	0.047	0.047
Rk (um)	4.701	4.701
Rpk (um)	2.415	2.415
Rvk (um)	1.184	1.184
Mr1 (%)	18.870	18.870
Mr2 (%)	93.351	93.351

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

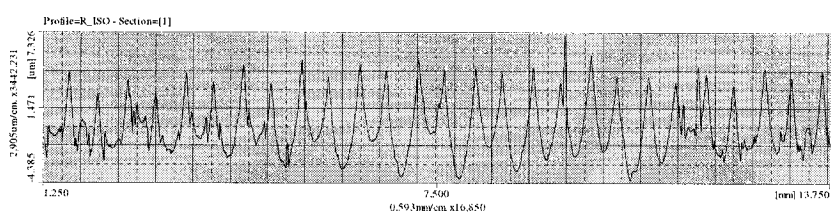
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evlu Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

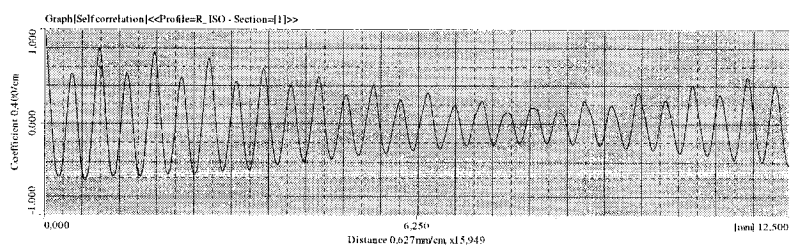
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AA(731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Arm compensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #4-G-AR

06/12/13





Rugosite Piece: #4-D-AV

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.200	1.200
Rq (um)	1.423	1.423
Rsk	0.436	0.436
Rku	2.280	2.280
Rp (um)	3.499	3.499
Rv (um)	2.211	2.211
Rz (um)	5.710	5.710
Rt (um)	7.283	7.283
Rc (um)	4.385	4.385
RSm (mm)	0.461	0.461
Rdq	0.023	0.023
Rk (um)	3.343	3.343
Rpk (um)	1.630	1.630
Rvk (um)	0.660	0.660
Mr1 (%)	20.307	20.307
Mr2 (%)	96.248	96.248

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

Kind of Profile:	R_ISO
Smplg Length(lc):	2.5 mm
No of Smplg(nlc):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

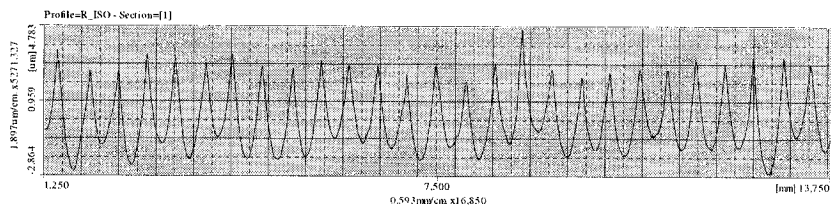
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

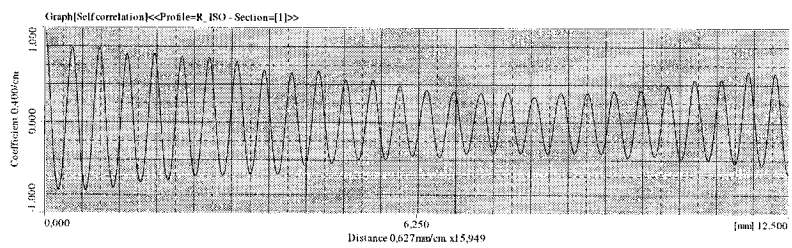
Rugosite Piece: #4-D-AV

06/12/13

Profile=R_ISO - Section=[1]



