

IMPROVING THE ACCURACY OF PARALLEL ROBOTS

THÈSE N° 1570 (1996)

PRÉSENTÉE AU DÉPARTEMENT DE MICROTECHNIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES TECHNIQUES

PAR

Peter VISCHER

Dipl. Masch.-Ing. ETH
originaire de Bâle (BS)

acceptée sur proposition du jury:

Prof. R. Clavel, directeur de thèse
Dr A. Codourey, corapporteur
Prof. M. Hiller, corapporteur
Prof. M. Troyanov, corapporteur

Lausanne, EPFL
1996

Gewidmet meinen Eltern und Michaela

Abstract

The main objective of this thesis is the accuracy improvement of parallel robots. Accuracy can be improved either by precise manufacturing and assembly or by calibration of each individual robot using a kinematic model which takes geometric deviations into account. The latter has the advantage of leading to low cost solutions but requires sophisticated modeling of the robot's structure which is usually considerably more complex than the derivation of its nominal model.

To substantiate the theoretical tools proposed in this thesis two examples of parallel structures are chosen. One of them is the Delta robot with three translational degrees of freedom whereas the second example is a novel structure called Argos having three rotational degrees of freedom. For experimental verification a mock-up was built for each of the two structures.

Four calibration steps, modeling, measurement, identification, and implementation are investigated. Investigations were restricted to static errors due to geometric deviations assuming rigid bodies.

First a formula is proposed which allows to calculate the number of independent kinematic parameters required for a complete model of a parallel structure. Then a systematic parameterization is introduced and applied to derive four calibration models, two for each example.

Two measurement devices are described which were built to determine the position and orientation (pose) of the end-effectors of the two robots. For the Delta robot two additional set-ups using no external (additional) measurement device are proposed.

For parameter identification different methods were tested by simulation. Calibration based on the implicit model is proposed as a standard method to calibrate parallel robots. Another calibration method is introduced, referred to as semiparametric calibration, which leads to low computational effort.

Fast solutions of the direct and inverse problems had to be found. For the first time all the solutions of the direct problem for the Delta robot were found by means of an algorithm introduced by Husty. In addition a fast numeric algorithm for the Delta's direct problem is proposed.

The main contribution of this thesis is the experimental verification of calibration methods to improve the accuracy of parallel robots. Using these calibration methods for the two robots, ARGOS and DELTA, between a three- to a twelve-fold improvement of accuracy was achieved and experimentally verified.

Résumé

Le but de cette thèse est l'augmentation de la précision absolue des robots parallèles. La précision peut être augmentée soit par une fabrication et un montage très soigneux, soit par calibration individuelle de chaque robot en utilisant un modèle cinématique plus complet que le modèle nominal du robot. La recherche se concentre sur le cas statique en tenant compte seulement des erreurs dues à la structure géométrique elle-même, celle-ci étant modélisée à l'aide de corps rigides.

Pour vérifier les outils théoriques proposés dans cette thèse deux exemples de structures parallèles ont été choisis. Il s'agit du robot Delta avec trois degrés de liberté en translation et d'une nouvelle structure, nommée Argos, conçue dans le cadre de ce travail; cette dernière comporte trois degrés de liberté en orientation. Une maquette de chaque structure a été construite. Elles montrent que les concepts de construction bon marchés proposés permettent d'obtenir une précision élevée. L'augmentation de la précision par calibration est ensuite traitée en quatre étapes: modélisation, mesure, identification et implémentation.

Une formule qui permet de calculer le nombre de paramètres indépendants pour un modèle complet d'une structure parallèle est proposée. Une approche systématique pour la paramétrisation est introduite et utilisée avec succès pour développer quatre modèles de calibrage, deux pour chaque exemple.

Deux systèmes de mesures ont été construits pour déterminer la position et l'orientation de l'organe terminal des deux robots. Pour le robot Delta deux arrangements supplémentaires n'ayant pas besoin d'un système de mesure externe sont proposés.

Différentes méthodes d'identification ont été simulées. La calibration basée sur le modèle implicite a été proposée comme méthode standard pour calibrer des robots parallèles. Une autre méthode intéressante est la calibration semiparamétrique qui diminue considérablement la complexité du calcul.

Pour l'implémentation des quatre modèles, il s'agit de trouver des solutions rapides pour les problèmes directs et inverses. Grâce à un algorithme proposé par Husty toutes les solutions du problème direct pour le robot Delta ont été trouvées. Pour le même problème, un algorithme numérique minimisant le temps de calcul a été développé.

La contribution principale de cette thèse est la vérification expérimentale des outils théoriques développés dans le cadre de ce travail. Les résultats expérimentaux pour l'ARGOS et le DELTA montrent une amélioration de la précision absolue entre un facteur trois et douze.

Remerciements

Je tiens à remercier toutes les personnes qui ont contribué à ma thèse. En particulier je tiens à remercier mon directeur de thèse, le Professeur Reymond Clavel, qui m'a proposé le sujet de thèse me permettant de faire connaissance avec les domaines de la robotique et de la microtechnique.

Je tiens aussi à remercier les corapporteurs, Messieurs les Professeurs Manfred Hiller et Marc Troyanov ainsi que Monsieur Alain Codourey pour la correction soignée du manuscrit et leurs propositions d'amélioration. Mes remerciements vont également Dieter Vischer, Boris Stevens et Michaela Kiess pour les corrections et à Heidi Banz pour le support administratif.

En plus je remercie chaleureusement Monsieur Willy Maeder, maître de construction, pour les nombreuses discussions menant à l'optimisation de la construction des maquettes utilisées pour la calibration. Je remercie également l'atelier mécanique de l'Institut de microtechnique (IMT), Messieurs Georges Perrenoud pour la fabrication d'une maquette et Jean-Jacques Crausaz pour l'aide aux travaux photographiques. Je souhaite ici mentionner et remercier spécialement Monsieur Laurent Rey, qui m'a non seulement fabriqué des cartes électroniques pour la lecture dynamique de mes encodeurs, mais qui a aussi consacré beaucoup de temps pour des discussions fructueuses.

Dans le cadre des projets de semestre, Messieurs Rodolfo Milos, Olivier Egloff et Denis Bubendorf ont contribué à cette thèse de différentes manières.

Ma gratitude va également à mes collègues de l'IMT, en particulier à Silvia Allegro, Wolfgang Andreasch, Roger Baumann, Christian De Graffenried, Olivier Demont, Lorenzo Flückiger, Max Hongler, Eric Pernet, Ivo Magnani, Attilio Vicario, Johannes Steinmetz, au Professeur Igor Kokcharov pour ses simulations concernant la méthode "simulated annealing" et Professeur Tigran Parikian pour ses conseils concernant la cinématique spatiale, ainsi qu'aux collaborateurs de l'Institut d'automatique, Olivier Chételat et Philippe Guglielmetti.

Finalement, toute ma gratitude à ma chère Michaela pour son encouragement et son soutien tout au long de cette thèse.

Ce travail a été rendu possible grâce à la bourse du conseil des écoles polytechniques fédérales: "Le doctorat dans l'autre EPF".

Notation

The notation used in this work is based on the textbook of Craig [89]:

1. Variables written in bold face represent vectors, matrices or tensors. Normal face variables are scalars. Because of extensive matrix calculation the dimension of vectors, matrices and tensors are indicated by "overstriking" the bold face variables once, twice or three times. Examples: a is a scalar, $\bar{\mathbf{a}}$ is a column vector if not specified otherwise, $\overline{\mathbf{A}}$ is a matrix, and $\overline{\overline{\mathbf{A}}}$ is a three dimensional tensor. $|\bar{\mathbf{a}}|$ is the Euclidean norm of vector $\bar{\mathbf{a}}$ and $|\overline{\mathbf{A}}|$ is the determinant of matrix $\overline{\mathbf{A}}$. An "overstrike" caret (^) indicates a measured value subjected to measurement noise.
2. Leading subscripts and superscripts indicate in which coordinate system a quantity is written. Examples: ${}^A\bar{\mathbf{p}}$ represents a position vector written in frame {A}, and ${}^A_B\overline{\mathbf{R}}$ is a rotation matrix which consists of the unit vectors of frame {B} written in frame {A}, relating vectors expressed in {B} to vectors expressed in {A}. $\{C\} = \{D\}$ indicates two frames having the same origin, which means that the transformation between these two frames is reduced to a pure rotation. *Rot3*, *Rot4*, *Trans4*, *Sw* and *Sh* are matrix functions defined in annex B.
3. Trailing superscripts indicate the inverse or transpose of a matrix. Since vectors are generally defined as column vectors a transposed vector is a row vector. Exceptions are the trailing superscripts n, o, i and *, which indicates nominal, accurate, identified and starting sets of parameters.
4. Trailing subscripts are not subjected to any strict convention but may indicate a vector component (e.g. x,y, or z), the number of the joint-link train (kinematic chain), the number for a measured point etc.
5. The notation for trigonometric functions may be shortened to: $\cos(\alpha) = c(\alpha) = c\alpha$

The different kinematic models are distinguished by the number of kinematic parameters which they contained. Example: Model 16 is a model containing 16 kinematic parameters.

Each of the seven *chapter* of this work is divided into *sections*, which are further divided into *paragraphs*. Therefore, the identifier 2.3 indicates the third section of chapter two, whereas the identifier 2.3.1 points to the first paragraph in the third section of chapter two.

Contents

Abstract.....	i
Résumé	iii
Remerciements	v
Notation	vii
Contents	ix
1. Introduction.....	1
1.1 Preliminaries and definitions	1
1.1.1 Industrial robots	1
1.1.2 Parallel robots	1
1.1.3 Static pose error	2
1.1.4 Methods for precise automation tasks	3
1.1.5 Direct, inverse and calibration problem.....	5
1.1.6 Kinematic calibration of robots	5
1.1.7 Conclusions.....	7
1.2 Motivation and objectives	7
1.3 State of the art	9
1.3.1 Introduction.....	9
1.3.2 Simulations	10
1.3.3 Experiments	13
1.3.4 Concluding remarks	15
1.4 Contributions.....	15
1.5 Report's outline	16
1.6 Examples of parallel robots.....	17
1.6.1 Introduction.....	17
1.6.2 The Stewart Platform	17
1.6.3 The Delta robot	18
1.6.4 The Argos structure	19
2. Conception	23
2.1 Introduction	23
2.2 Basic concepts for accurate robots	23
2.3 Mock-up of the Delta robot.....	25
2.3.1 Introduction.....	25

2.3.2	Construction.....	25
2.4	Mock-up of the Argos structure.....	28
2.4.1	Introduction.....	28
2.4.2	Construction.....	28
2.5	Conclusion	31
3.	Modeling.....	33
3.1	Introduction	33
3.2	Properties of a good model	34
3.2.1	Completeness	34
3.2.2	Proportionality	39
3.2.3	Calibration model for a serial SCARA robot	43
3.3	Models for the Stewart Platform.....	44
3.3.1	Introduction.....	44
3.3.2	Model 42.....	45
3.3.3	Conclusion	46
3.4	Models for the Delta robot.....	47
3.4.1	Introduction.....	47
3.4.2	Parameterization	48
3.4.3	Model 54.....	51
3.4.4	Model 24.....	52
3.4.5	The nominal model	53
3.4.6	Conclusion	54
3.5	Models for the Argos structure	55
3.5.1	Introduction.....	55
3.5.2	Parameterization	56
3.5.3	Model 27.....	59
3.5.4	Model 9.....	60
3.5.5	The nominal model	62
3.5.6	Conclusion	63
3.6	Error propagation in serial compared to parallel robots	63
3.6.1	Introduction.....	63
3.6.2	Results from simulation	63
3.7	Conclusion	66
4.	Measurement.....	67
4.1	Introduction	67
4.2	Set-ups for the Delta mock-up	69
4.2.1	Introduction.....	69

4.2.2	Position unit	69
4.2.3	Orientation unit	70
4.2.4	Full-pose measurement set-up	71
4.2.5	Set-up in another configuration	72
4.2.6	Plane/Spheres set-up	73
4.2.7	Short-cut set-up	74
4.3	Set-up for the Argos mock-up	75
4.3.1	Introduction	75
4.3.2	Position unit	75
4.3.3	Orientation unit	76
4.3.4	Full pose measurement set-up	77
4.4	Conclusion	78
5.	Identification	79
5.1	Introduction	79
5.2	The calibration problem	80
5.2.1	Introduction	80
5.2.2	The merit function	82
5.2.3	Unconstrained non-linear least-squares estimation	83
5.2.4	Conclusion	83
5.3	Simulation of the identification step	84
5.3.1	Introduction	84
5.3.2	A planar, slider-crank structure	84
5.3.3	Implicit calibration	88
5.3.4	Forward calibration	94
5.3.5	Inverse calibration	95
5.3.6	Linear calibration	96
5.3.7	Cascaded calibration	97
5.3.8	Semiparametric calibration	98
5.3.9	Optimal selection of measurement set	100
5.3.10	Conclusion	102
5.4	Parameter identification for the Delta robot	104
5.4.1	Introduction	104
5.4.2	Full position measurement and model 24	105
5.4.3	Full pose measurement and model 54	110
5.4.4	Plane/Spheres set-up and model 24	112
5.4.5	Short-cut set-up and model 54	114
5.4.6	Conclusion	115

5.5	Parameter identification for the Argos structure	116
5.5.1	Introduction.....	116
5.5.2	Full orientation measurement and model 9	118
5.5.3	Full pose measurement and model 27	120
5.5.4	Conclusion	122
5.6	Conclusion	122
6.	Implementation	123
6.1	Introduction	123
6.2	The four basic problems.....	124
6.2.1	Introduction.....	124
6.2.2	The general position problem	126
6.2.3	The general orientation problem.....	128
6.2.4	The general pose problem.....	132
6.2.5	The general joint angle problem	136
6.2.6	Conclusions.....	137
6.3	Solving model 24 of the Delta robot	137
6.3.1	The direct problem.....	137
6.3.2	The inverse problem	138
6.4	Solving model 54 of the Delta robot	139
6.4.1	Introduction.....	139
6.4.2	The direct problem of the set of equations G1	142
6.4.3	The inverse problem of the set of equations G1	142
6.4.4	The orientation problem of the set of equations $\Delta G2$	143
6.4.5	Conclusion	143
6.5	Solving the nominal model of the Argos structure	144
6.5.1	The direct problem.....	144
6.5.2	The inverse problem	146
6.6	Solving model 9 of the Argos structure	147
6.6.1	The direct problem.....	147
6.6.2	The inverse problem	148
6.7	Solving model 27 of the Argos structure	149
6.7.1	Introduction.....	149
6.7.2	The direct problem.....	149
6.7.3	The inverse problem	150
6.7.4	Conclusion	150
6.8	Conclusion	151

7. Conclusion	153
7.1 Summary	153
7.2 Future work	154
Annex A Additional figures.....	155
A.1 Generic derivation of spherical mechanisms	155
A.2 Solutions of the general pose problem	156
Annex B Vector and matrix functions	157
B.1 The Rot3 (Rotate)-function	157
B.2 The Sw (Swap)-function	157
B.3 The Sh (Shrink)-function	159
Annex C Non-linear least-squares estimation	160
C.1 Newton's method.....	160
C.2 Gauss-Newton method	163
C.3 Singular Value Decomposition (SVD)	165
C.4 Levenberg-Marquardt method	167
Annex D Solving non-linear systems of equations.....	169
D.1 Polynomial systems.....	169
D.1.1 Applying resultants to check for common roots	169
D.1.2 Application of determinants for solving polynomial systems	170
D.1.3 Application of resultants for solving polynomial systems	170
D.2 Transcendental systems.....	172
D.2.1 Use of the trigonometric identity	172
D.2.2 Use of resultants	173
References.....	175
Curriculum Vitae	187

1. Introduction

1.1 Preliminaries and definitions

1.1.1 Industrial robots

According to ISO/TR 8373 an *industrial robot* is a manipulator with several degrees of freedom, which is automatically controlled, reprogrammable, and has the ability to fulfill different tasks in industrial automation. Its mechanical part consists of a *base*, an *intermediate structure* and an *end-effector* connected with *joints*. Variable parameters of a joint are called *joint coordinates* which together form the *joint space*. A set of joint coordinates describes a *configuration* of the robot. A *kinematic*¹ *model* may be implemented into the robot *controller* linking the *joint coordinates* to the *world coordinates* used to describe the location of the end-effector within the *world space*².

1.1.2 Parallel robots

Most of today's industrial robots have a *single chain* of bodies as intermediate structure quite similar the human arm. These robots belong to the class of *serial robots*. If a heavy box has to be moved, humans tend to use both arms. This concept of *several chains working in-parallel* was also applied in robotics [Pollard 42, Hunt 83] leading to

¹ Kinematics is the science of motion (position, velocity, acceleration) not taking into consideration the forces and moments causing these motions. This work is limited to static considerations only, sometimes referred to as position kinematics or geometry. However, the more general term "kinematics" was adopted in order to indicate the relationship between geometric error analysis and velocity kinematics, shown by the appearance of Jacobian matrices, which are varying linear transformation matrices.

² This terminology is somewhat field dependent. Authors working in the field of *control theory* may use "operational space" [Guglielmetti 94], in *general kinematics* the term "Cartesian space" can be found [Craig 89], whereas in *automation* the space is sometimes referred to as "task space". However in the field of *manipulator calibration* the term "world space" is used by different authors [Whitney 86, Shamma 87, Mooring 91].

another class of so called *parallel robots*. Examples for parallel robots are given in section 1.6. Since several kinematic chains are involved, a parallel robot contains *closed loops* in its *topology*. It is inherent to closed loops, that some of the *joint coordinates* are *dependent*.

The analogy of a person moving a heavy box with both arms immediately shows the presence of dependent joint coordinates: In order not to drop the box during motion the relative distance between the hands must be kept constant. The movement of the arms is therefore *constrained by the closure condition to maintain the loop*. Thus, the movements of both arms are dependent on each other¹.

In literature various definitions may be found for parallel robots [Merlet 90, Clavel 91, Merlet 93a]. The following definition is employed for this thesis:

A robot with *dependent joint coordinates* caused by one or several *closed loops* in its topology belongs to the class of *parallel robots*.

1.1.3 Static pose error

A rigid body in space has *six degrees of freedom*, *three in position* and *three in orientation*. The combination of position and orientation is referred to as *pose* [Mooring 91]. The pose of the robot's end-effector is subjected to a *static error* after all transient parts (over- and undershoot, vibrations) have vanished. How fast the transient parts will vanish depend of the chosen type of trajectory and of the eigenfrequency of the mechanical structure [Demaurex 79].

According to ISO 9283 two criteria of performance, *repeatability* and *accuracy*, specify the static pose error of robots [Van Brussel 90, Engel 91, Mooring 91]. Slightly varying criteria are defined in VDI 2861 [VDI 88].

¹The analogy between a manipulator and an arm doesn't hold when examined in detail:

- a) One arm taken as analagon for a serial robot contains already closed loops (bone, agonist, antagonist).
- b) The analogy is poor for robots containing prismatic joints such as Cartesian robots.
- c) In contrast to non-redundant parallel robots a person moving a box with both arms is redundant in actuation (as two cooperating robots).

Repeatability is the ability of a robot to return to a previous pose.

Accuracy is the difference between the desired and actual pose.

Repeatability is influenced by back lash in the gear box, stick-slip, hysteresis, and so on, which are *stochastically changing events*. Accuracy is influenced by the difference between the real structure and the kinematic model used by the robot controller, which are *systematically occurring events*. The upper limit of accuracy is given by the repeatability as summarized in the figure below:

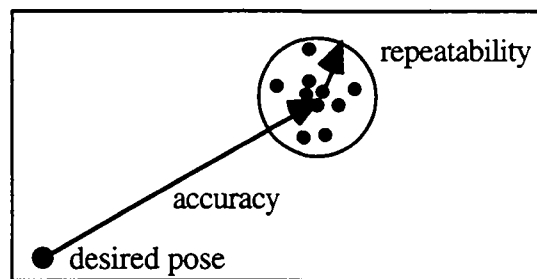


Fig. 1.1: Relation between repeatability and accuracy

Repeatability of a parallel robot was measured by Stevens [95] on a direct drive Delta robot [Codourey 91], yielded 0.2 millimeters in position and 0.1 degrees in orientation. Masory [94] has measured the repeatability of an other parallel robot, the Stewart Platform [Stewart 65], to be 1 millimeter in position and 0.2 degrees in orientation.

Repeatability is mainly improved by eliminating the source of the random errors by changing the mechanical concepts such as replacing a sliding by a rolling contact in order to avoid stick-slip or by preloading the gear box to avoid backlash.

1.1.4 Methods for precise automation tasks

Five different methods may be appropriate for precise automation tasks. The first two methods 1) and 2) *omit* the use of a kinematic model in the controller and are therefore only dependent on the robot's repeatability. Method 4) and 5) aim at improving the overall accuracy of the robot whereas method 3) is in between:

- 1) The classical method of robot programming is "*teach-in*". Thereby an operator guides the robot manually through a set of key configurations of the task. The joint coordinates of these configurations are stored in the robot controller. "Teach-in" has to be done on the robot itself. Thus, drawbacks of this method are that the robot's

productivity is blocked during "teach-in" time and each robot must be taught individually (programs are *not* interchangeable).

- 2) A second method uses *end-effector feedback* [Heeren 92] which requires a sensor measuring the pose of the end-effector. However, such sensors measuring in real-time the pose of a body in space with sufficient precision are not yet available on the market. Fast parallel robots such as the Delta robot causes additional problem for real-time end-effector feedback due to high accelerations of up to 40 g [Codourey 91].
- 3) Splitting the movement of the robot into a rough motion moving the end-effector *close to the target* and a *fine motion* moving the end-effector into target position is another method. For fine motion a passive *remote center compliance* (RCC) [Asada 88] or an additional active mechanical device dedicated to fine motions [Hollis 91] can be used. Other approaches are the local use of a *vision system* [Sakakibara 89], *contact force sensing* [Vischer 92] and high speed *random search* [Badano 93]. All these methods are task dependent, which can be a disadvantage.
- 4) A fourth method consists of adapting the robot to the kinematic model used by the controller. *Tolerances* of the mechanical parts, which is believed to be one of the major error sources concerning accuracy [Mooring 91, Schröder 93] are therefore specified *before manufacturing*. Such an approach was chosen by Zobel [94] to allocate tolerances to a Delta robot [Clavel 85]. However, smaller tolerances generally increase manufacturing costs. In chapter 2 some low-cost concepts are proposed to reduce tolerances.
- 5) A last method is to adapt the kinematic model to each robot individually by *calibration after manufacturing*. Deviations of each mechanical part are evaluated and stored in the controller. This process is the main subject of this thesis and treated in chapters 3 to 6.

The last four methods allows to *program* a robot *off-line*, for example by simulation in a CAD/CIM environment and download of the generated code to the robot. Because the robot itself is not needed for off-line programming, the production is not affected. This is an advantage compared to the "teach-in" method.

1.1.5 Direct, inverse and calibration problem

For a given set of joint coordinates, the *direct problem* corresponds to solving the kinematic model for the corresponding set of world coordinates (fig. 1.2). This is sometimes also referred to as *direct model*. For parallel robots it is generally difficult to solve the direct problem, whereas for serial robots it is easy and straight forward to establish.

For a given set of world coordinates, the *inverse problem* corresponds to solving the kinematic model for the corresponding set of joint coordinates (fig. 1.2), which is sometimes also referred to as *inverse model*. For parallel robots the inverse problem is generally easier to solve than the direct problem. For serial robots the inverse problem is always more difficult to solve than the direct one.

For given sets of joint *and* world coordinates, the kinematic *calibration problem* corresponds to solving the kinematic model for the corresponding set of kinematic parameters (fig. 1.2), which is sometimes also referred to as *identification*. For parallel as well as for serial robots this is a difficult task due to the *non-linear and overdetermined system* to solve.

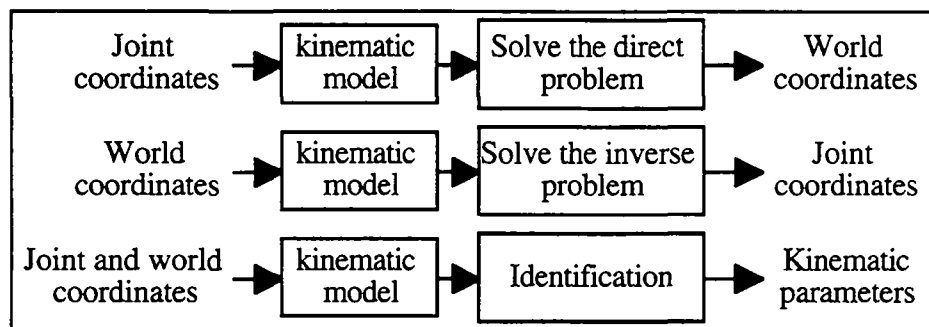


Fig. 1.2: The direct, inverse and calibration problem

1.1.6 Kinematic calibration of robots

Kinematic calibration is limited to errors of the mechanical parts only. Calibration is a modification of the software and won't affect any hardware of the robot. Figure 1.3 shows a general flowchart of *forward calibration* [Whitney 86], which is well suited for *serial robots*.

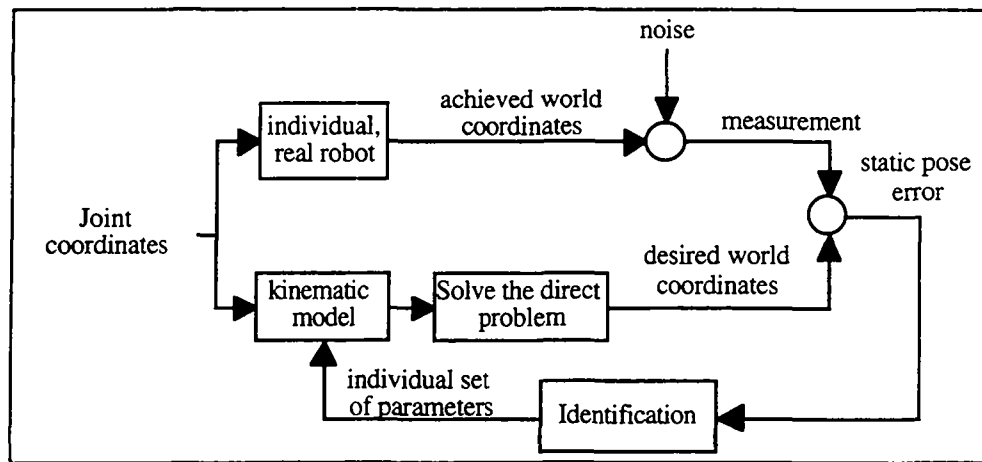


Fig. 1.3: Forward calibration process well adapted for serial robots.

However, as will be shown later in this work, forward calibration is not well suited for calibration of *parallel robots* because the direct problem has to be solved. Using *implicit calibration* is therefore proposed (fig. 1.4), where the solution of the direct problem is not required:

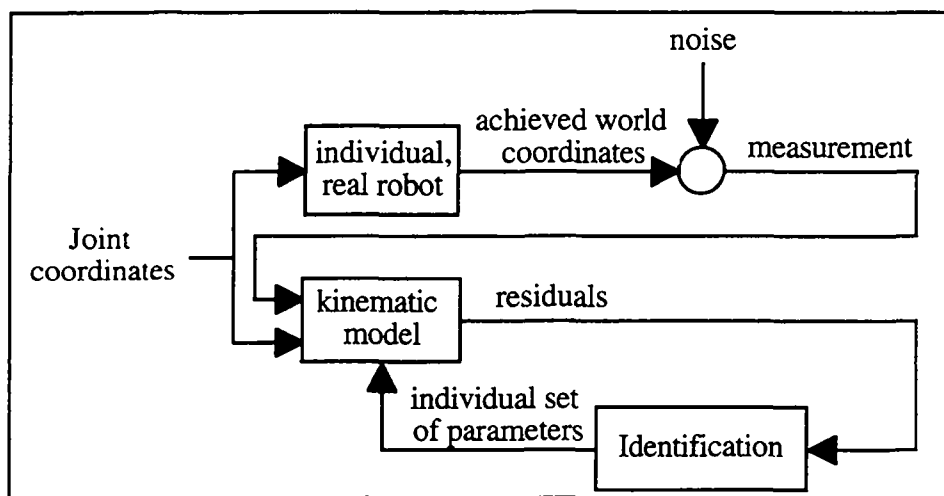


Fig. 1.4: Implicit calibration process well suited for parallel robots

According to Mooring [91] a calibration process consists of four different steps (fig. 1.5): *modeling*, *measurement*, *identification* and *implementation*. "Modeling" provides a suitable kinematic model for calibration. "Measurement" acquires a set of input data for the "identification", where a better fitting set of parameters is calculated. "Implementation" deals with the resolution of the direct and inverse problem of the calibrated model.

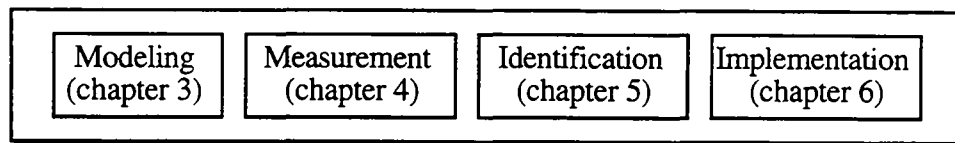


Fig. 1.5: The four steps of calibration

1.1.7 Conclusions

The main difference between serial and parallel robots is the presence of *dependent joint coordinates*, which makes the direct problem for parallel robots much harder to solve than for serial robots. This difference is the reason that standard calibration techniques developed for serial robots get very cumbersome or even fail for parallel robots. Techniques for the calibration of parallel robots have therefore to be adapted or newly developed.

1.2 Motivation and objectives

Most of today's industrial robots are still programmed by "teach-in". Changing this method for the less time consuming and cheaper off-line programming (paragraph 1.1.4) is of great interest. Unfortunately, simple downloading of the generated code to the real robot is not possible since the kinematic model doesn't reflect precisely the behavior of the real robot. The amount of deviation depends on the manufacturing tolerances of each individual robot, which has an influence on the static pose error. This explains the need for minimizing the static pose error, or in other words for *improving the accuracy*. Depending on the task, off-line programming is unavoidable. The programming of a robot for brain surgery [Flury 94] for instance has to be done off-line because the robot will be guided using information from a tomograph.

Parallel robots are generally regarded as being highly accurate due to the non-cumulative joints errors [Hunt 83, LeeK 88, Nguyen 91, Wang 92, Pernette 96]. First applications of parallel robots, where high precision and high stiffness is required, support this assumption. An example is a prototype built at the European Synchrotron Radiation Facility (ESRF) capable to place a load of 500 kilograms with an accuracy of *1 micrometer* in a cubic workspace of 4 centimeters side length [Merlet 93a]. In applications, where *high accuracy, high stiffness or high speed* is required, parallel structures replace serial topologies more and more. Examples are found in the machine tool industry [Schulz 94a, Schulz 94b, Geodetics 94, Honegger 96, Wiegand 96], optics [Zeiss 94] and microrobotics [Magnani 95, Pernette 96]. In the future parallel robots

may also be applied in metrology for high speed quality control [Schmidt 91, Clavel 91] or in automation for surface mounted device (SMD) technology.

The proposition to use parallel structures as milling machines (fig. 1.6) goes back to a remark of G.H. Meier and J. Tindale to an article of Stewart [65].

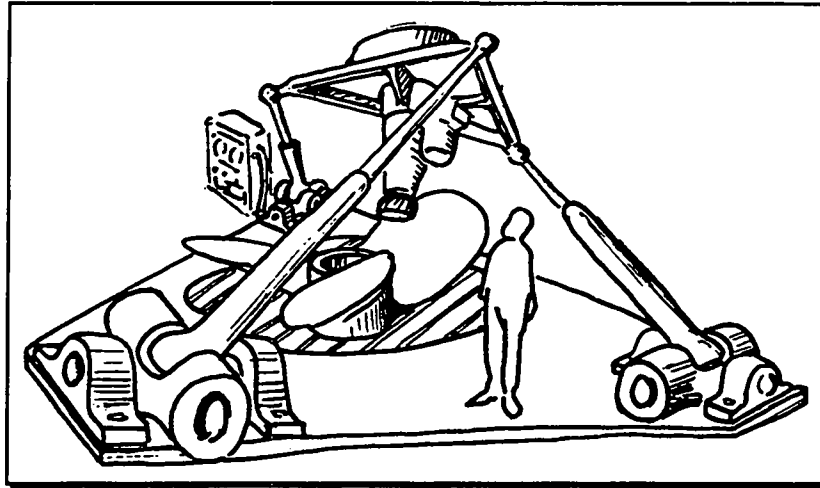


Fig. 1.6: Use of a Stewart Platform as a milling machine as proposed by Tindale

The theory for parallel topologies being more complicated and therefore less developed was one of the reasons why Meier's and Tindale's idea couldn't be realized successfully for more than thirty years.

This work aims at contributing to the theoretical as well as to the practical knowledge of accuracy improvement of parallel robots:

Minimization of the static pose error and thus improvement of the accuracy of parallel robots taking into account geometric deviations only is the principal goal of this work.

In this work only geometric deviations are considered which may be justified by the work of Judd [90], who found that 95% of the static pose errors are caused by geometric deviations. The need for the accuracy improvement of parallel robots is also demonstrated in figure 1.7, where a prototype of a parallel Delta robot 580 [Clavel 91] follows a square with a spring loaded pen on different heights in its workspace. Deviations of several millimeters are observed.

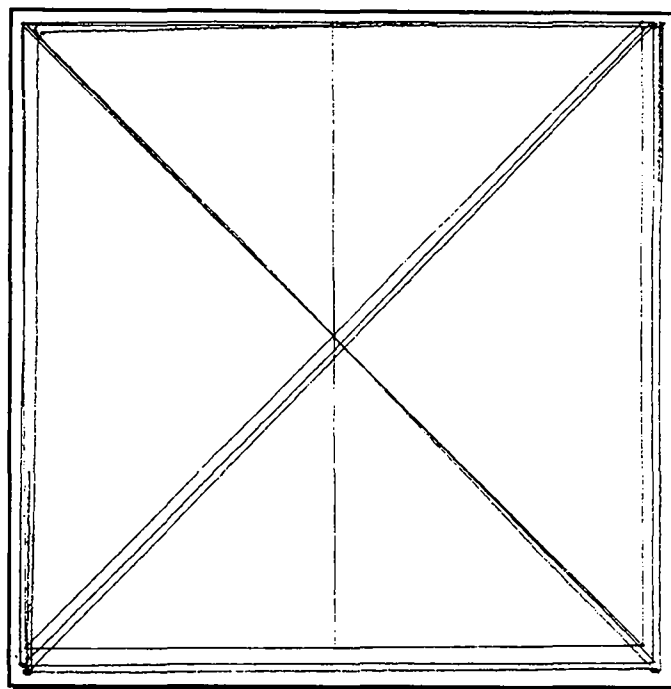


Fig. 1.7: Pose error of a prototype of a parallel Delta robot 580

1.3 State of the art

1.3.1 Introduction

As opposed to serial robots, calibration methods for parallel robots are not covered well in Literature. For the calibration of serial robots some standardized methods exist. Books [Kim 87, Mooring 91, Schröder 93, Bernhardt 93], software packages [Craig 92, Schröder 94], various good reviews, excellent work [Sugimoto 85, Hayati 85, Whitney 86, Roth 87, Everett 87, Veitschegger 88, Hollerbach 89, Khalil 89, Spur 89, Judd 90, Kozakiewicz 90, Renders 91] and PhD-thesis [Payannet 85, Stone 87, Veitschegger 87, Tanner 90, Flury 94] are available.

Much less work¹ has been done on calibration of parallel robots. No standardized calibration method has been proposed yet.

¹ About a dozen of papers of ten different groups

1.3.2 Simulations

Everett [88a, 89] and Lin [89] proposed a calibration procedure for different planar and spatial single-loop structure. They suggested to model closed-loops with two types of equations. The first type is a sequence of multiplications of homogenous matrices describing the *open loop structure* (serial) between base and end-effector, whereas the second type of equations contains the closure or *constraint equations* for the closed-loops. The constraint equations were added to the open loop equations by means of *Lagrange multipliers*. The resulting *unconstrained optimization problem* was solved by the Gauss-Newton method. With an assumed measurement noise of 25 micrometers and 10.3 arcseconds and a maximum deviation of the initial guess of 10% their calibration procedure could successfully identify the 24 parameters of a 2R2SR¹-single loop. The introduced Lagrange multipliers increase the number of variables to identify, which may be a disadvantage.

Toyama [89] proposed to use *screw algebra* for calibration of the Stewart platform. They introduced *dual-quaternions* in order to represent *three screw operators* required to describe the six kinematic parameters of an *extended Denavit-Hartenberg* parameterization [Denavit 55]. An *inverse calibration* was performed which is *decoupled for each joint-link train* using the *steepest descent method* to identify 36 parameters.

Zhuang [91] proposed for the calibration of a Stewart platform not to move the leg which parameters should be identified in order to collect special sets of error equations. The difference between two error equations of the same set neither contains the transducer offset nor the transducer readings (leg length). Their method allows to cascade the calibration problem into the identification of 36 parameters (attachment points of the S-joints) followed by the identification of the remaining 6 transducer offsets. Such a kind of calibration, which bases on *manipulations of the error equations*, is referred to as *cascaded calibration*. For simulation different levels of uniformly distributed measurement noise of 200 - 1000 micrometers and 7.2 - 36 arcseconds were assumed. Sets of 7 - 16 randomly distributed measurement points were

¹R indicates a revolute joint and S a spherical joint. The loop is built of five rigid bodies including the base. It has two degrees of freedom not counting the isolated degree of freedom of the ballbar (2S). The notation is further explained in paragraph 1.6.3.

chosen for each leg. The Gauss-Newton method based on *singular value decomposition* (SVD) was used to solve the non-linear optimization problem, which converged typically within three iterations. They concluded that the error of the orientation measurement has a strong influence on the quality of the identified parameters. Twelve measurement points seem to be enough to identify six parameters. The approach of Geng [94] is closely related to this work. It investigates the number of legs to be kept at a fixed length during data acquisition. Innocenti [95] developed a polynomial solution for the parameter identification problem outlined by Zhuang. In order to obtain a determined system of equations Innocenti used eight measurement points for seven parameters. In Zhuang [96] self-calibration of parallel robots is proposed. They simulated the identification of 30 kinematic parameters required to describe a Stewart Platform with no arbitrarily chosen base and moving frame. To get information for calibration, they added three and six supplementary joint transducers, respectively. Measurement noise of 100 - 500 micrometers and 21 - 103 arcseconds was assumed. Using the Levenberg-Marquardt algorithm an accuracy improvement of at least one magnitude could be gained within three iterations. However, redundant joint sensing must be provided, which may be expensive to install on each individual robot. A further disadvantage is that the base frame as well as the moving frame cannot be located arbitrarily. The origin of the base frame is in the center of one of the six universal joints at the proximal end of the legs. The problem of locating a work piece relative to this physically inaccessible base frame is obvious.

Ananthanarayanan [92] proposed to use the *bisection method* [Press 89] to find the root of a nonlinear, determined¹ system of equations, which he used to describe the parameter identification problem. However, no results are given for the introduced planar single-loop structure.

Wampler [92] pointed out that the measurement step in calibration is not restricted to measurements of the end-effector's pose but may also be performed by measurements of a passive joint coordinate. This was demonstrated successfully by simulating the

¹It is generally accepted that more measurement points than unknown parameters have to be acquired [Schröer 93]. (as a rule of thumb about twice as much [Zhuang 91]) Thus, the problem of parameter identification is not that of finding the root for a determined nonlinear system of equations but of minimizing an appropriate merit function based on the residuals of an *overdetermined* system of nonlinear equations.

calibration of a planar multi-loop structure using the Gauss-Newton method. A thirty micrometer and 2.4 arcminutes level of measurement noise and 34 randomly distributed measurement points for the identification of 11 parameters was used. Furthermore the addition of passive constraints for calibrating robots *without* using *external measurement devices* was proposed.

Wang [92, 93] and Masory [93] simulated *forward calibration*, the standard method for serial robots, of a Stewart platform. Using the Gauss-Newton method the 42 parameters could be identified within four to five iterations and within 6 - 7.5 *hours of calculation time* on a SUN 3/260. Taking 20 randomly distributed measurement points in the workspace with an added measurement noise of 61 micrometers and 10 arcseconds, they have shown that a fifteen- to twentyfold accuracy *improvement* is possible. It was further concluded that the influence of the *errors* of the *passive joints* are *neglectably small* compared to other geometric deviations and that the accuracy of parallel robots is of the *same order of magnitude* as for a comparable serial robot.