Graph[Selfcorrelation]<<Profile=R_ISO - Section=[1]>>



Rugosite Piece: #4-D-AR

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.396	1.396
Rq (um)	1.713	1.713
Rsk	0.362	0.362
Rku	2.653	2.653
Rp (um)	4.441	4.441
Rv (um)	3.278	3.278
Rz (um)	7.718	7.718
Rt (um)	9.439	9.439
Rc (um)	5.356	5.356
RSm (mm)	0.342	0.342
Rdq	0.054	0.054
Rk (um)	4.332	4.332
Rpk (um)	2.259	2.259
Rvk (um)	1.211	1.211
Mr1 (%)	13.352	13.352
Mr2 (%)	92.838	92.838

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

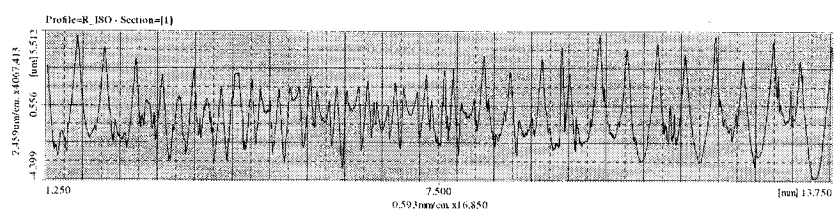
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

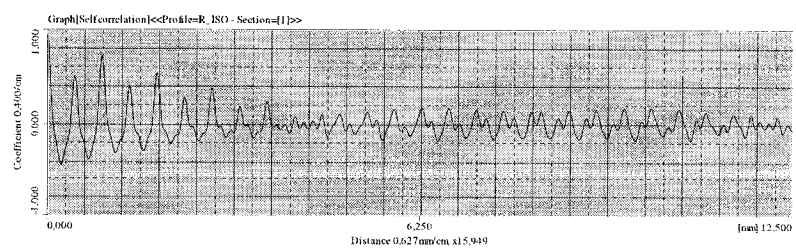
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #4-D-AR

06/12/13





Rugosite Piece: #4-C-AV

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.180	1.180
Rq (um)	1.388	1.388
Rsk	0.544	0.544
Rku	2.169	2.169
Rp (um)	3.221	3.221
Rv (um)	2.043	2.043
Rz (um)	5.264	5.264
Rt (um)	5.781	5.781
Rc (um)	4.362	4.362
RSm (mm)	0.459	0.459
Rdq	0.023	0.023
Rk (um)	3.246	3.246
Rpk (um)	1.601	1.601
Rvk (um)	0.356	0.356
Mr1 (%)	21.152	21.152
Mr2 (%)	95.742	95.742

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

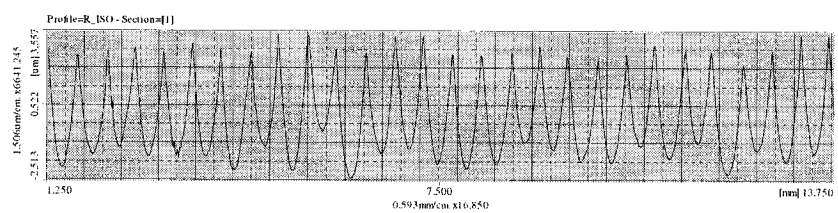
Kind of Profile:	R_ISO
Smplg Length(lc):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(ln):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

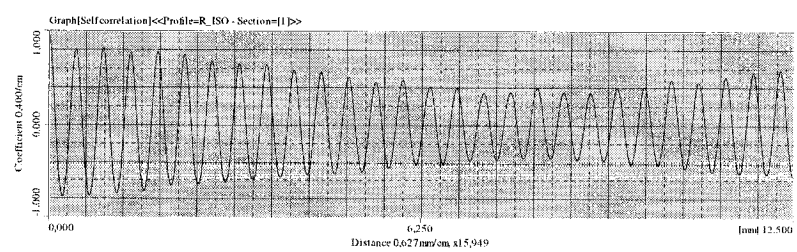
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #4-C-AV