Kugiumtzis [94] simulated the calibration of a parallel robot equipped with a passive 4R joint-link train. The calibration model contained 30 parameters to identify. Application of the Gauss-Newton method showed severe problems of rank deficiency of the *identification Jacobian*, indicated by *high singular values* of the *pseudoinverse*, due to seven *almost linearly dependent* parameters. After removing these seven parameters found by manual inspection, identification could be successfully simulated with a noise level of hundred micrometer and six arcminutes and thirtyfive measurement points. The accuracy of the position was improved by a factor of twelve.

Ropponen [95] simulated a slightly inaccurately manufactured Stewart platform in order to map the resulting accuracy. The latter was based on an analytically derived identification Jacobian of the inverse model between error of world coordinates, encoder offsets and geometric deviations.

Jagadeesh [95] simulated the calibration of a single loop structure using a calibration model which takes into account errors in the kinematic parameters of the linkage, non-ideal gears and deflections due to different payloads. The calibration of the structure was simulated with a set of 72 simulated measurement points subjected to three different payloads leading to 216 end-effector positions for the 39 parameters to identify. Furthermore they proposed a two-step calibration procedure identifying first the 27 parameters required to model non-ideal gears and the errors of the kinematic linkage and in a second step the 12 parameters describing the deflection model. In a

third approach, identification of the 14 non-ideal gear parameters and the 13 kinematic parameters were separated leading to a three-step calibration procedure. For the first step the *single joint method* described in Mooring [91] and Judd [90], based on the work of Stone [86] was used. The first two methods worked well, decreasing the maximal position error of 8 millimeters to about 8 micrometers within *five to seven hours* of calculation time on a PC 486. The third method was less successful with an improvement of the accuracy to 0.5 millimeters. *No measurement noise* was added, but to generate the measurement points a more complicated deflection model was used than for calibration.

1.3.3 Experiments

Zobel [93] proposed a calibration model for a Delta robot containing 18 parameters. In order to identify these parameters a premeasured fixture (precision plate) for *full position measurement* with six touch points was used. The fixture could be placed in three different positions on the base plate of the robot. In a first experiment three parameters were identified using three error equations and it was shown that an error of ten millimeters could easily be identified. To confirm the experimental results the three *parameters* were also *directly measured* for comparison with the values obtained by calibration.

Hollerbach [93] calibrated a multi-loop structure with six degrees of freedom by using a premeasured fixture with twelve different locations allowing full pose measurement. Furthermore, a calibration was performed *without* using *external measurement devices*. Their multi-loop structure is all-in all equipped with nine sensors in order to be able to calculate a unique solution for the direct problem. Thus, the *internal sensors* provide three redundant measurements, which were used for calibration. 18 parameters were identified using the Levenberg-Marquardt method. Simulations showed that the initial guess could deviate up to ten percents from the accurate parameter set. They concluded that if no external measurement devices is used the convergence of the identification algorithm to a non-trivial solution *depends strongly* on the *selected measurement points*. Similar results were presented by Nahvi [94] for a redundant parallel robot with three degrees of freedom. Nahvi [96] addressed the question of choosing the optimal set of measurement points. It was concluded by comparison that three existing observability indices don't work satisfying. They proposed a new index referred to as *noise amplification index*, which is defined to be the smallest singular value divided by the condition number of the identification Jacobian. Application of this index to a redundant, spherical, parallel robot, yielded smaller standard deviations for the

estimated parameters of measurement sets with *high* noise amplification indices than of the ones with *low* noise amplification indices. They concluded that a high noise amplification index improves the robustness of the parameter estimation.

Wampler [95] calibrated a closed loop structure based on the implicit model, which will be referred to as *implicit calibration*. They repeated the calibration experiments of Hollerbach [93] confirming in general their results. In addition they identified fifty parameters of a Stewart platform by means of eleven internal measurement devices. Six of them were linear transducers to measure the length of legs. On one leg all five passive joint coordinates were measured by means of wired potentiometers in order to obtain a unique solution for the direct problem. Thus, the five redundant measurements and twenty measurement points will lead to hundred error equations for the fifty parameters to identify.

Zhuang [95] calibrated a Stewart platform using a theodolite for *full pose* measurement. Measurement with the theodolite had a pose accuracy of about 127 micrometer, which was sufficient compared to the repeatability of 2.5 millimeters. A set of twenty points were collected of which fifteen were used for identification and five for validation of the result. *Inverse calibration* using the Gauss-Newton method was performed and an *analytically differentiated* identification Jacobian established. The resulting *accuracy* almost reached the level of the *repeatability*.

Maurine [96] proposed a recalibration procedure for a Delta robot [Clavel 91] based on displacement measurement of a single Laser sensor (triangulation). They stated that a calibrated robot moved to a new work cell has to be recalibrated with respect to its environment and that the offsets of the joint transducers must be newly identified. To identify nine parameters they proposed a two-step method. In a first step a plane is precisely located parallel to the x-y plane of the base frame and a first set of six parameters is identified with this set-up. In a second step small cylinders are arranged in a circle on this plane and the remaining three parameters identified. Their approach resembles the plane/spheres set-up proposed in this work (paragraphs 4.2.6 and 5.4.4). However, there is a major difference: No external measurement device is needed for the plane/spheres set-up. Simulation showed the accuracy improvement to be not very sensitive to the number of measurement points acquired. Simulations were performed with 200 to 600 micrometers of measurement noise in order to show the robustness of the proposed method. For the experimental part of their work they added a third step by identifying first the orientation of the plane as well as the location of the cylinders. They identified *different* sets of parameters *depending* on the *initial values* for the

iterative non-linear least square algorithm. This confirms the *problem of multiple minima* reported for the plane/spheres set-up in paragraph 5.4.4.

1.3.4 Concluding remarks

To the author's knowledge these are all the publications available on the subject of calibration of parallel robots. Their number is small compared to the several thousands on calibration of serial robots.

Some authors, whose work is based on simulations, assume a low and often unrealistic noise level of their simulated orientation measurements. In this work high resolution Laser encoders with a maximum deviation of ± 25 arcseconds per revolution are used (chapter 2). Measurements of the three orientation angles of a body in space with even higher accuracy is very difficult and expensive using current technology.

1.4 Contributions

Listed below are various aspects of this work which are thought to contribute to the current research on the improvement of the accuracy of parallel robots:

1. *Two examples* of parallel structures, which are not as well explored as the Stewart platform, are chosen to be *investigated* with respect to *accuracy improvement*.
2. *Mock-ups of the two robots* were built aiming at verifying some of the proposed *concepts for accuracy improvement*.
3. One of these two examples is an *novel design* for a parallel spherical structure with three degrees of freedom called *Argos*. A generic deviation of *Argos* with two degrees of freedom leads to another original design called *PantoScope*¹.
4. A *simple formula* allowing the calculation of the *minimal number of parameters* for a complete calibration model is proposed. Two carefully chosen parametric calibration models for each of the two examples are introduced.
5. For each of the two examples a *measurement system* was *built* and some additional interesting measurement set-ups are proposed.

¹Its basic design can be found in annex A.1. The structure was further developed by Baumann [96] to a force feedback manipulator for laparoscopic surgery.

6. It is shown that *accuracy improvement of parallel robots is indeed possible*. The use of *implicit or semiparametric calibration* leads to a gain in speed and accuracy.
7. For *real-time applications an iterative algorithm* solving the direct and inverse problem of the calibration models is proposed.

The *main contribution* of the thesis consists of the *experimental verification* of the *proposed methods* to improve the accuracy of parallel robots.

1.5 Report's outline

Concepts which minimize certain mechanical tolerances to lower the modeling effort are described first (chapter 2) and two mock-ups of the Delta and Argos robot are presented.

The following four chapters deal with the calibration which consist of four steps, these are modeling, measurement, identification and implementation.

In chapter 3 some general properties of a good calibration model are introduced. Several parametric models for the Delta and the Argos robots are proposed.

In chapter 4 measurement set-ups for each of the two examples were elaborated based on a thorough investigation of commercially available systems. Promising propositions for industrial calibration of the Delta robot are presented.

Chapter 5 deals with the identification of the parameters of the proposed models. Suitable calibration methods are evaluated by simulation on a single-loop structure. These methods are successfully applied to identify the kinematic parameters for both, the Delta and Argos structure.

Chapter 6 shows how the implicit, identified models can be solved for the direct and the inverse problem. In particular efforts are made to obtain real-time solutions for the Delta robot.

In chapter 7 results are summarized and an outlook is presented on topics in the field of accuracy improvement of parallel robots which would be interesting to investigate in the future.

1.6 Examples of parallel robots

1.6.1 Introduction

In most Literature on parallel robots the Stewart Platform is employed as an examples of a fully parallel, non-redundant manipulator with six degrees of freedom. In this work it will also be used as an examples. However, some of the problems encountered in accuracy improving of parallel manipulators cannot be demonstrated with the Stewart Platform.

Some errors in the pose of the end-effector which occur for manipulators with less than six degrees of freedom cannot be influenced by its actuators. These non-influenceable errors are imposed by the mechanical structure of the robot and cannot be corrected by a calibration procedure. Taking for instance a SCARA robot with its four degrees of freedom, it is easy to see that the remaining, non-influenceable two degrees of freedom correspond to the perpendicularly of the end-effector with respect to its base.

To take the considerations above into account, two further examples are added, the first is the Delta robot with three degrees of freedom in position. The second example is a novel parallel structure named Argos with three degrees of freedom in orientation.

Having no common degree of freedom the Delta robot and the Argos structure are somewhat contrary, whereas the Stewart Platform joins these two structures representing a general parallel structure having all possible degrees of freedom in space. The latter will mainly be used for comparison with existing work.

1.6.2 The Stewart Platform

The "*Stewart (Gough) Platform*" (fig. 1.8) is the most often used name for the *first parallel robot with six degrees of freedom* [Stewart 65, Fichter 86]. The end-effector is connected to its base by six identical kinematic chains. Each chain consists of a spherical joint (S) attached to the base plate, a linear actuator (P) and another spherical joint (S) attached to the end-effector adding up to a 6[SPS] structure. In such a construction, each joint-link train could revolve around its longitudinal axis which is referred to as an *isolated degree of freedom*. To avoid this spinning motion, most designers replace one of the two spherical joints by a universal joint (U). A joint-link train of a Stewart platform is sometimes simply referred to as "leg".

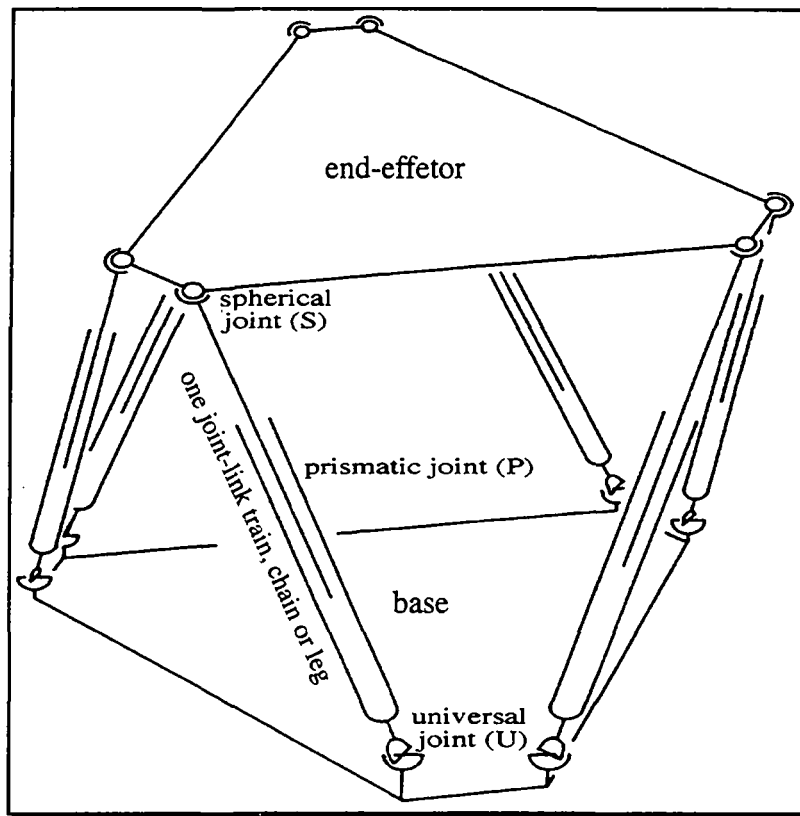


Fig. 1.8: Stewart Platform having six degrees of freedom [Fichter 86]

1.6.3 The Delta robot

The *Delta* robot (fig. 1.9) is a $3[R(2S/2S)]^1$ parallel manipulator with three degrees of freedom whereby the movement of the end-effector is a pure translation in space [Clavel 85]. The concept keeps the end-effector in a parallel position with respect to the base, which has been termed a *spatial parallelogram* [Clavel 91]. The end-effector is connected to the base by three identical kinematic chains. These chains have a tree-structure and are composed of a revolute joint (R) attached to the base whereas the next element, the "*arm*", branches out into two rods by means of two spherical joints (2S).

¹The structural notation $3[R(2S/2S)]$ is interpreted as follows: Start at the inner most bracket (2S/2S), which is interpreted as two bodies lying in parallel equipped with two spherical joints (S) each. Continue with the next bracket $[R(2S/2S)]$, which is interpreted as a branching body equipped with one revolute joint (R) and two S-joints. This is the description for one joint-link train. Three of them connect the base to the end-effector. The base is always on the left hand side of the formula, whereas the end-effector is on the right hand side.

Each of the six rods called "*forearms*" is then connected to the end-effector by another spherical joint (S).

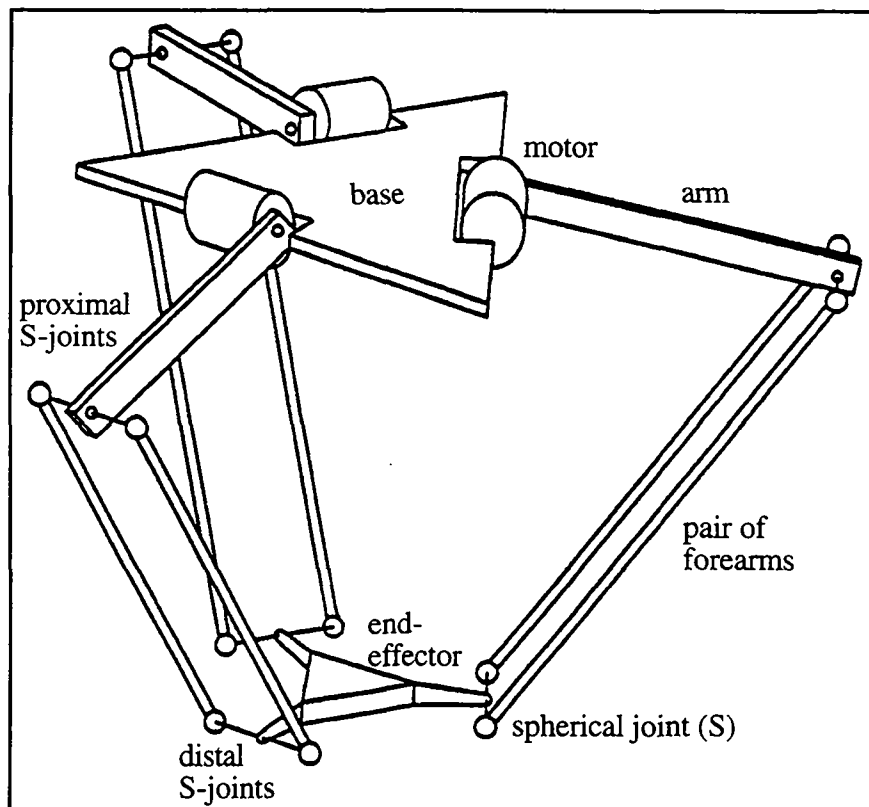


Fig. 1.9: Delta robot with its three translational degrees of freedom

1.6.4 The Argos structure

The *Argos* structure (fig. 1.10) is a *novel* $3[R(2R/2S)S]$ design of a rotative parallel manipulator with three degrees of freedom [Vischer 95]. A generic derivation of spherical mechanisms can be found in annex A including a version of the *Argos* structure with only two degrees of freedom named *PantoScope*.

The end-effector is connected to its base by three identical kinematic chains. Each chain is attached to the base plate by a revolute joint (R) with its axis pointing to the virtual center of rotation. This axis carries a pantograph. A pantograph is a planar parallelogram equipped with four revolute joints in its corners. In order to avoid static over-determination in space, it is possible to replace one pair of R-joints by a pair of S-joints as shown in figure 1.10. The long end of the pantograph's distal link describes a sphere around the virtual center of rotation. An S-joint links the pantograph and the end-effector.

Two kinds of motorizations are possible. The motor can either actuate the first R-joint (fig. 1.10) or by means of conical gearwheels the second R-joint. The first solution is easier to carry out, but the pantograph is subjected to bending stress. In the second case the pantograph works in a push-pull mode, which is generally preferable. However, a detailed study of the second case had shown that the mechanism *remains in a singularity* right at its nominal position [Vischer 95]. This is not the case for the first possibility of motorization, which will therefore be further discussed.

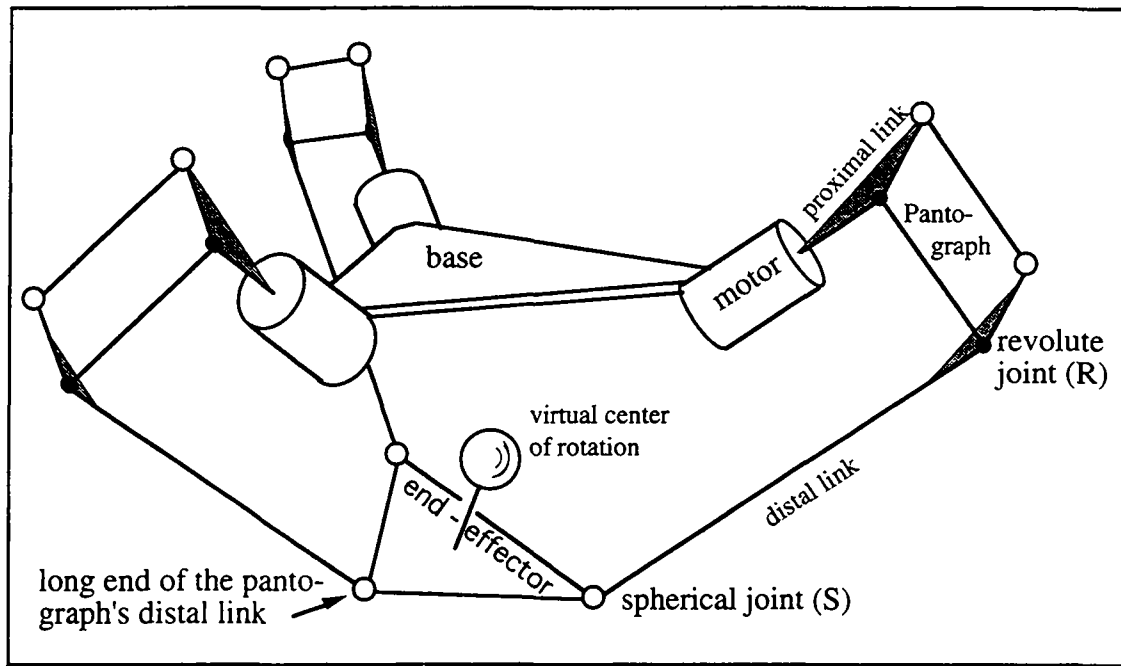


Fig. 1.10: Argos structure with its three rotational degrees of freedom

A criterion for the mobility (MO) of spatial multibody mechanisms can be found in Hunt [78] and Clavel [91]. It is a modified Grübler criterion:

$$MO = \sum MO_i - 6 \cdot b_o \quad (1.1)$$

where MO_i stands for the degrees of freedom of each joint of the mechanism and b_o for the number of independent loops.

Applying eq. 1.1 to the Argos structure shows that for an S-pair only five instead of six degrees of freedom have to be counted. This is due to the fact that the connecting rod can be turned around its longitudinal axis without influencing the pose of the end-effector representing an isolated degree of freedom. For the calculation one of the S-joints of an S-pair can be considered as an universal joint (U) blocking off the isolated degree of freedom:

$$MO = 3 \cdot (1 + 2 + 5 + 3) - 6 \cdot 5 = 3 \quad (1.2)$$

However, no information is given about the nature of the degrees of freedom to be expected, e.g. whether they are rotative or translative. This is due to the lack of information on the arrangement of the joints relative to each other. Therefore eq. 1.1 must be applied with care or rather with extended knowledge about the arrangement of the joints in the mechanisms under investigation. In case of planar or spherical mechanisms the factor of 6 has to be replaced by a factor of 3. However, eq. 1.1 fails for example when applied to Bennet's linkage [Hunt 78], which consists of a spatial loop of four non-parallel revolute joints (R), still having a mobility of one and not of minus two as predicted by eq. 1.1.

2. Conception

2.1 Introduction

Allocating little or *no restrictions* to the tolerances for the mechanical parts of a robot would *lower its manufacturing costs*. After manufacturing, all of the deviations could be identified by an appropriate calibration procedure and used to control the robot accurately. Unfortunately, this is not possible for two reasons:

- If the robot has less than six degrees of freedom, some pose errors cannot be influenced by its actuators and therefore not be corrected by the software. This is the case for both of the chosen examples. For the Delta robot the orientation of the end-effector is imposed by the structure whereas for the Argos structure it is the position of the virtual rotation center.
- If the passive joints of several degrees of freedom are not designed as precisely as possible the computational difficulty of a calibration process becomes enormous. Spherical joints are fairly easy to manufacture precisely. Assuming perfect spherical joints the number of parameters to identify drops from 138 to 54 for the Delta robot (chapter 3). Identification and implementation are consequently greatly simplified.

In this chapter the conception and construction phases of the two mock-ups is described with the aim of obtaining *high accuracy* in the end-effector pose especially *in the non-controllable world coordinates*.

2.2 Basic concepts for accurate robots

Concepts presented here are based on the use of a *common machine park*. For instance milling, turning and grinding are possible whereas other techniques such as laser cutting or electro-polishing were excluded in order to minimize the manufacturing costs of the two unmotorized mock-ups.

In the conception phase of parallel structures not only the questions of how to produce one single, accurate kinematic chain arise, but also the question of how to align precisely the chains relative to one another. For spatial parallel structures the positioning of three or six joint-link trains is typical.

The following concepts were used to manufacture the two mock-ups:

- *Spheres* as well as *cylinders* can be manufactured very precisely and quite cheaply. This may be explained by historical reasons such as the use of both elements in bearings. These two mechanical elements are therefore used whenever possible. An examples is the construction of the spherical joints (S) of both the Argos and the Delta mock-up.
- Cylinders are fixed to the mock-up in *V-groves*. This offers not only the advantage of well defined contact lines between cylinder and V-grove, but also location of two V-groves relative to each-other is generally easier than of two holes. Cylinders as well as V-groves can easy be modified to get contact points instead of contact lines.
- V-groves offer the advantage that two parallel axes can be manufactured in *one clamping*, which reduces the error of the parallelism to the milling machine error.
- Assembly *fixtures* are used in order to avoid allocate tolerances to each single mechanical part.
- *Hyperstatism* is *avoided* in order not to generate internal forces deforming the structure. An example is the concept of a hole, a V-grove and a plane to fit two parts together. The hole fixes the position of the parts with respect to each other whereas the V-grove and the plane fix the orientation. This is applied for the clamping of the base of the Delta mock-up to the measuring machine by three solenoids having solid state joints.
- To control thermal expansion, parts of the Delta mock-up such as the forearms and the arms are built of *Invar*. Invar is a nickel based alloy with a tenfold smaller thermal dilatation factor (10^{-6} K^{-1}) as compared to ordinary steel.

Apart from the claim for accuracy, simplicity of the mechanical parts of the prototype is important to keep manufacturing costs low. Further constraints are a large workspace, pushing the singularities to the border of the workspace, simplicity of the nominal model, possibility for different measurement set-ups, possibility of motorization, a modular system and so on.

2.3 Mock-up of the Delta robot

2.3.1 Introduction

The main effort of the conception phase for the Delta robot consists in *maintaining* the *spatial parallelogram* as *perfect* as possible since orientation errors in the end-effector pose cannot be corrected by calibration. This imposes the S-joints to be as perfect as possible and the connecting line between the proximal S-joints, as well as between the distal S-joints, must be as parallel as possible to the motor axis (fig. 1.9). In paragraph 3.4.6 a set of 6 parameters per main joint-link train (P1) disturbing this parallelism is detected. Minimization of these six parameters is therefore the main goal of this section besides the design of perfect S-joints.

2.3.2 Construction

Figure 2.1 shows a design worksheet of the mock-up of the Delta robot. Laser encoders Canon-M1 with 50'000 physical increments per revolution were chosen to measure the motor angles. Their specifications are:

Measurement volume (mechanical stoppers on the mock-up) -30° to 135°

Resolution 6.5"

Accuracy (accumulated error per revolution) 25"

Only one of the joint-link trains is represented stretched out in figure 2.1

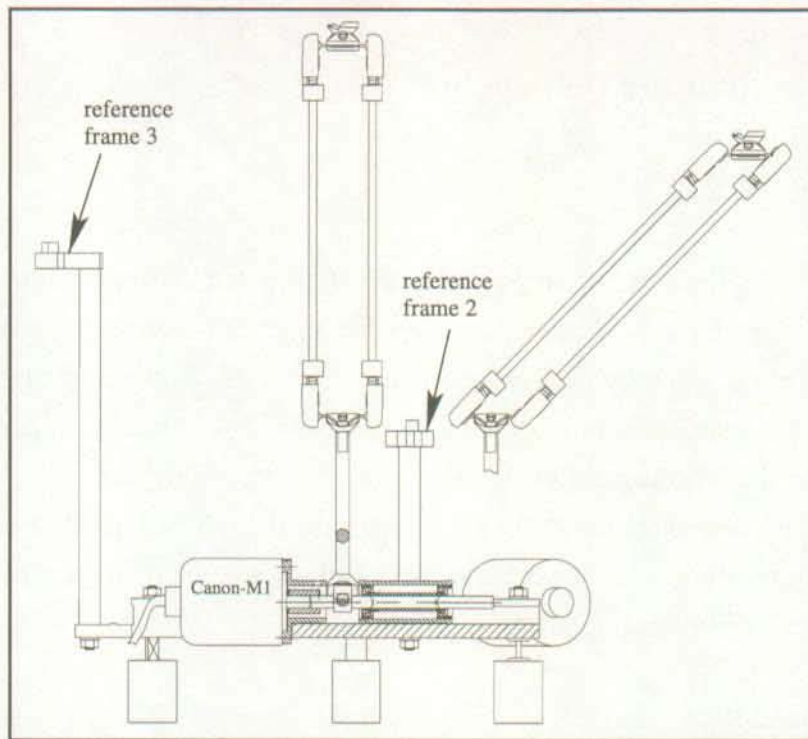


Fig. 2.1: Design worksheet of the Delta mock-up.

Figure 2.2 shows in detail how the cylindrical housings containing the motor axes are clamped to the base plate according to the cylinder / V-grove concept.

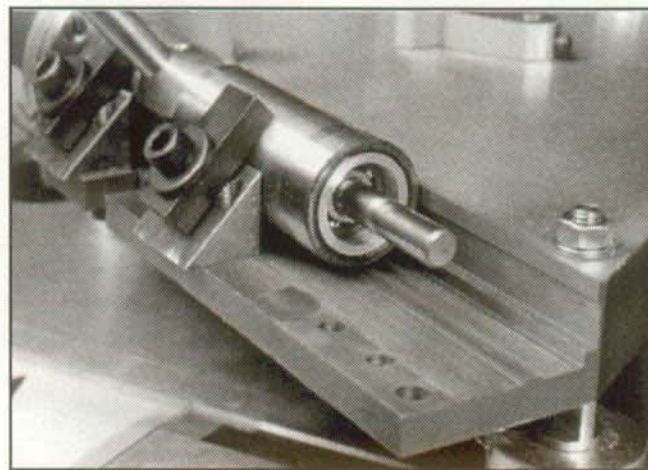


Fig. 2.2: The cylinder / V-grove concept applied to the motor axis

Figure 2.3 shows a further application of a proposed concept: The end-effector was screwed to the base during manufacturing. The two V-groves indicated by arrows are milled in the same clamping to get the two V-groves as parallel as possible. The V-grove indicated by the left arrow builds the support for the distal S-joints at the end-effector, whereas the V-grove on the base supports the motor axis. As remarked in the

introduction of this section (paragraph 2.3.1) these two lines have to be parallel in order to maintain the space parallelogram. The rather complicated shape of the end-effector is required for the orientation measurement device proposed in paragraph 4.2.3.

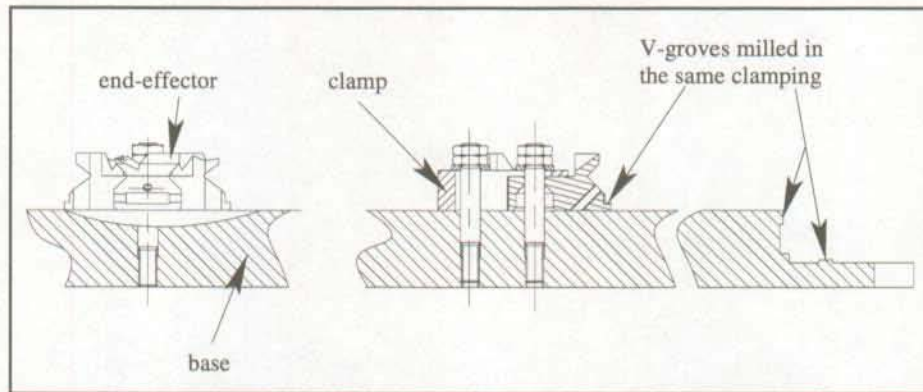


Fig. 2.3: Milling in one clamping of two V-groves for the motor axis and the distal S-joints.

Figure 2.4 shows how ball-socket joints can be produced very precisely. They consist of two precision parts, namely a sphere and a hollow cylinder, which is honed inside. The cylinder slides on the sphere with its interior circle. The contact force is established with a spring. Magnets or vacuum may be used instead.

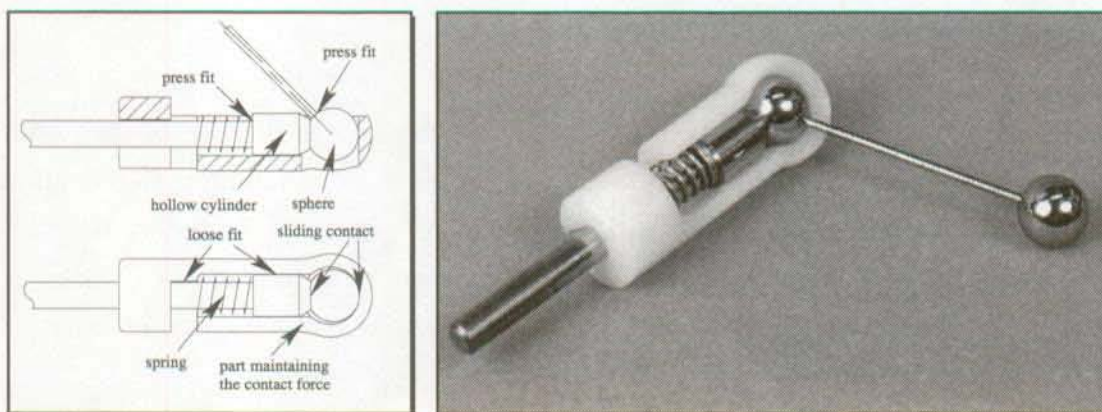


Fig. 2.4: Precise ball-socket joint (S) with an access angle of over ± 45 degrees

Figure 2.5 shows the fully assembled mock-up in a hanging position as the industrial version of the Delta robot.

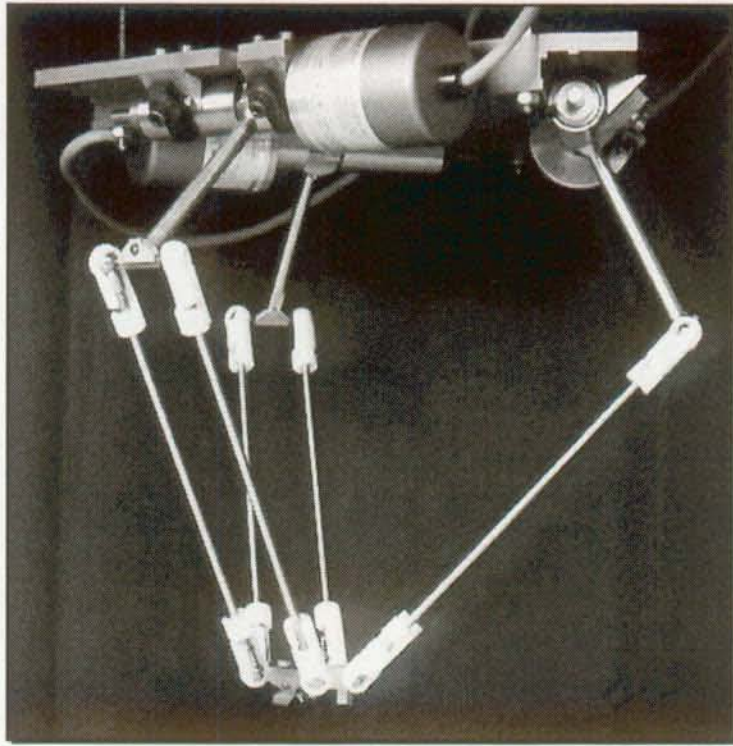


Fig. 2.5: The Delta mock-up

2.4 Mock-up of the Argos structure

2.4.1 Introduction

In the conception phase of the Argos structure deviations of the position of the virtual center must be minimized since they cannot be corrected by calibration. This requires again perfect S-joints. Furthermore, the pantographs (fig. 1.10) have to describe a perfect circle with their distal link. The motor axes have to be located in the planes of the pantographs and pointing to the virtual center. The three S-joints must be arranged on the same sphere around the virtual center.

2.4.2 Construction

Figure 2.6 shows a design worksheet of the mock-up of the Argos structure. The same Laser encoders as for the Delta mock-up are used to measure the motor angles (paragraph 2.3.2).

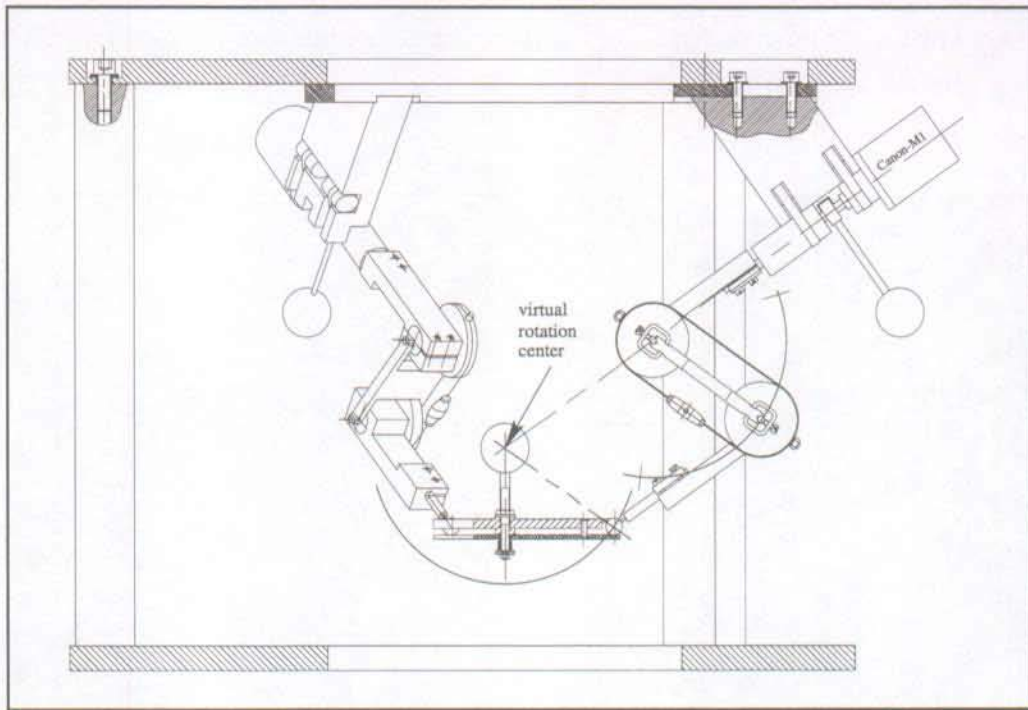


Fig 2.6: Design worksheet of the Argos mock-up [Bubendorf 96].

Figure 2.7 shows in detail how the housing of the motor axis is fixed to the support by means of a double V-groove and a hose band clip.

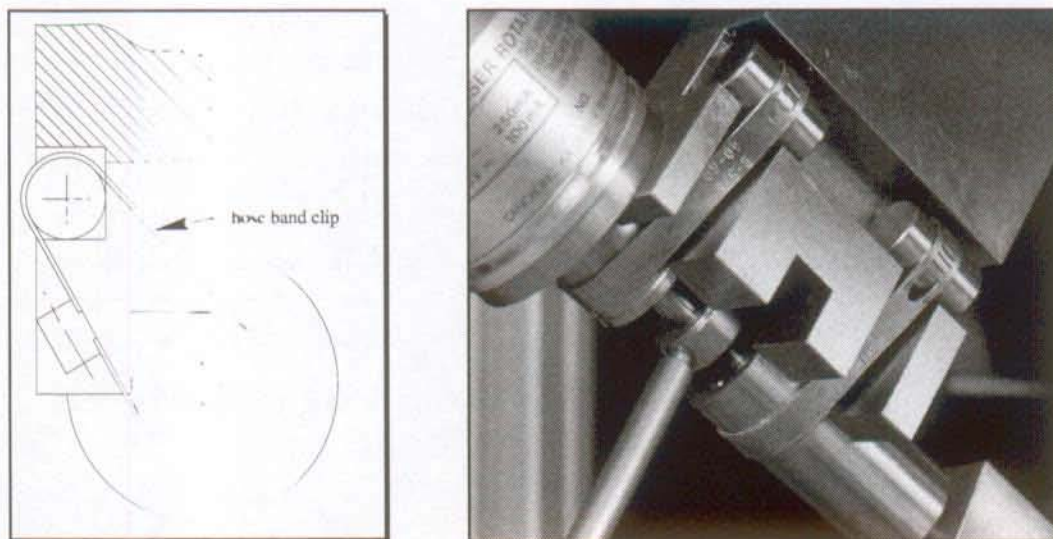


Fig. 2.7: Fixation of the housing of the motor axis by a double V-groove and a hose band clip.

Figure 2.8 shows a *steel cable pantograph*. Compared to a conventional design it has the advantages to require only two instead of four revolute joints and the stiffness remains constant within the full circle. The conventional design suffers of a singularity with zero stiffness when the parallelogram joins the antiparallelogram [Dijksman 76].

Constant stiffness is advantageous for the Argos structure because the pantographs have a large mobility range of over ± 60 degrees. To increase stiffness the steel cable can be replaced by a steel strip.

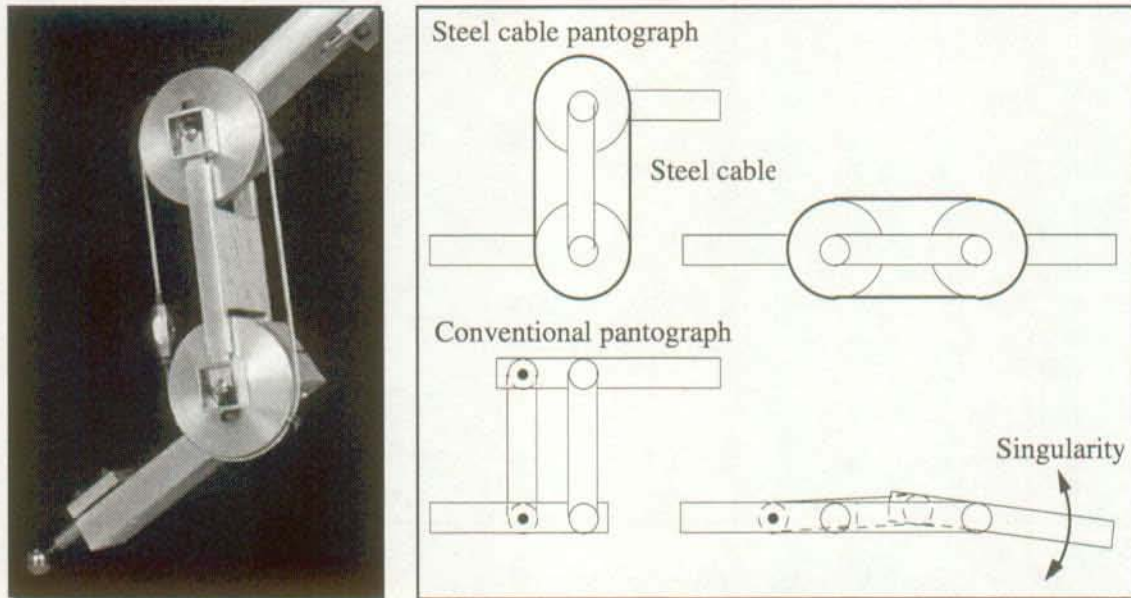


Fig. 2.8: A singularity-free pantograph with a steel cable. To support the axes of the pantograph an extra large H-shaped part was designed to absorb torsional stress.