06/12/13





Rugosite Piece: #4-C-AR

06/12/13

Parameter SumTable

	Profile=R_ISO - Section=[1]	Average Value
Ra (um)	1.408	1.408
Rq (um)	1.666	1.666
Rsk	0.217	0.217
Rku	2.309	2.309
Rp (um)	3.926	3.926
Rv (um)	3.332	3.332
Rz (um)	7.258	7.258
Rt (um)	8.816	8.816
Rc (um)	5.127	5.127
RSm (mm)	0.351	0.351
Rdq	0.050	0.050
Rk (um)	5.082	5.082
Rpk (um)	1.351	1.351
Rvk (um)	0.858	0.858
Mr1 (%)	7.224	7.224
Mr2 (%)	95.858	95.858

Evaluate Condition List<<Profile=R_ISO - Section=[1]>>

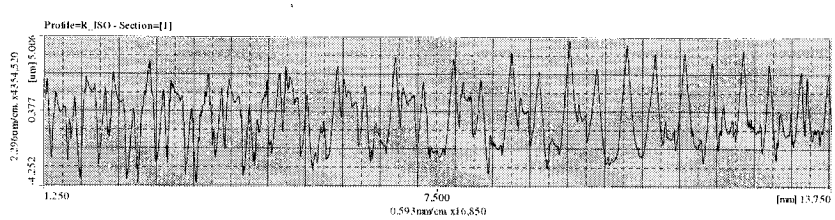
Kind of Profile:	R_ISO
Smplg Length(le):	2.5 mm
No of Smplg(nle):	5
Lc	2.5 mm
Ls	0.008 mm
Kind of Filter:	Gaussian
Evltn Length(lm):	12.5 mm
Pre-Travel:	1.25 mm
Post-Travel:	1.25 mm
Smooth Connection	Off

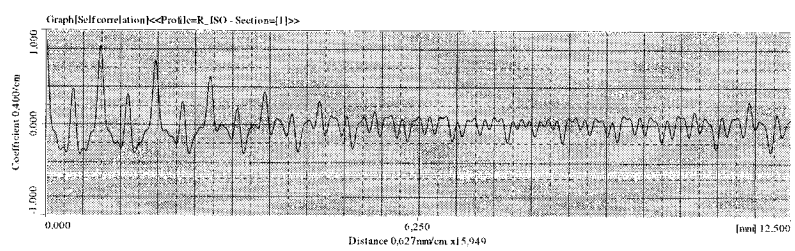
Measurement Condition

Measurement Length	15.0 mm	Measurement Start Position	0.0 mm
Column Escape	5.0 mm	Measurement Axis Escape	Return
Auto-Leveling	Off	Range	800.0 um
Speed	1.0 mm/s	R-Surface Auto-Measurement	Off
Over Range	Abort	Stylus Start Position	0.0 um
Pitch	1.0 um	Number of Points:	15000
Machine	SV-C4000H4	Measurement Axis	Drive Unit(100mm)
Detector	Detector(4mN)	Stylus	Standard (12AAC731-12AAB355)
Polar Reversal	Off	Straightness Compensation	Off
Armcompensation	Off	Stylus Radius Compensation	Off
Auto-Notch(+)	Off	Auto-Notch(-)	Off
Compensation Method	Off		