The left hand side of figure 2.9 shows how the proximal link is aligned parallel to the distal link by means of a U-shaped fixture. The right hand side of figure 2.9 shows the alignment procedure of the supports of motor axes with respect to the virtual rotation center of the structure using two additional fixtures.

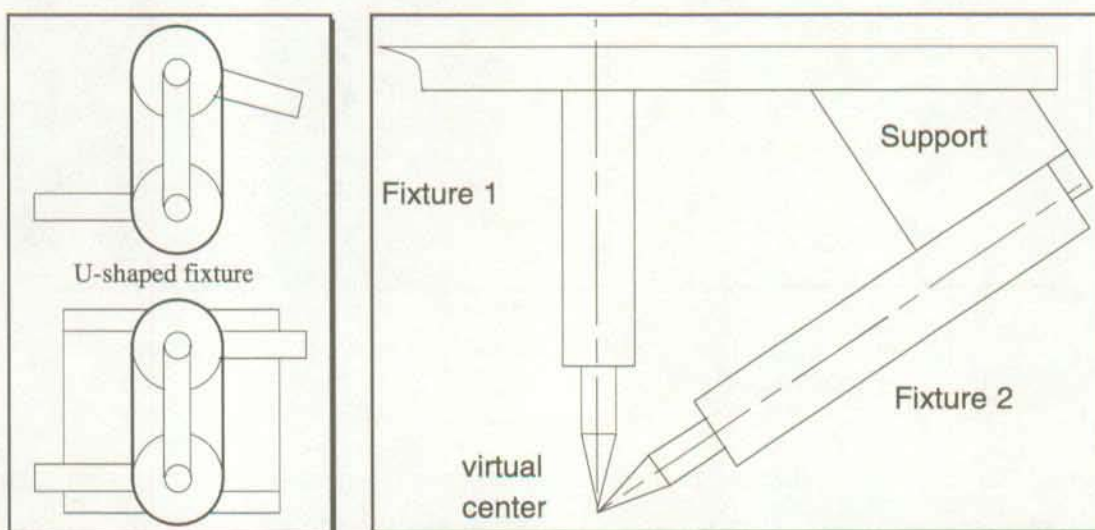


Fig. 2.9: Alignment of mechanical parts by means of precise fixtures.

Figure 2.10 shows the fully assembled Argos mock-up.

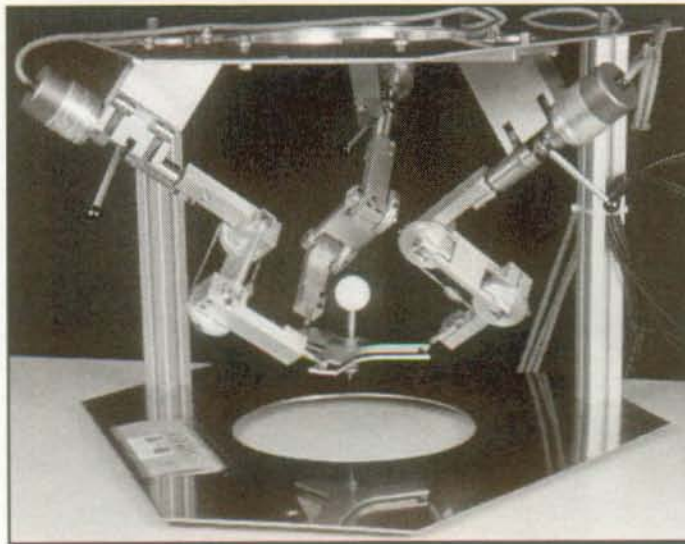


Fig. 2.10: The Argos mock-up

2.5 Conclusion

In this chapter concepts to built accurate robots were proposed and applied to two mock-ups of parallel robots. Interference of different joint-link trains, singularities and alignment of design elements in space cause severe problems. A powerful three-dimensional computer aided design program (3D-CAD) is therefore very helpful since the complexity of structural features often overloads the power of imagination. Different aspects of the construction of parallel robots are discussed in detail by Clavel [91].

The chosen cylinder/V-groove concept offers an advantage for testing different identification procedures. Small mechanical deviations can easily be realized by clamping a sheet of steel under the cylinder. If the thickness is known, it can be tested whether the identification procedure is capable to find this known deviation.

To illustrate the efficiency of the proposed concepts the deviations between the measured and the predicted end-effector's pose will be given (for more details see figure 5.9). Scanned all over the workspace is the Euclidean norm of the end-effector's error vector for both the Delta and the Argos mock-up. Figure 2.10 shows two histograms of the orientation error and the position error, respectively.

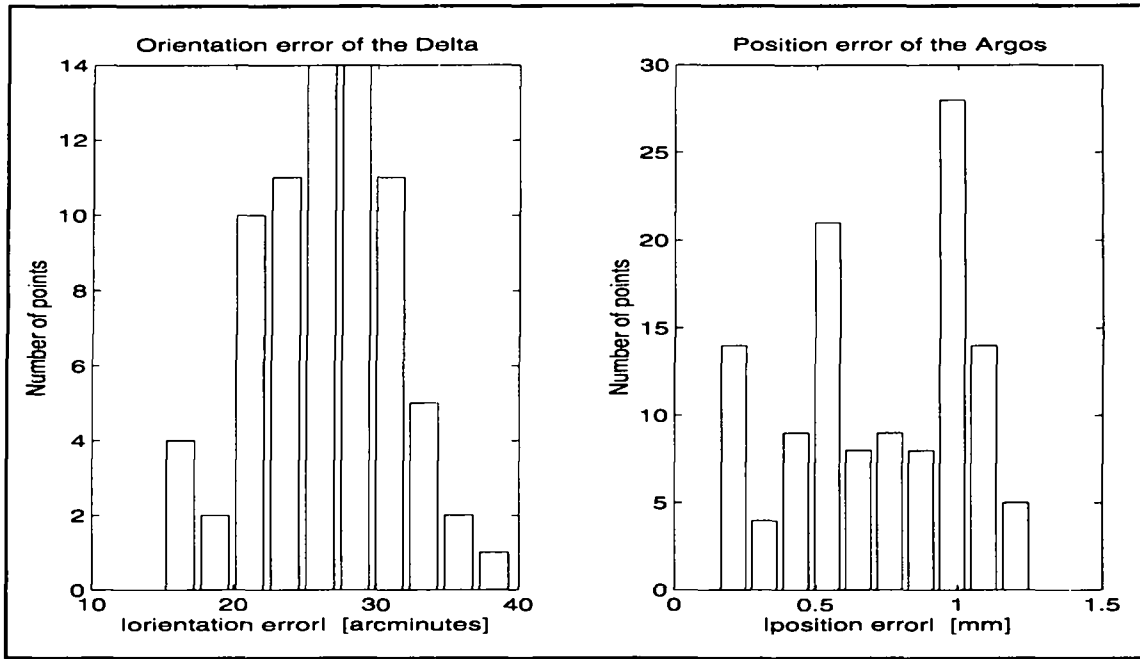


Fig. 2.11: Histograms showing the orientation error of the Delta and the position error of the Argos, respectively

Deviations in the position of the virtual rotation center of the Argos structure may further be improved by better alignment of mechanical parts whereas the deviations of the orientation of the end-effector of the Delta are small with a mean value of 27 arcminutes and a standard deviation of 5 arcminutes.

3. Modeling

3.1 Introduction

The aim of the modeling phase is to create one or a set of appropriate representation functions. This set of functions will later be used to store the geometrical relations of a robot. It can include either parameters of the underlying system or not [Hollerbach 89], and will be referred to as *parametric* and *nonparametric modeling*, respectively.

Nonparametric modeling is carried out by taking *mathematical functions* having *no physical interpretation*. Examples are polynomial fitting [Mooring 91] and neural networks [Kung 89, LeeS 91, Flury 94]. The modeling effort is much reduced since no inside knowledge of the underlying system is necessary. In other words: the system is treated as a "black box". The mathematical functions will be selected in such a way that they are convenient and are easy to treat during the identification step as well as in the implementation step, as are for example polynomials. Nevertheless the disadvantage of nonparametric modeling is on the measurement side, which is probably the most "expensive" step of a calibration procedure. Measurement points have to be considered in the entire volume where the calibration should be valid. The number of measurement points to be taken into account increases with the increasing non-linearity of the underlying system.

Parametric modeling is based on *physical relations* of the system under investigation. An example is the Navier-Stokes equation as physical model for any kind of fluid flow. The advantage over non-parametric modeling is already given by its definition, incorporating more inside knowledge about the underlying system. It is the more laborious way to establish a model, but its difficulty decreases if restricted only to the geometrical deviations of rigid bodies. The parametric model is global and covers the entire workspace. As will be seen later in chapter 4, parametric modeling allows some interesting propositions for set-ups in the measurement phase. Examples are the low-cost "*plane/spheres*" and "*short-cut*" set-ups proposed for the Delta robot.

Another important fact to keep in mind is that both of the chosen examples have only three of the six degrees of freedom of the Euclidean space. The other three degrees of freedom are blocked by the structure itself. Taking the Delta robot for instance, the

orientation can not be influenced by the actuators, but will still slightly vary over the working volume due to the tolerances of the mechanical parts. For that kind of robot parametric modeling offers an enormous advantage with its ability to analyze the source of the orientation error. In a *quality control* step the mechanical part out of tolerance can be detected by an identification procedure and may be exchanged with parts lying within the tolerances in order to guarantee that the orientation changes stay within pre-specified boundaries.

An additional advantage of parametric models is the fact that they can be used to simulate the deviations at the end-effector caused by estimated tolerances of the mechanical parts of the robot.

In order to learn as much as possible about the mathematics describing the geometry of parallel robots, this work is restricted to the study of parametric modeling.

3.2 Properties of a good model

Everett [87] proposes *three properties* for a good parametric model for calibration. First, the model must contain a sufficient number of parameters to describe the kinematics of the robot under investigation without being redundant. This is referred to as *completeness*. The second property required for a model is the reflection of small changes in the geometry by small variations of its parameters. This property is called *proportionality*. The third property is the ability to translate one valid model into any other accepted model, which is called *equivalence*. Any two complete models are therefore necessarily equivalent. It follows that equivalence prevents one complete model from producing greater accuracy than another one [Mooring 91].

The claim for equivalence imposes that there are still several ways to do the parameterization of a good model. This freedom will be used to create models which are easy to treat in the implementation phase (chapter 6). In this section the first two properties are discussed in details and an example on how to establish a model for a SCARA robot is given at the end of the section.

3.2.1 Completeness

For serial robots (open-loop) a simple equation to determine the minimal number of geometrical parameters for a complete model has been proposed [Everett 88b]:

$$C = 4R + 2P + 6 \quad (3.1)$$

where R is the number of revolute joints, P is the number of prismatic joints, and C is the total number of independent parameters for a complete model. Everett [88c] proved this equation mathematically.

For single-loop parallel robots some examples as well as an equation for the minimal number of geometrical parameters for a complete model are given in Lin [89]. In addition to open-loop structures these single-loop structures contain unsensed spherical joints (S) as well as revolute (R) and prismatic joints (P) which can be either sensed or unsensed¹. SS counts the number of pairs of S-joints. The number of encoders (E) or more general measurement devices is counted by E .

$$C = 3R + P + SS + E + 6 \quad (3.2)$$

For multi-loop parallel robots no such equation has been found in the literature. By extending eq. 3.2 the following equation for multi-loop structures is proposed, where the number of loops corresponds to L and the number of arbitrarily located frames to F :

$$C = 3R + P + SS + E + 6L + 6(F - 1) \quad (3.3)$$

This equation was empirically tested on several examples and seems to be valid under the following assumptions:

- The structure contains only lower joints³ of one degree of freedom [Denavit 55], which are either sensed or unsensed such as revolute joints (R) as well as prismatic joints (P).
- The only allowed lower joint with several degrees of freedom is an unsensed spherical joint (S). This is due to the fact that spherical joints can be manufactured very precisely

¹*Sensed and unsensed* indicates if the joint is equipped with a sensor or not.

²This corresponds to the formula proposed by Lin [89], who mentioned also that there are another 6 parameters to be added for a transformation on the end-effector link in order to get to an arbitrarily located tool-frame. Therefore eq. 3.2 can be written as: $C = 3R + P + SS + E + 6 + 6$

³For the definition of lower joints (∇ higher joints) see Mooring [91]. Lower joints have a single and unique axis of motion, whereas higher joints take into account deviations in their movements. It is a common assumption that real joints of robots can be modeled in a first approximation as lower joints.

as discussed in chapter 2. Therefore S-joints can be assumed to be perfect with all three axes intersecting in a common point.

- All remaining lower joints of several degrees of freedom have to be replaced by R- and P-joints [Hiller 88]. This corresponds to the statement that with the exception of an S-joint all lower joints with several degrees of freedom cannot be assumed to be perfect. Therefore an universal joint has to be replaced by two R joints which are neither exactly perpendicular to each other nor do the axes intersect at a common point.
- F counts frames which are fully arbitrarily chosen. A fully arbitrary choice means that there is no restriction on any of the 6 parameters needed for a general transformation to get from one frame to another one in space. Fully arbitrarily chosen frames are indicated by uppercase letters i.e. $\{B\}$ or $\{P\}$ for the base and the moving frame, respectively. On the other hand lowercase letters will be used to indicate fully restricted frames such as are $\{b\}$ and $\{p\}$.
- A SS-pair is composed of two S-joints connected by a simple rod without any intermediate joint or frame attached to it. Excluded from this category are joints composed of a branching between rigid links with three or more S-joints as well as a link equipped with two S-joints which is chosen as an end-effector with an arbitrarily moving frame attached to it.

For practical considerations in eq. 3.3 the possibility is given to choose the base and the moving frame in an arbitrary location ($F=2$), which requires an even higher number of independent parameters. According to the Denavit-Hartenberg (DH)-parameterization¹ [Denavit 55] a base frame $\{b\}$ that is not arbitrarily located would lie with its z-axis in the direction of the first joint axis and the x-axis in direction of the common normal to the second joint axis (figure 3.3). It is nearly impossible to locate an object, especially the end-effector measurement device in the working volume of the robot relative to this base frame $\{b\}$. This is due to the fact that the joint axis and in particular the common normal are virtual lines in space which in other words are not represented physically on the robot itself. Claiming the presence of a physically existing base frame as for examples hole / V-grove / plane arrangement corresponds to the claim for an arbitrarily located base frame $\{B\}$. If a calibration model is only used for simulation, no arbitrarily located base and moving frame is necessary.

¹For definition of the DH-parameters see paragraph 3.2.2.

Application of eq. 3.3 to some examples with $F=2$ leads to the following results:

robots	reference	joint-link-trains	R	P	SS	E	L	C
SCARA	-	2RP	2	1	0	3	0	16
Puma	-	6R	6	0	0	6	0	30
Kite ¹	[Vischer 95]	3R + 2R	5	0	0	2	1	29
PantoScope ¹	"	R(2R/2S)2R + 4R	9	0	1	2	2	48
Argos	"	3[R(2R/2S)S]	3*3	0	3*1	3	5	69
Argos	"	3[R(2R/5R)3R]	3*11	0	0	3	5	138
Agile Eye	[Gosselin 94]	3[3R]	3*3	0	0	3	2	48
Agile Eye	"	3[2RS]	3*2	0	0	3	2	39
Agile Eye	"	3[5R]	3*5	0	0	3	2	66
Delta-Cardan	[Clavel 91]	3[5R]	3*5	0	0	3	2	66
Linear-Delta	"	3[P(2S/2S)]	0	3*1	3*2	3	5	48
Linear-Delta	"	3[P(5R/5R)]	3*10	3*1	0	3	5	132
Delta	"	3[R(2S/2S)]	3*1	0	3*2	3	5	54
Delta	"	3[R(5R/5R)]	3*11	0	0	3	5	138
Stewart ²	[Stewart 65]	6[2S]	0	0	6*1	0	5	42
Stewart	"	6[SPS]	0	6*1	0	6	5	48
Stewart	"	6[2RP3R]	6*5	6*1	0	6	5	138

Table. 3.1: Minimal number of design parameters for a complete calibration model for some examples.

For the Argos structure a simplified model will be introduced assuming that the whole structure is perfectly spherical. The question arising then is how to calculate the minimal number of parameters (C) for a structure which is either perfectly plane or spherical. On spheres as well as on planes the degrees of freedom are reduced from six to three. The following equation is proposed for perfectly planar or spherical mechanisms:

$$C = R + S + E + 3L + 3(F - 1) \quad (3.4)$$

It is interesting to note that spherical joints (S) do behave exactly like rotative joints (R) in a plane and on a sphere. If they are unsensed, interchanging the joints is possible

¹See Annex A.1

²The prismatic actuators (P) are assumed to be perfectly assembly. For a more detailed discussion see paragraph 3.3.1.

without influencing the model. Prismatic joints (P) do not count. As an example the plane complementary to the Stewart Platform is taken [Merlet 90, Gosselin 88, Gosselin 91]:

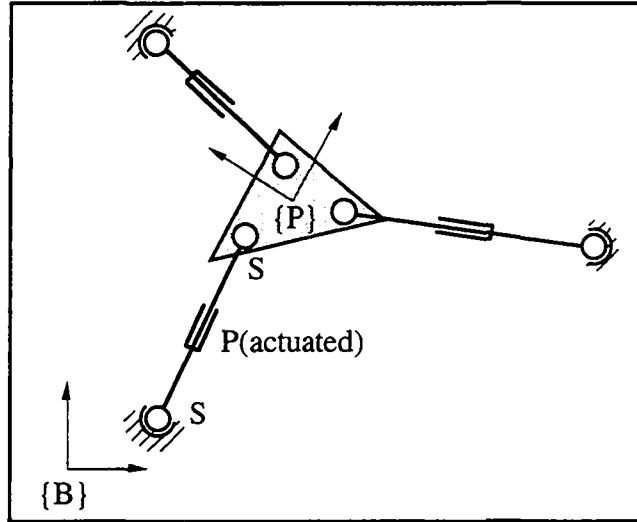


Fig. 3.2: Planar "Stewart"-Platform

Application of eq. 3.4 leads to:

$$C = 0 + 6 + 3 + 3 * 2 + 3(2 - 1) = 18$$

Two coordinates are required to locate a S-joint in the plane. Twelve parameters are therefore needed to describe the position of the six S-joints in the {B}-frame and in the {P}-frame, respectively. To these twelve parameters another three angles have to be added, indicating the misalignment of the axes of the prismatic actuators to the connecting line of the two S-joints. Finally another three parameters for the three encoder offsets have to be added to end up with the proposed eighteen parameters. Replacement of the S-joints by R-joints will not affect the model, but will cut off the three isolated degrees of freedom enabling each joint-link train (SPS) to rotate about its longitudinal axis.

Another example is the spherical parallel robot of Gosselin [88] with its 3[3R] structure. According to eq. 3.4 twenty-one parameters are necessary for a complete model assuming that it is perfectly spherical. For the same structure forty-eight parameters were found in space (table 3.1). The assumption that a mechanism is perfectly plane or spherical leads to a considerable reduction of the parameters required for a complete model.

3.2.2 Proportionality

It has been shown so far how many parameters have to be taken into account for a complete calibration model. In this paragraph the question "what of kind of parameters can be used for a calibration model" will be addressed.

In the identification phase (chapter 5) the chosen parameters will be slightly varied in order to fit the calibration model to the measurement data. This leads to an important claim for the parameterization of calibration models. Small variations in the robot topology should be represented by small variations in the chosen parameterization. This property is referred to as proportionality. In nominal modeling one does not generally intend to vary the design parameters slightly. Thus, propositions made for the parameterization of nominal models of robots cannot be used for calibration models without further consideration. In particular it has to be checked whether each proposed parameterization fulfills the criteria of proportionality and completeness or not.

These two properties will be illustrated using the DH(Denavit&Hartenberg)-parameterization proposed by Denavit [55] for nominal modeling, which is widely used in robotics. As mentioned by Hayati [83], the DH-parameterization fails to be proportional for nearly parallel axes. Everett [87] pointed out that the DH-parameterization is complete for R-joints, but becomes redundant for P-joints by introducing four instead of two parameters (see eq. 3.1).

In literature three main groups of propositions for systematic parameterization of robots can be found:

1. for nominal models of serial structures [Denavit 55]
2. for calibration models of serial structures [Hayati 83, Stone 86, Chen 86, Broderick 88, Zhuang 90]. Hollerbach [91] has given a good overview about these parameterizations.
3. for nominal models of parallel structures [Sheth 71, Meghed 83, Khalil 86]

To the author's knowledge no systematic approach exists for the parameterization of calibration models of parallel robots. Therefore, propositions made for serial structures will be adapted. The parameterizations mentioned under point 2 generally do not respect the claim for being non-redundant because they contain four to nine parameters independent of the joint type (R,P). Redundancy in the parameters causes severe problems in the identification phase because the identification Jacobian becomes singular (chapter 5). Schröer [93] investigated the claim for proportionality. He concluded that a non-redundant parameterization that always stays proportional doesn't exist. Thus, he proposed to choose the parameterization as a function of the topology of

the robot. This is an extension of a concept already proposed by Hayati [83] using for nearly perpendicular joint axes the DH-parameterization. For nearly parallel joint axes a new parameterization is referred to as the H(Hayati)-parameterization. These two cases represent the two main topologies of joint axes in robotics:

Nearly perpendicular axes	->	DH (Denavit-Hartenberg)-parameterization
Nearly parallel axes	->	H (Hayati)-parameterization

For the DH-parameterization slightly varying definitions can be found [Denavit 55, Paul 81, Craig 89]. In this work the definition given by Paul is being used. Figure 3.3 represents the non-redundant case of two consecutive revolute joints.

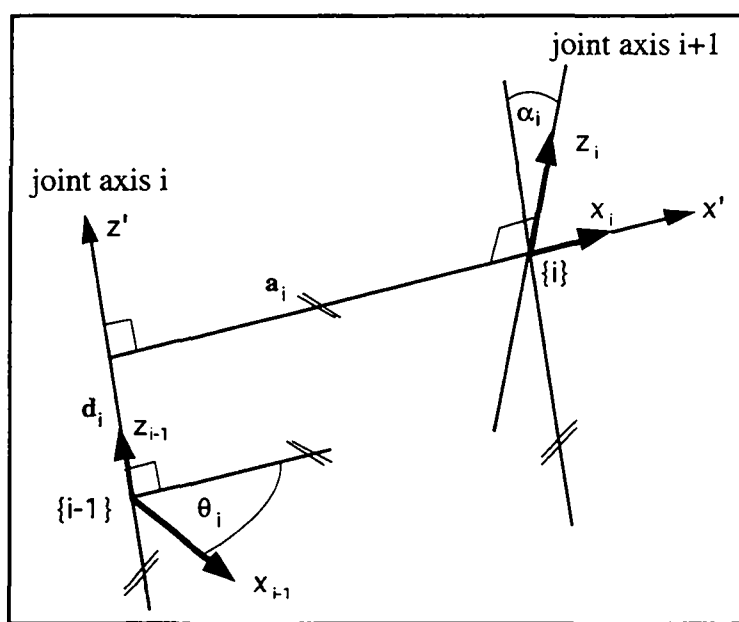


Fig. 3.3: The DH(Denavit&Hartenberg)-parameterization for revolute joints

The frame $\{i-1\}$ is attached to the joint axis i whereas the frame $\{i\}$ is fixed on the joint axis $i+1$. The origin of the new frame $\{i\}$ is defined by the common normal between the two joint axes. The transformation matrix between the two frames is given by:

$${}^{i-1}\bar{T}_i = Rot4[z_{i-1}, \theta_i] \cdot Trans4[z_{i-1}, d_i] \cdot Trans4[x', a_i] \cdot Rot4[x', \alpha_i]^1 \quad (3.5a)$$

The transformations are performed as follows:

Turn axis x_{i-1} about axis z_{i-1} by the joint angle θ_i until the x_{i-1} axis becomes parallel to the common normal between the two joint axes. Move by the link offset d_i along the

¹For the matrix functions Rot4 and Trans4 see annex B and Craig [89].

joint axis i to the lower end of the common normal. Carry on moving along the common normal by the link length a_i to reach the origin of the frame $\{i\}$. Turn the parallel line to the joint axis i by a twist angle α_i until the final position of the frame $\{i\}$ on the joint axis $i+1$ is reached.

The DH-parameterization fails to be proportional for nearly parallel axes. For very small changes of the relative position of the two axes with respect to each other results a non-proportional change of the offset d_i [Hayati 83, Kim 87, Mooring 91].

The H-parameterization (fig. 3.4) introduced in Hayati [83] stays proportional for nearly parallel axes. The origin of the frame $\{i\}$ is no more defined by the common normal, but by the intersection of the xy -plane of the frame $\{i-1\}$ and the joint axis $i+1$:

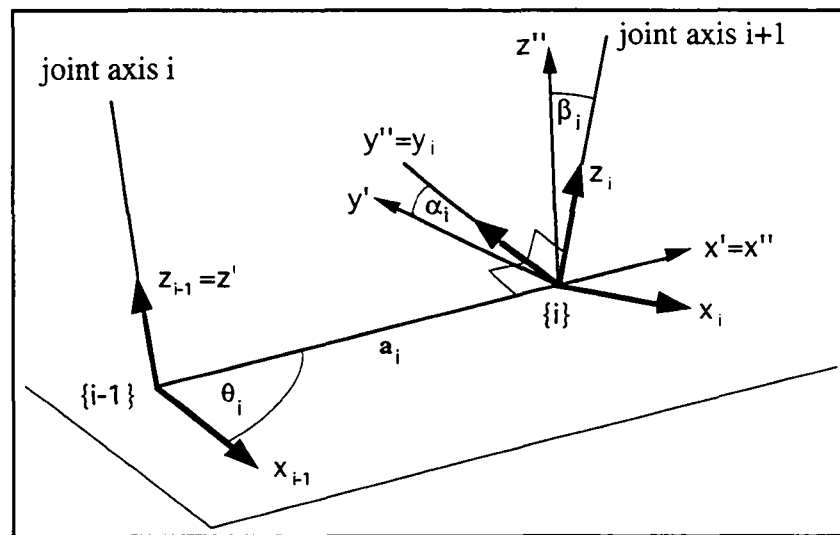


Fig. 3.4: The H(Hayati)-parameterization for revolute joints for nearly parallel axes

The H-parameterization is given by:

$${}^{i-1}\bar{T}_i = Rot4[z_{i-1}, \theta_i] \cdot Trans4[x', a_i] \cdot Rot4[x', \alpha_i] \cdot Rot4[y', \beta_i] \quad (3.5b)$$

The transformations are performed as follows:

Turn axis x_{i-1} about the axis z_{i-1} by the joint angle θ_i until the connecting line between the two frame origins is reached. Move along the new axis x' by the link length¹ a_i to reach the origin of the frame $\{i\}$. Turn axis y' about the axis x' by a first twist angle α_i until reaching the common normal between the connecting line and the

¹Note that the link length defined here is not the same as in the DH-representation.

joint axis $i+1$. This common normal defines the axis y_i of the new frame $\{i\}$. It remains to turn the axis z'' about the axis y_i by a second twist angle β_i to reach the final position of the frame $\{i\}$ on the joint axis $i+1$.

The H-parameterization becomes singular if the two axes of the joints become orthogonal to each other because the origin of the frame $\{i\}$ becomes undefined.

To summarize this paragraph a single transformation¹ containing the DH- as well as the H-parameterization is introduced:

$${}^{i-1}\bar{\mathbf{T}} = \text{Rot4}[z_{i-1}, \theta_i] \cdot \text{Trans4}[z_{i-1}, d_i] \cdot \text{Trans4}[x', a_i] \cdot \text{Rot4}[x', \alpha_i] \cdot \text{Rot4}[y'', \beta_i] \quad (3.6)$$

According to Schröder [93] the parameterization for a complete and proportional model can be rewritten as shown in table 3.5. The crosses indicate which parameters have to be taken, $\{C\}$ stands for the current frame, J_R for the joint axis of a revolute joint and J_P for the joint axis of a prismatic joint.

	joint angle θ	joint offset d	link length a	twist angle 1 α	twist angle 2 β	typ of parameters
$\{C\} \perp J_R$	x	x	x	x	-	DH
$\{C\} \parallel J_R$	x	-	x	x	x	H
$\{C\} \perp J_P$	x	-	-	x	-	DH
$\{C\} \parallel J_P$	-	-	-	x	x	H

Table 3.5: Topology dependent choice of parameters for a complete and proportional model using the representation given in eq. 3.6

The concept of parameterization summarized in table 3.5 works for a lot of robot topologies with the exception of sensed, consecutive, prismatic joints for example, where problems arise to model the encoder offset [Schröder 93]. This case is very rare in parallel robot topologies and therefore will not be treated here.

For a calibration model it is important to ask for what kind of topologies singularities arise in a chosen parameterization? As has been shown in this paragraph, the difference in the singularities between the DH- and the H-parameterization is only generated by the difference of the definition of the origin of two consecutive frames. The definition,

¹Notice that the definition of the joint angle, the link offset, the link length and the twist angles depends of the chosen parameterization and is therefore different for the DH-parameterization then for the H-parameterization.

that the z-axis is laying on the joint axis is only a standardized nomenclature and is therefore arbitrary.

If in this work is referred to one of the two parameterization, then only the definition of the origin of the consecutive frame is meant and no restriction is given on the nomenclature. For the Argos structure for instance a DH-parameterization is used having the x-axis on the joint axis.

3.2.3 Calibration model for a serial SCARA robot

As an example a serial SCARA robot (fig. 3.6) having three degrees of freedom will be parameterized. The main characteristics of SCARA robots are the possession of three nearly parallel joint axes. According to table 3.5 H-parameters will be used to get from the first to the last joint axis. A partial DH-transformations will lead to a intermediate frame {3} having its z-axis already coincident with the z-axis of the arbitrarily located {P}-frame. It remains to twist the x-axis of {3} about the z-axis until it becomes coincident with the x-axis of the {P}-frame¹.

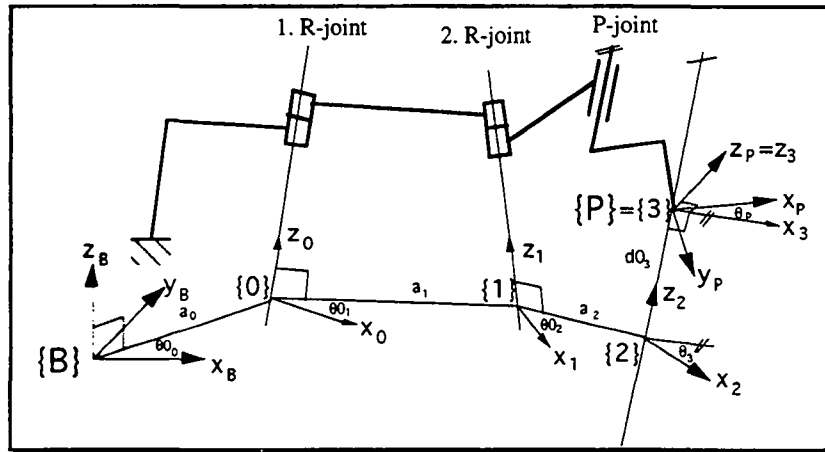


Fig. 3.6: Parameterization² of a SCARA-robot with a 2RP structure

¹Notice that the H-parameterization fails for coincident z-axes because the connection line between the two intersection points with the x-y plane gets undefined.

²The notation $\{P\}=\{3\}$ indicates that the two frames {P} and {3} have the same origin and the transformation between is therefore a pure rotation.

According to table 3.1 sixteen parameters are required for a complete robot calibration model which is referred to as model 16. Due to the serial topology of the SCARA robot, the calibration model 16 can be established by simple multiplication of homogeneous matrices. Together with table 3.7 the model 16 is given by:

$$\boxed{{}^B\overline{\overline{T}}_P = {}^B\overline{\overline{T}}_0 \cdot {}^0\overline{\overline{T}}_1 \cdot {}^1\overline{\overline{T}}_2 \cdot {}^2\overline{\overline{T}}_3 \cdot {}^3\overline{\overline{T}}_P} \quad (3.7)$$

	joint angle	joint offset	link length	twist angle 1	twist angle 2	parameter
${}^B\overline{\overline{T}}_0$	θ_0	-	a_0	α_0	β_0	H
${}^0\overline{\overline{T}}_1$	θ_1	-	a_1	α_1	β_1	H
${}^1\overline{\overline{T}}_2$	θ_2	-	a_2	α_2	β_2	H
${}^2\overline{\overline{T}}_3$	θ_3	d_0_3	-	α_3	-	DH
${}^3\overline{\overline{T}}_P$	θ_P	-	-	-	-	undef.

Table 3.7: Parameters to be substituted into eq. 3.6 for the complete and proportional calibration model 16

3.3 Models for the Stewart Platform

3.3.1 Introduction

According to table 3.1, 138 parameters are required for a complete model of the general Stewart platform taking into account all possible geometrical deviations. Wang [92]¹ used such a model to simulate the final end-effector accuracy resulting from design tolerances. He concluded that the resulting accuracy is about the same as for a comparable serial structure and that the effect of deviations in passive joints on the end-effector accuracy is negligibly small compared to the other deviations. With perfect spherical joints the number of independent parameters drops from 138 to 48 parameters (Table 3.1). Wang [92], Innocenti [95] and Zhuang [95] used a model with 42 parameters for calibration assuming that with exception of the encoder offsets the

¹In Wang [92] only 132 independent parameters including the encoder offset were found. The missing parameter per joint-link train is caused by the assumption that the last virtual R-joint of the upper S-joint has to lie in the z-y-plane of the end-effector frame {P}. Dropping this constraint leads to the proposed 138 parameters.

prismatic actuators (P) are perfectly assembled. In order to apply eq. 3.3, the SPS joint-link train has to be replaced by an SS pair (Table 3.1). The model 42 is introduced in the following section.

3.3.2 Model 42

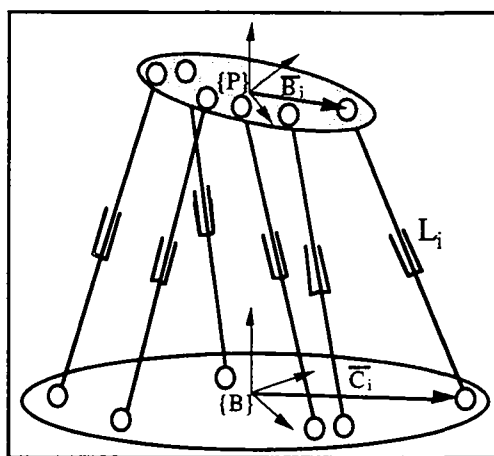


Fig. 3.8: Parameterization for the model 42 of the Stewart - Platform

The chosen parameterization is shown in figure 3.8. Note that the model 42 can be established with the definition of only two frames {B} and {P}. The six closure equations representing the model 42 are given by:

$$\begin{aligned} & \overline{\mathbf{CB}}_i^T \cdot \overline{\mathbf{CB}}_i = L_i^2 \\ \rightarrow & \text{with} \\ & \overline{\mathbf{CB}}_i = \overline{\mathbf{P}} + \overline{\mathbf{R}} \cdot \overline{\mathbf{B}}_i - \overline{\mathbf{C}}_i \end{aligned} \quad i=1..6 \quad (3.8)$$

where $\overline{\mathbf{P}}$ is a vector pointing of the origin of the {B}-frame to the origin of the {P}-frame containing the three world coordinated x,y and z. $\overline{\mathbf{R}}$ is a rotation matrix describing the orientation of the {P}-frame with respect to the {B}-frame by means of another three world coordinates as for example three Euler angles.

Having three coordinates describing the location of the ball socket joint (S) on the base plate and another three for the location of the S-joint on the end-effector leads to six parameters for each joint link train. Adding one parameter more for the encoder offset¹ results in the proposed 42 parameters.

¹ Replacing in eq. 3.8 L_i by $L_i + LO_i$, where L_i is the transducer reading and LO_i the offset.

3.3.3 Conclusion

The system of equations 3.8 apparently seems to be quite simple. Nevertheless, due to its complexity the direct problem remained unsolved for many years. In a remarkable work Husty [94] was able to find a univariate polynomial of 40th degree. It still remains some open questions that await resolution such as for example the analytical expression of the factors of the 40th order polynomial or the question of whether there is a configuration where all of the 40 solutions are real. By numerical search never more than 16 real solutions could be found.

3.4 Models for the Delta robot

3.4.1 Introduction

In this section two different calibration models for the Delta robot will be developed. A model considering all possible geometrical deviations would have 138 parameters. To apply eq. 3.3 a S-pair has to be modeled as a 5R joint-link train in order to get rid of the isolated degree of freedom¹. However, the increase of accuracy gained by considering errors in passive joints is very small compared to the tremendous increase of the mathematical complexity [Wang 92]. This was one of the reasons to search for concepts allowing to produce S-joints as perfect as possible (chapter 2) in order to model them without errors. In this case the number of independent parameters drops according to table 3.1 from 138 to 54. This model will be referred to as model 54. Assuming further that the spatial parallelogram remains perfect, model 54 can be reduced to a model with 24 parameters. This model 24 will finally be converted to the nominal model [Clavel 91].

The following remarks about models 54 and 24 are important:

- The parameterization for the two calibration models should be the same. In other words 24 parameters have to be identically chosen for both models. This prevents from transforming a set of parameters into another, and makes the different models easier to compare.
- Rotation and translation are treated separately as in ordinary vector analysis, which is the traditional way to establish models for the Delta robot [Sternheim 87, Pierrot 90, Clavel 91, Codourey 91, Devaquet 92 and Guglielmetti 94].
- As shown in figure 3.9 the end-effector will be attached to the base-plate for the parameterization. This will allow the reduction of model 54 to model 24 with the end-effector shrinking to a single point (fig. 3.11).
- In order to maintain proportionality (paragraph 3.2.2) special care is taken about three nearly parallel lines. These three lines are the axis of the motor, the line connecting the proximal S-joints and the line joining the distal S-joints. For model 24 these three lines have to be exactly parallel in order to maintain the spatial parallelogram.

¹ For "isolated degree of freedom" see paragraph 1.6.2 (Stewart Platform).

- The thirty joint angles of the passive S-joints (excluding the six isolated degrees of freedom) are variables depending on the robots configuration. The closure equations are established in such a way that these depending variables don't appear. This corresponds to the simplification introduced by Clavel [91] compared to the model of Sternheim [87].
- Instead of working with six closure equations for each forearm subchain, the sum and the difference of the two equations of a pair of forearms will be used to represent one main chain.

3.4.2 Parameterization

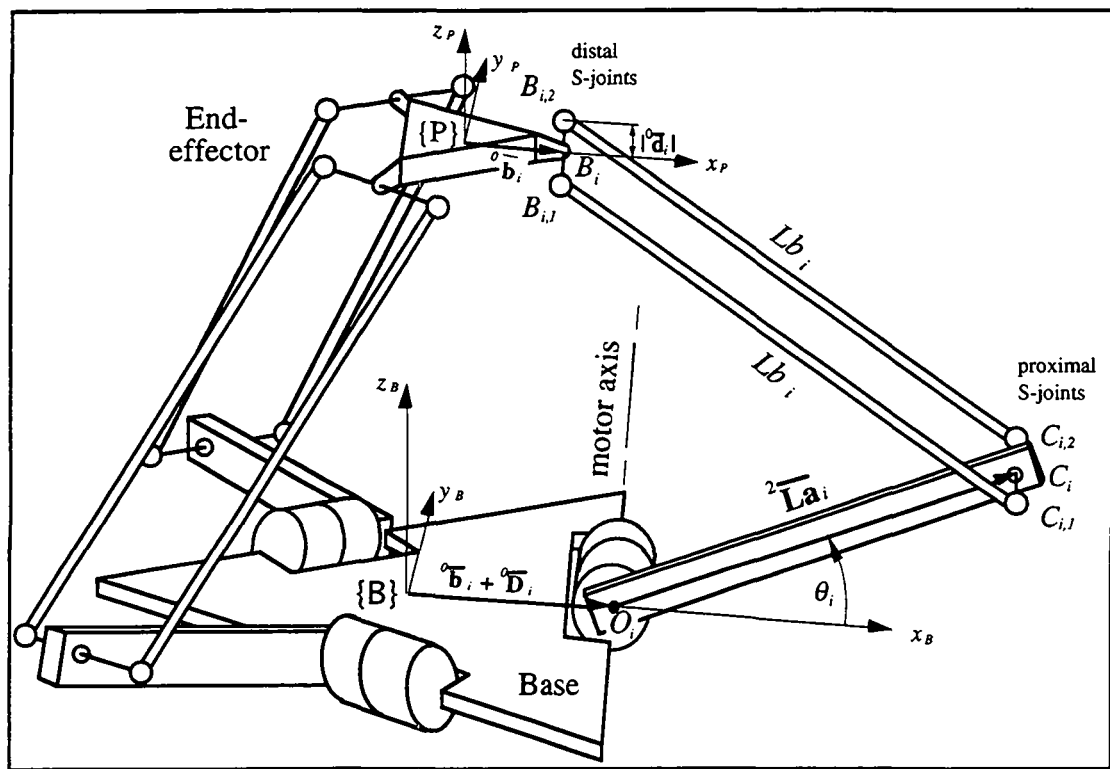


Fig. 3.9a: Parameterization of a Delta robot *without* geometric deviations.