Rugosite Piece: #4-C-AR

06/12/13





APPENDIX II

MATLAB PROGRAMM BASE ON ALGORITHM 1.1

Printed by \hojat


```
%##########################################################################
%
%This program calculate the DKM and the jacobian matrix base on algorithm 1.1
%
%##########################################################################

clear;
clc;

syms X Y Z d 1
C=sym('C','real');
A=sym('A','real');
B=sym('B','real');
p=sym('p','real');

%Right hand DH parameters
thetar=[C-p/2 A+p/2 -p/4 0];

ar=[0 0 0 0];

br=[-d 0 Y 0];

alphar=[p -p/4 -p/2 -p/2];

thetal=[-p/2 0 0 p/2];

al=[0 0 0 0];

bl=[0 X Z+d -1];

alphal=[p/2 -p/2 0 0];

R=eye(3);
T=eye(4);

%Calculate right hand HTM

for m=1:size(thetar,2)

    lambdar=cos(alphar);
    mur=sin(alphar);

    Qr(:,:,m)=[cos(thetar(m)) -lambdar(m)*sin(thetar(m)) mur(m)*sin(thetar(m));
        sin(thetar(m)) lambdar(m)*cos(thetar(m)) -mur(m)*cos(thetar(m));
        0 mur(m) lambdar(m)];

    aar(:,m)=[ar(m)*cos(thetar(m));
        ar(m)*sin(thetar(m));
        br(m)];

    Ar(:,:,m)=[Qr(:,:,m)' -Qr(:,:,m)'*aar(:,m);
        0 0 0 1];

    Q(:,:,m)=Qr(:,:,m)';
    aa(:,m)=-Qr(:,:,m)'*aar(:,m);

    T=T*Ar(:,:,m);
end

%Calculate left hand HTM

for n=1:size(thetal,2)

    lambdal=cos(alphal);
    mul=sin(alphal);

    Ql(:,:,n)=[cos(thetal(n)) -lambdal(n)*sin(thetal(n)) mul(n)*sin(thetal(n));
        sin(thetal(n)) lambdal(n)*cos(thetal(n)) -mul(n)*cos(thetal(n));
        0 mul(n) lambdal(n)];
```



```
    aal(:,n)=[al(n)*cos(thetal(n));
        al(n)*sin(thetal(n));
        bl(n)];

    Al(:,:,n)=[Ql(:,:,n) aal(:,n);
        0 0 0 1];

    Q(:,:,m+n)=Ql(:,:,n);
    aa(:,m+n)=aal(:,n);
%Matrix T represents DKM
    T=T*Al(:,:,n);
end

%Calculate the Jacobian matrix

for i=1:m+n
    R=R*Q(:,:,i);
    RR(:,:,i)=R;
end

k=[0 0 -1]';
h=1;
%e(:,1)=[0 0 -1]';
for i=2:1:m
    e(:,i)=RR(:,:,h)*k;
    h=h+1;
end

for i=1:n
    e(:,i+m)=RR(:,:,h)*[0 0 1]';
    h=h+1;
end
e(:,1)=[0 0 -1]';
r(:,m+n)=aa(:,m+n);

for i=m+n-1:-1:1
    r(:,i)=aa(:,i)+Q(:,:,i)*r(:,i+1);
end

r(:,1)=r(:,1);

for i=m+n:-1:2
    r(:,i)=RR(:,:,i-1)*r(:,i);
end

for i=1:m+n
    BB(:,i)=cross(e(:,i),r(:,i));
end


for m=1:size(thetar,2)

    if thetar(m)=='A'||thetar(m)=='B'||thetar(m)=='C'||thetar(m)=='C-1/2*p'||the
tar(m)=='A+1/2*p'

        J1(:,m)=[e(:,m+1);
                 BB(:,m+1)];
    elseif br(m)=='X'||br(m)=='Y'||br(m)=='Z'||br(m)=='Z+d-1'||br(n)=='Z-1'||br(
n)=='Z+d'

        J1(:,m)=[[0 0 0]';
                 e(:,m+1)];
    end

end

for n=1:size(thetal,2)
```

```
    if thetal(n)=='A'||thetal(n)=='B'||thetal(n)=='C'||thetal(n)=='C-1/2*p'||the
tal(n)=='A+1/2*p'

        J1(:,m+n)=[e(:,m+n);
                   BB(:,m+n)];
    elseif bl(n)=='X'||bl(n)=='Y'||bl(n)=='Z'||bl(n)=='Z+d-l'||bl(n)=='Z-l'||bl(
n)=='Z+d'

        J1(:,m+n)=[[0 0 0]';
                   e(:,m+n)];
    end

end

t=sym([0 0 0 0 0 0]');
j=1;

for i=1:size(J1,2)

    if isequal(J1(:,i),t)==0

        J(:,j)=simplify(J1(:,i));
        j=j+1;
    end
end

T=simplify(T);

%The DKM of machine-tool

T=limit(T,p,pi);

%The Jacobian matrix

J=limit(J,p,pi);
```