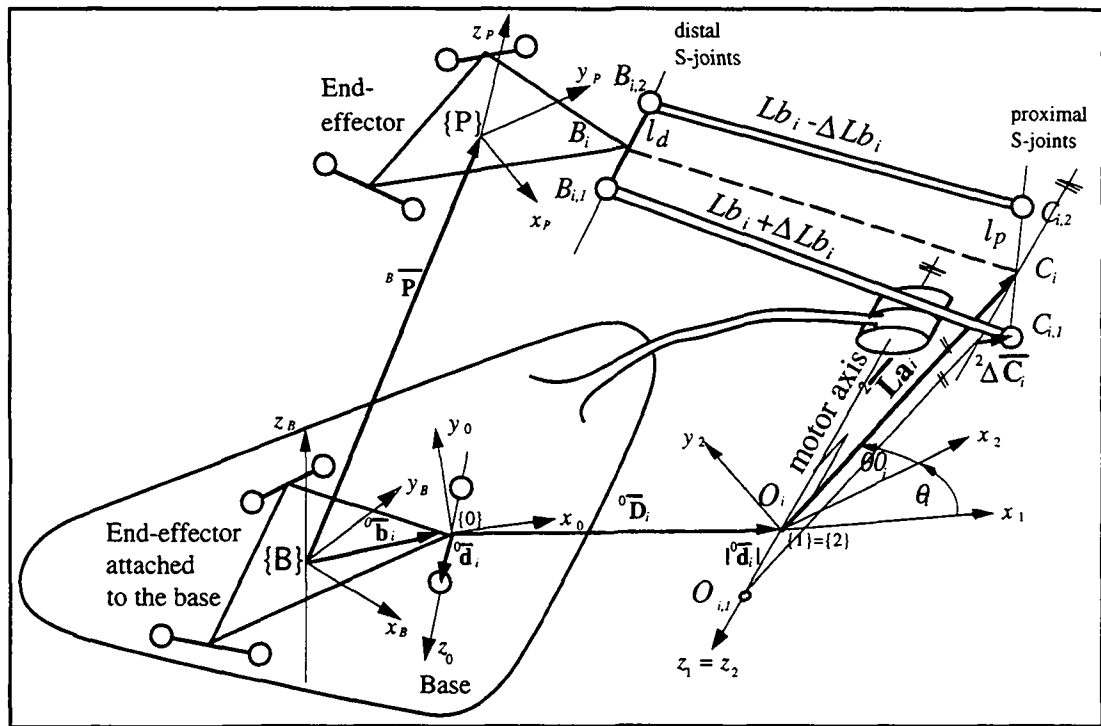


Fig. 3.9b: Parameterization of one joint-link train of the Delta robot *with* geometric deviations. The end-effector is thought to be attached to the base. For convenience the frames {0}, {1} and {2} are represented in a translated position. Actually, they are related by pure rotations to the {B}-frame.

According to figure 3.9a and 3.9b the following points, lines and frames are defined:

- $B_{i,1,2}$: Center points of the S-joints attached to the end-effector \rightarrow distal S-joints
- $C_{i,1,2}$: Center points of the S-joints attached to the arms \rightarrow proximal S-joints
- ℓ_d, ℓ_p : Connecting strait section from $B_{i,1}$ to $B_{i,2}$ and $C_{i,1}$ to $C_{i,2}$, respectively
- B_i, C_i : Mid-point of the section ℓ_d and ℓ_p , respectively
- O_i : Projection point of C_i on the motor axis
- $O_{i,1,2}$: Points on the motor axis located in distance $\pm \ell_d/2$ of O_i
- {B}: The base frame {B} is arbitrarily fixed to the base
- {P}: The moving frame {P} is arbitrarily fixed to the end-effector
- {0}: The z-axis of the distal S-joint frame {0} is parallel to ℓ_p
- {1}: The z-axis of the motor frame {1} is parallel to the motor axis
- {2}: The twisted motor frame {2} is the frame {1} twisted by the motor angle

Position vector and rotation matrix containing the six world coordinates	
${}^B\overline{\overline{\mathbf{R}}} = Rot3(z, \gamma) \cdot Rot3(y, \beta) \cdot Rot3(x, \alpha)$	Rotation matrix describing the orientation of the {P}-frame relative to the {B}-frame by means of fixed angles
${}^B\overline{\mathbf{P}} = \{x, y, z\}^T$	Vector describing the origin of the {P}-frame relative to the {B}-frame by means of three Cartesian coordinates
Rotation matrix containing the three joint coordinates	
${}^1_2\overline{\overline{\mathbf{Q}}} = Rot3(z, \theta_i)$	Rotation matrix containing the motor angle θ_i
Scalars, vectors and rotation matrices containing the 54 geometric parameters:	
${}^B_0\overline{\overline{\mathbf{T}}}_i = {}^P_0\overline{\overline{\mathbf{T}}}_i = Rot3(z, \vartheta_i) \cdot Rot3(x, \alpha_i)$	Rotation matrix describing the {0}-frame relative to the {B} and {P}-frame, respectively, which is done by means of DH-parameters for $\approx \perp$ axes
${}^0_1\overline{\overline{\mathbf{T}}}_i = Rot3(x, \Delta\alpha_i) \cdot Rot3(y, \Delta\beta_i)$	Rotation matrix describing the {1}-frame relative to the {0}-frame by means of H-parameters for $\approx \parallel$ axes
${}^0\overline{\mathbf{D}}_i = \{D_{xi}, D_{yi}, D_{zi}\}^T$	Vector pointing from the point B_i of the attached end-effector to the point O_i on the motor axis
${}^2\overline{\mathbf{L}}_i = \{La_{xi}, La_{yi}, 0\}^T$ with $La_{xi} = La_i \cos(\theta\theta_i)$ $La_{yi} = La_i \sin(\theta\theta_i)$	Vector pointing from the point O_i to the point C_i including the encoder offset $\theta\theta_i$ and the arm length La_i
${}^0\overline{\mathbf{b}}_i = \{b_{xi}, b_{yi}, b_{zi}\}^T$	Vector pointing from the origin of the frame {B} and {P}, respectively, to the point B_i
${}^0\overline{\mathbf{d}}_i = \{0, 0, d_{zi}\}^T$	Vector whose z-component is half as long as line ℓ_d
${}^2\overline{\overline{\mathbf{C}}}_i = \{\Delta C_{xi}, \Delta C_{yi}, \Delta C_{zi}\}^T$	Error vector defined as the difference of the vector $\overline{\mathbf{OC}}_{i,1} (= C_{i,1} - O_{i,1})$ and the vector $\overline{\mathbf{OC}}_i (= C_i - O_i)$.
Lb_i	Average length of the forearms
ΔLb_i	Half the difference of the forearms' lengths

Table 3.10: Parameterization of one of the three R(2S/2S) joint link train of the Delta robot

3.4.3 Model 54

According to Hiller [88] the forearms as a "characteristic pair of joints" of the type "distance of two points" will be removed. This leads to a set of six closure equations which are free of dependent passive joint variables. To close a joint link train it is claimed that the Euclidean norm of the vector between the proximal and the distal S-joint is equal to the length of the corresponding forearm.

The two closure equations for one main joint link train i can be written as:

$$(\bar{\mathbf{B}}_{i,1,2} - \bar{\mathbf{C}}_{i,1,2})^T \cdot (\bar{\mathbf{B}}_{i,1,2} - \bar{\mathbf{C}}_{i,1,2}) = (Lb_i \pm \Delta Lb_i)^2$$

with

$$\bar{\mathbf{B}}_{i,1,2} = {}^B\bar{\mathbf{P}} + {}^B\bar{\mathbf{R}} \cdot {}^P\bar{\mathbf{T}}_i \cdot ({}^0\bar{\mathbf{b}}_i \pm {}^0\bar{\mathbf{d}}_i) \quad i=1..3 \quad (3.9)$$

$$\bar{\mathbf{C}}_{i,1,2} = {}^B\bar{\mathbf{T}}_i \cdot ({}^0\bar{\mathbf{b}}_i + {}^0\bar{\mathbf{D}}_i + {}^0\Delta\bar{\mathbf{T}}_i \cdot (\pm {}^1\bar{\mathbf{d}}_i + {}^1\bar{\mathbf{Q}}_i \cdot ({}^2\bar{\mathbf{L}}a_i \pm {}^2\Delta\bar{\mathbf{C}}_i)))$$

For simplicity the leading sub- and superscript will be dropped and eq. 3.9 rewritten as:

$$(\bar{\mathbf{C}}\bar{\mathbf{B}}_i + \Delta\bar{\mathbf{d}}_i)^T \cdot (\bar{\mathbf{C}}\bar{\mathbf{B}}_i + \Delta\bar{\mathbf{d}}_i) = (Lb_i + \Delta Lb_i)^2 \quad i=1..3 \quad (3.10)$$

$$(\bar{\mathbf{C}}\bar{\mathbf{B}}_i - \Delta\bar{\mathbf{d}}_i)^T \cdot (\bar{\mathbf{C}}\bar{\mathbf{B}}_i - \Delta\bar{\mathbf{d}}_i) = (Lb_i - \Delta Lb_i)^2 \quad i=1..3 \quad (3.11)$$

with

$$\bar{\mathbf{C}}\bar{\mathbf{B}}_i = \frac{(\bar{\mathbf{B}}_{i1} + \bar{\mathbf{B}}_{i2}) - (\bar{\mathbf{C}}_{i1} + \bar{\mathbf{C}}_{i2})}{2} \quad \text{and} \quad \Delta\bar{\mathbf{d}}_i = \frac{(\bar{\mathbf{B}}_{i1} - \bar{\mathbf{B}}_{i2}) - (\bar{\mathbf{C}}_{i1} - \bar{\mathbf{C}}_{i2})}{2}$$

By addition and subtraction of eqs. 3.10 and 3.11 to and from each other, model 54 is created, which is well adapted to the Delta robot:

G1: $\bar{\mathbf{C}}\bar{\mathbf{B}}_i^T \cdot \bar{\mathbf{C}}\bar{\mathbf{B}}_i + \Delta\bar{\mathbf{d}}_i^T \cdot \Delta\bar{\mathbf{d}}_i = Lb_i^2 + \Delta Lb_i^2$

G2: $\bar{\mathbf{C}}\bar{\mathbf{B}}_i^T \cdot \Delta\bar{\mathbf{d}}_i = Lb_i * \Delta Lb_i$

with

$\bar{\mathbf{C}}\bar{\mathbf{B}}_i = \bar{\mathbf{P}} + \bar{\mathbf{R}} \cdot \bar{\mathbf{T}}_i \cdot \bar{\mathbf{b}}_i - \bar{\mathbf{T}}_i \cdot (\bar{\mathbf{b}}_i + \bar{\mathbf{D}}_i + \Delta\bar{\mathbf{T}}_i \cdot \bar{\mathbf{Q}}_i \cdot \bar{\mathbf{L}}a_i)$

$\Delta\bar{\mathbf{d}}_i = \bar{\mathbf{R}} \cdot \bar{\mathbf{T}}_i \cdot \bar{\mathbf{d}}_i - \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{T}}_i \cdot (\bar{\mathbf{d}}_i + \bar{\mathbf{Q}}_i \cdot \Delta\bar{\mathbf{C}}_i)$

i=1..3 (3.12)

Adding and subtracting the closure equations of a forearm pair leads to splitting of the less adapted set of closure equations given in eq. 3.9 into two more characteristic sets of closure equations G1 and G2. Two similar sets of closure equations will be found for the Argos structure (paragraph 3.5.3).

3.4.4 Model 24

To establish model 24, model 54 is simplified by assuming that the end-effector remains perfectly parallel to the base frame. In other words: The spatial parallelogram remains perfect. This is the same assumption on which the nominal models of Sternheim [87] and Clavel [91] are based. It can therefore be said that model 24 is an extended nominal model. Such a model for the Delta robot containing 18 parameters was introduced by Zobel [93]. For completeness (paragraph 3.2.1) three encoder offsets (table 3.10: θ_{0i}) and the three tilt angles of the motors (table 3.10: α_i) have to be added. The encoder offsets have a major influence on the resulting accuracy, as can be seen in the identification phase in chapter 5. The tilt angles become particularly important for the linear Delta [Stevens 94], where two of these three angles are inclined. Therefore a complete model has 24 parameters. Any extended nominal model doesn't allow errors in the spatial parallelogram, thus the end-effector is always perfectly parallel to the base plate:

$$\overline{\overline{\mathbf{R}}} = \overline{\overline{\mathbf{I}}} \quad (3.13)$$

where $\overline{\overline{\mathbf{I}}}$ is the 3x3 identity matrix.

The number of world coordinates is reduced from six to the three Cartesian coordinates describing the origin of the {p}-frame. This simplification is only valid if the three lines given by the axis of the motor, the connecting line of the proximal S-joints and the connecting line of the distal S-joints stay perfectly parallel to each other. For that 18 parameters will be fixed to their nominal values.

$$\Delta \overline{\overline{\mathbf{T}}}_i = \overline{\overline{\mathbf{I}}}, \quad \Delta \overline{\overline{\mathbf{C}}}_i = \overline{\overline{\mathbf{0}}}, \quad \Delta Lb_i = 0 \quad \text{Parameter set P1} \quad i=1..3 \quad (3.14)$$

Substituting eq. 3.13 and eq. 3.14 in eq. 3.12 shows that the $\overline{\mathbf{b}}_i$ - as well as the $\overline{\mathbf{d}}_i$ -Vector containing all together another 12 parameters vanish.

$$\overline{\mathbf{b}}_i, \overline{\mathbf{d}}_i \rightarrow \text{vanish} \quad \text{Parameter set P2} \quad i=1..3 \quad (3.15)$$

Geometrically, this corresponds to the reduction of the end-effector to a single point and the degeneration of the R(2S/2S)] joint-link train to a R2S chain (fig. 3.11). A further consequence of equation 3.15 is the degeneration of the second set of eqs. G2 given in 3.12 to identity $0 = 0$ whereas the first set G1 leads to model 24:

$$\begin{aligned} \overline{\overline{\mathbf{C}\mathbf{B}}}_i^T \cdot \overline{\overline{\mathbf{C}\mathbf{B}}}_i &= Lb_i^2 \\ \text{with} \\ \overline{\overline{\mathbf{C}\mathbf{B}}}_i &= \overline{\overline{\mathbf{P}}} - \overline{\overline{\mathbf{T}}}_i \cdot (\overline{\overline{\mathbf{D}}}_i + \overline{\overline{\mathbf{Q}}}_i \cdot \overline{\overline{\mathbf{L}\mathbf{a}}}_i) \end{aligned}$$

$$\rightarrow \quad i=1..3 \quad (3.16)$$

This model can be applied to the Delta robot assuming that its end-effector always stays perfectly parallel to the base.

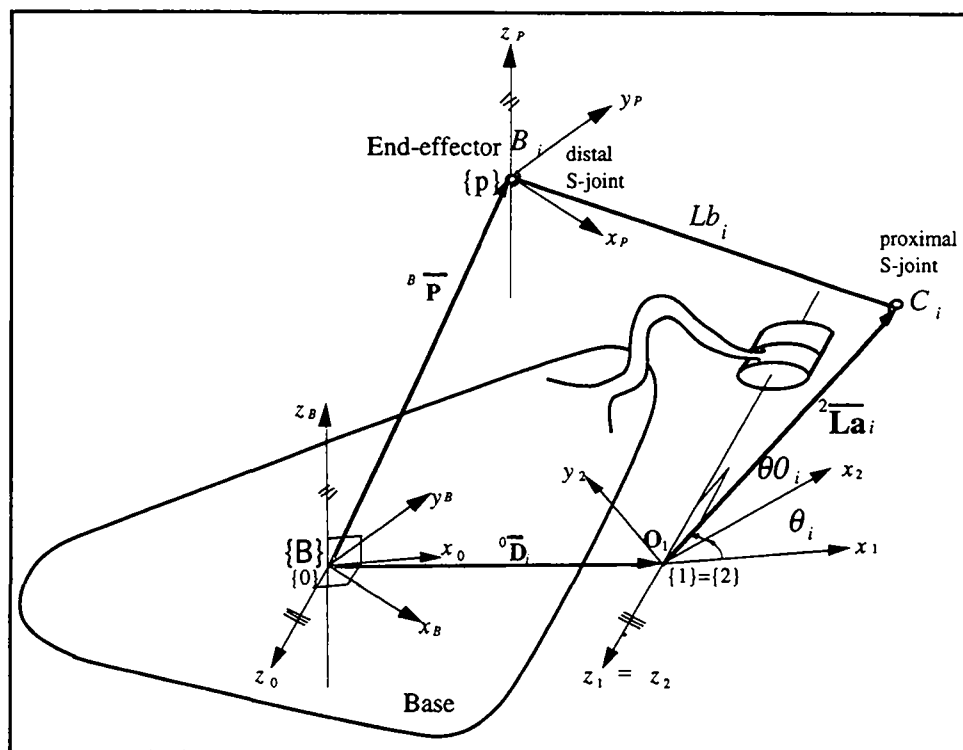


Fig. 3.11: Geometric interpretation of model 24 as a spatial 3[R2S] structure

3.4.5 The nominal model

To convert model 24 into the nominal model proposed by Clavel [91], the part of table 3.10 containing the geometric parameter will be rewritten as:

Scalars, vectors and rotation matrices containing the 6 geometric parameters:	
$\bar{\mathbf{T}}_i = Rot3(z, \theta_i) \cdot Rot3(x, \pi/2)$	for a symmetrical Delta robot: $\theta_i = 2 * (i - 1) * \pi/3 \quad i=1..3$
$\bar{\mathbf{D}}_i = \{R, 0, 0\}^T$	$R = R_b - R_a$ the difference between the radius of the base and the end-effector
$\bar{\mathbf{L}}a_i = \{La, 0, 0\}^T$	unique arm length and no encoder offset
$Lb_i = Lb$	unique length of the forearms

Table 3.12: Parameter collection for the nominal model derived from table 3.10

Substituting table 3.12 into eq. 3.16 leads to the nominal model suitable for the Delta mock-up.

3.4.6 Conclusion

In this section two complete and proportional calibration models with 54 and 24 independent kinematic parameters for the Delta robot were established. As shown in chapter 5 and 6 the difference of the complexity of these models is enormous. A possible question is whether there is another model between model 54 and model 24 with an intermediate mathematical complexity. However, in spite of an intensive search, no such model could be found.

Several characteristic properties based on the chosen parameterization of model 54 are listed below:

- a) The first set of closure eqs. G1 contains model 24, whereas the second set G2 will degenerate to identity (eq. 3.12)
- b) Only 30 of the 54 parameters have an influence on the orientation of the end-effector. These 30 parameters could further be split into two subsets:
- c) The generating set P1 contains 18 parameters of magnitude Δ reflecting small errors in the joint-link train (eq. 3.14)
- d) The amplifying set P2 contains 12 parameters of magnitude 1 describing the dimensions of the end-effector and the distance between the forearms (eq. 3.15)

These two sets are related in an interesting way: If P1 is considered to be zero, P2 has no influence on the pose of the end-effector. It can be said that the set P2 cannot generate pose errors by itself, but if P1 is not zero, it will amplify them. In figure 3.13 this is represented symbolically by a simple triangle.

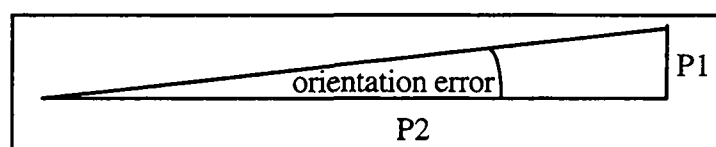


Fig. 3.13: Symbolical representation of the relation between the parameter set P1 and P2.

From figure 3.13 it can further be concluded:

- Variation of P1 affects the orientation much more than variation of P2. Thus, P1 is much more sensitive to orientation errors than P2. This shows that efforts to build Delta robots with the smallest possible orientation errors in the end-effector pose (cf. chapter 2) have to be concentrated on the 18 parameters of set P1.
- The fact that a parameter set is less sensitive to the resulting error of the end-effector can be also stated inverse. "A parameter which causes small end-effector errors over the whole working volume is nearly unobservable in the identification

phase". For the identification phase (paragraph 5.4.3) the parameter of set P2 will therefore be fixed to their nominal values and only the remaining 42 parameters will be identified.

- The bigger P2 becomes, the smaller is the orientation error. This is useful for the choice of the nominal parameters for an application of the Delta robot aiming at being very precise (e.g. milling center or measurement robot). In this case, P2 should be chosen as big as possible with respect to the remaining parameters. A larger distance between the forearms for example will decrease the resulting orientation error of the end-effector.

3.5 Models for the Argos structure

3.5.1 Introduction

In this section two different calibration models for the Argos structure will be developed. In contrast to the modeling performed so far, non-arbitrarily located base and moving frames will be used, allowing to solve the direct problem without changing the parameterization as will be shown in chapter 6.

In the following section the independent parameters are always given under the assumption that the base frame $\{b\}$ and moving frame $\{p\}$ are not arbitrarily chosen (eq. 3.3: $F = 0$).

According to eq. 3.3 a model taking all design errors into account would have had 126 parameters. Neglecting the errors of the spherical joint leads to 57. Attempting to establish this model shows that the dependent joint variables of the six passive revolute joints cannot be bypassed. Therefore, twelve closure equations for the twelve unknowns would be needed.

The fact that the pantograph behaves in a perfect manner, i.e. the S-joints attached to it are moving on a circle is introduced as a further simplification. This assumption is justified by the efforts described in chapter 2. This leads to a spatial 3[2RS] structure similar¹ to the ones of LeeK [88] or Magnani [95]. The model of this structure will be referred to as model 27 according to the 27 parameters needed to be complete.

¹ Similar in the sense of having the same solution of its direct problem.

Claiming that the three S-joints move on the same sphere leads to a model with 9 independent parameters.

A last simplification step to get the nominal model of the Argos mock-up is made by assuming that the axes of the motors as well as the vectors pointing to the S-joints stays perfectly perpendicular to one another. Some guidelines about the calibration models 27 and 9 are listed below:

- The parameterization for all three models should be the same.
- Homogenous matrices will be used for the spatial model 27. They will degenerate to pure rotation matrices for the remaining two spherical models.
- The pantographs assumed to be perfect will perform a circle about a virtual axis orthogonal to the pantograph plane (fig. 3.14). For modeling reasons a pantograph is therefore replaced by a single rotative joint on this virtual axis.
- Twelve joint angles are passively moving and therefore depending variables. The closure equations are established without dependent variables.
- All distances are made dimensionless by dividing by the nominal radius R of the spherical structure.

3.5.2 Parameterization

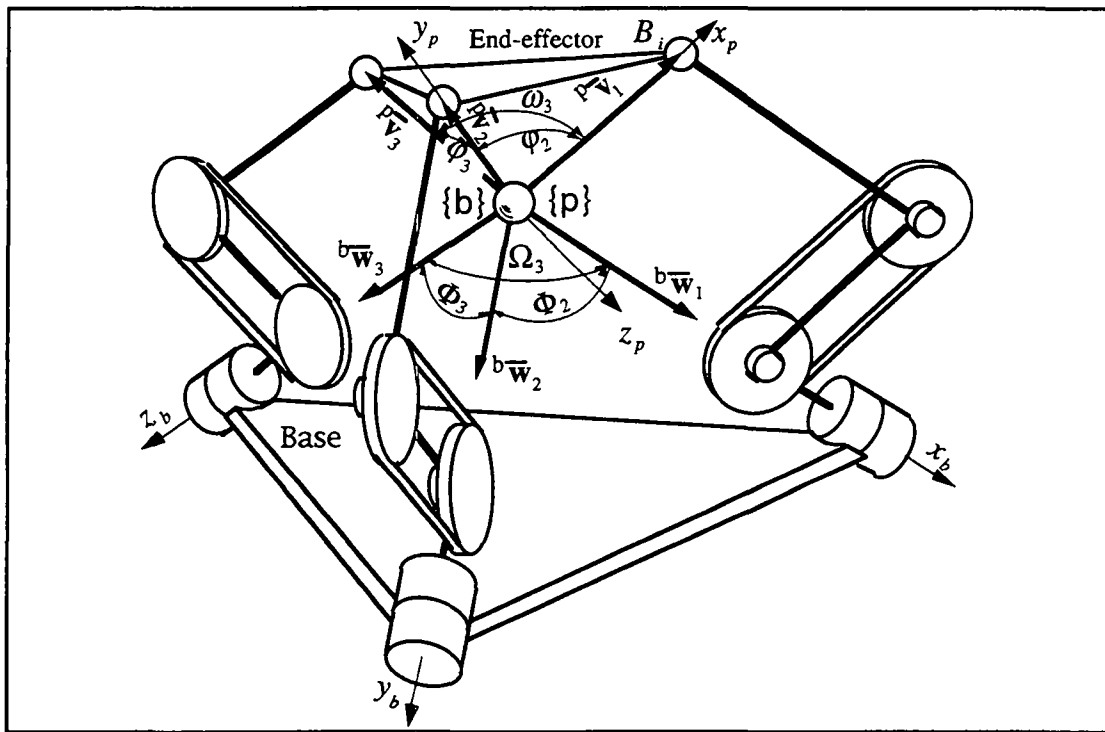


Fig. 3.14a: Parameterization of a Argos structure *without* geometric deviations.

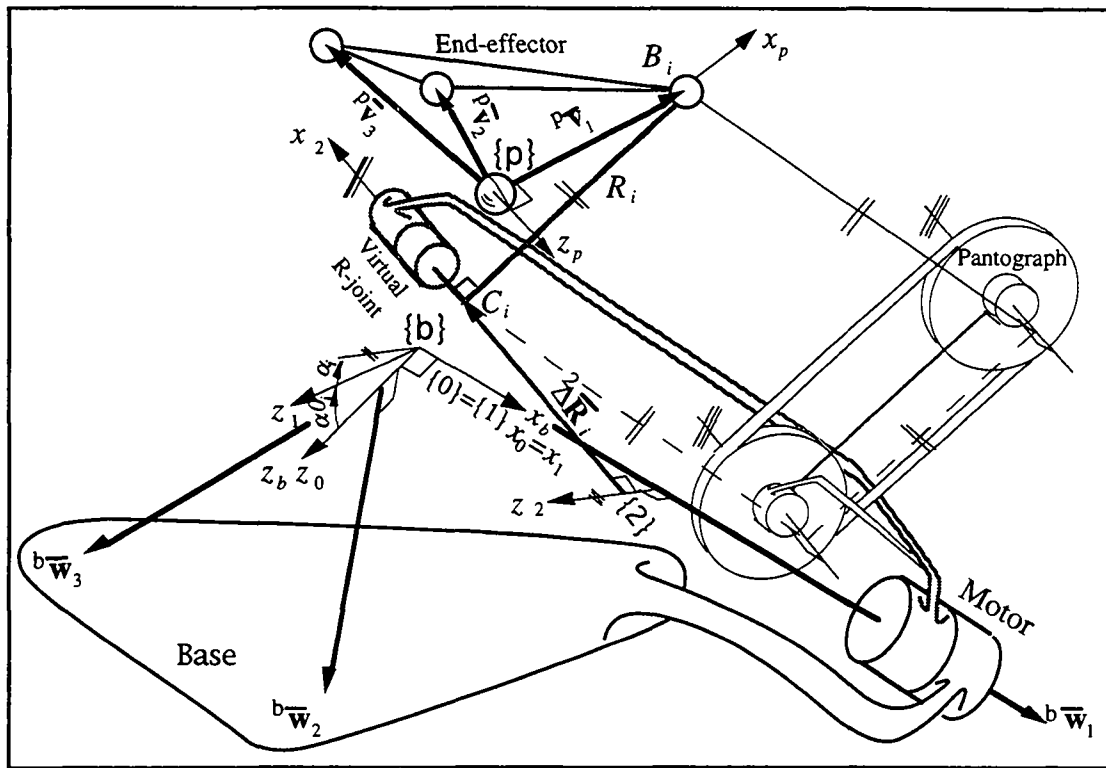


Fig. 3.14b: Parameterization of one joint-link train of the Argos structure *with* geometric deviations. The pantograph is thought to be replaced by one virtual R-joint. Thus, the modeling is done with the 2RS joint-link train as represented in gray bold lines.

According to figure 3.14a and 3.14b the following points and frames are defined:

- B_i : Center points of the S-joints assumed to be perfect;
- C_i : Projection points of B_i on the axes of the virtual R-joints;
- $\{b\}$: The base frame $\{b\}$ is attached to the base with its x-axis coincident with the first motor axis and its z-axis coincident with the common normal between the first motor axis and the second;
- $\{p\}$: The moving frame $\{p\}$ is attached to the end-effector in such a way that all three S-joints are located on a unit-sphere around its origin. Furthermore, the x-axis of the $\{p\}$ -frame points to B_1 whereas B_2 lies in its xy-plane;
- $\{0\}$: The motor frame $\{0\}$ is attached to the motor axis. The $\{0\}$ -frame of the first chain is coincident with the frame $\{b\}$;
- $\{1\}$: The twisted motor frame $\{1\}$ is the $\{0\}$ -frame twisted by the motor angle;
- $\{2\}$: The virtual R-joint frame $\{2\}$ is attached to the axis of the virtual R-joint.

Homogeneous matrix containing the six world coordinates:	
${}^p\bar{\bar{T}} = Trans4(x, y, z) \cdot Rot4(x, \vartheta_1) \cdot Rot4(y, \vartheta_2) \cdot Rot4(x, \vartheta_3)$	Homogeneous transformation matrix describing the moving frame {p} relative to the base frame {b} containing three Euler angles and three Cartesian coordinates
Rotation matrices containing the three joint coordinates:	
${}^0_1\bar{Q}_i = Rot4(x, \alpha_i)$	Homogeneous rotation matrix containing the motor angle
Homogeneous matrices, vectors and scalars containing the 27 geometric parameters:	
${}^b_0\bar{\bar{S}}_1 = \bar{\bar{I}};$ ${}^b_0\bar{\bar{S}}_2 = Trans4(0, 0, \Delta C_2) \cdot Rot4(z, \Phi_2);$ ${}^b_0\bar{\bar{S}}_3 = Rot4(x, \Omega_3) \cdot Trans4(\Delta A_3, 0, \Delta C_3) \cdot Rot4(z, \Phi_3)$	Homogeneous transformation matrix describing the motor frame {0} relative to the {b}-frame DH-parameters for $\approx \perp$ axes ¹
${}^1_2\bar{\bar{T}}_i = Rot4(x, \alpha 0_i) \cdot Trans4(\Delta a_i, 0, \Delta c_i) \cdot Rot4(z, \gamma_i)$	Homogeneous transformation matrix describing the virtual joint frame {2} relative to the twisted motor frame {1} using DH-parameters for $\approx \perp$ axes.
${}^b\bar{w}_i = {}^b_0\bar{\bar{S}}_i \cdot \{1, 0, 0, 1\}^T$	Unit vector lying on the motor axis
${}^p\bar{v}_1 = \{1, 0, 0, 1\}^T;$ ${}^p\bar{v}_2 = Rot4(z, \varphi_2) \cdot \{1, 0, 0, 1\}^T;$ ${}^p\bar{v}_3 = Rot4(x, \omega_3) \cdot Rot4(z, \varphi_3) \cdot \{1, 0, 0, 1\}^T;$	Unit vector pointing from the origin of the moving frame {p} to the point B_i , the center point of the S-joint.
${}^2\Delta\bar{R}_i = \{\Delta R_{ix}, 0, 0, 1\}^T;$	Vector pointing from the origin of the frame {2} to point C_i , the projection point of B_i on the axis of the virtual R-joint
R_i	Radius of the circle of the perfect pantograph

Table 3.15: Parameterization of one of the three 2RS-joint link trains {i} of the Argos structure

¹As mentioned in paragraph 3.2.2 DH-parameterization does only define how the origin of the consecutive frame is defined and *not* which axis of the frame have to be located on the joint axis.

3.5.3 Model 27

According to Hiller [88] the virtual link between the virtual R-joint and the S-joint will be removed as "characteristic pair of joints" of the type "distance of a point from a line". The resulting closure equations are free of dependent, passive joint variables. Two different claims have to be satisfied:

The *first* claim is that the vector between C_i and B_i must stay perpendicular to the axis of the passive R-joint (${}^2\bar{\mathbf{u}}_i$) leads to a first set of closure equations (G1) which will be established in the {b}-frame:

$$G1: {}^b\bar{\mathbf{u}}_i^T \cdot {}^b\bar{\mathbf{CB}}_i = 0 \quad i=1..3 \quad (3.17)$$

with

$${}^2\bar{\mathbf{u}}_i = (\{1,0,0,1\} - \{0,0,0,1\})^T = \{1,0,0,0\}^T \quad i=1..3 \quad (3.18)$$

$${}^b\bar{\mathbf{u}}_i = {}^b\bar{\mathbf{S}}_{i,1}^0 \cdot \bar{\mathbf{Q}}_{i,2}^1 \cdot \bar{\mathbf{T}}_i^2 \cdot {}^2\bar{\mathbf{u}}_i \quad i=1..3 \quad (3.19)$$

$${}^b\bar{\mathbf{CB}}_i = {}^b\bar{\mathbf{T}}^p \cdot \bar{\mathbf{v}}_i - {}^b\bar{\mathbf{S}}_{i,1}^0 \cdot \bar{\mathbf{Q}}_{i,2}^1 \cdot \bar{\mathbf{T}}_i^2 \cdot \Delta\bar{\mathbf{R}}_i \quad i=1..3 \quad (3.20)$$

The *second* claim is that the Euclidean norm of the vector between C_i and B_i must be equal to the radius R_i of the corresponding circles in the pantograph planes which leads to a second set of closure equations (G2). This set is established in the {b}-frame, too:

$$G2: {}^b\bar{\mathbf{CB}}_i^T \cdot {}^b\bar{\mathbf{CB}}_i = R_i^2 \quad i=1..3 \quad (3.21)$$

Model 27 is now established by eq. 3.17 and eq. 3.21. For simplicity the leading sub- and superscript will be dropped:

$$\begin{aligned} G1: & \quad \bar{\mathbf{u}}_i^T \cdot \bar{\mathbf{CB}}_i = 0 \\ G2: & \quad \bar{\mathbf{CB}}_i^T \cdot \bar{\mathbf{CB}}_i = R_i^2 \\ \text{with} & \\ \bar{\mathbf{u}}_i &= \bar{\mathbf{S}}_i \cdot \bar{\mathbf{Q}}_i \cdot \bar{\mathbf{T}}_i \cdot \{1,0,0,0\}^T \\ \bar{\mathbf{CB}}_i &= \bar{\mathbf{T}} \cdot \bar{\mathbf{v}}_i - \bar{\mathbf{S}}_i \cdot \bar{\mathbf{Q}}_i \cdot \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{R}}_i \end{aligned}$$

$$\rightarrow \quad i=1..3 \quad (3.22)$$

This model can be applied to the Argos structure assuming that the pantographs stay in planes and that the S-joints attached to it describe perfect circles of radius R_i .

¹ ${}^2\bar{\mathbf{u}}_i$ is a *free* unit vector. Such a vector is constant in length and direction and can freely be moved in space. Its homogeneous component is *zero* (translations don't affect a free vector). A free vector stays in contrast to a *bounded* vector (to a point) with its homogeneous component equal to one.

3.5.4 Model 9

This model is a simplification of model 27. It is based on the following *two* assumptions:

1. The structure is perfectly spherical. All three S-joints must remain on a single sphere of radius one about the virtual rotation center. Geometrically, this simplification can be interpreted as a spherical 3[RPS] structure (Fig. 3.17). The number of independent parameters for a complete model is 12 when calculated according to eq. 3.4.
2. The motor axes are located on the pantograph planes. To model an arbitrarily located pantograph plane relative to the motor axis is unrealistic in relation to the already introduced simplifications.

The first simplification claims that the virtual rotation center of the structure stays fixed:

$$\{x, y, z\}^T = \bar{\mathbf{0}} \quad (3.23)$$

The number of world coordinates is reduced to three Euler angles describing the orientation of the {p}-frame relative to the {b}-frame. This simplification is only valid if 9 parameters are set to their nominal values and 3 constraints are fulfilled:

1. *Chain:* $\Delta a_1 = \Delta c_1 = \Delta R_{1x} = 0, R_1^2 + \Delta R_{1x}^2 = 1$
2. *Chain:* $\Delta C_2 = \Delta a_2 = \Delta c_2 = \Delta R_{2x} = 0, R_2^2 + \Delta R_{2x}^2 = 1,$
3. *Chain:* $\Delta A_3 = \Delta C_3 = \Delta a_3 = \Delta c_3 = \Delta R_{3x} = 0, R_3^2 + \Delta R_{3x}^2 = 1;$

(3.24)

The second simplification claims that the virtual link stays perpendicular to the motor axis and the perfect circles are great circles on the sphere:

$$\gamma_i = \pi/2 \quad R_i = 1 \quad \Delta R_{ix} = 0 \quad i=1..3 \quad (3.25)$$

Substituting eq. 3.23, 3.24 and 3.25 into table 3.15 shows that all transformation matrices are converted into pure rotation matrices, which allows dropping of the homogenous representation (table 3.16). Furthermore, the second set of eqs. G2 in eq. 3.22 degenerates to identity $1=1$. If eqs. 3.24 and 3.25 are substituted into the first set G1 given in 3.22, model 9 can be found after some manipulations. For simplicity the leading sub- and superscript will be dropped. The vectors $\bar{\mathbf{u}}_i, \bar{\mathbf{v}}_i$ as well as the rotation matrix $\bar{\mathbf{R}}$ are given in table 3.16:

$$\rightarrow \boxed{\bar{\mathbf{u}}_i^T \cdot \bar{\mathbf{R}} \cdot \bar{\mathbf{v}}_i = 0} \quad i=1..3 \quad (3.26)$$

This model can be applied to the Argos structure assuming that its virtual rotation center stays stationary, thus, that the structure is perfectly spherical. A second simplification is the assumption that the motor axis lies on the pantograph plane and the the three circles traced by the S-joints are great circles.

Rotation matrix containing the three world coordinates	
$\bar{\mathbf{R}} = Rot3(x, \vartheta_1) \cdot Rot3(y, \vartheta_2) \cdot Rot3(x, \vartheta_3)$	Rotation matrix describing the moving frame
Rotation matrices containing the three joint coordinates	
$\bar{\mathbf{Q}}_i = Rot3(x, \alpha_i)$	Rotation matrix containing the motor angle
Rotation matrices, vectors and scalars containing the 9 geometric parameters	
$\bar{\mathbf{S}}_1 = \bar{\mathbf{I}}; \bar{\mathbf{S}}_2 = Rot3(z, \Phi_2); \bar{\mathbf{S}}_3 = Rot3(x, \Omega_3) \cdot Rot3(z, \Phi_3)$	Rotation matrix describing the motor frame
$\bar{\mathbf{u}}_i = \bar{\mathbf{S}}_i \cdot \bar{\mathbf{Q}}_i \cdot Rot3(x, \alpha_{0i}) \cdot \{0, 1, 0\}^T$	Vector perpendicular to the twisted pantograph plane
$\bar{\mathbf{v}}_1 = \{1, 0, 0\}^T; \bar{\mathbf{v}}_2 = Rot3(z, \varphi_2) \cdot \{1, 0, 0\}^T;$ $\bar{\mathbf{v}}_3 = Rot3(x, \omega_3) \cdot Rot3(z, \varphi_3) \cdot \{1, 0, 0\}^T;$	Unit vector pointing to the S-joint.

Table 3.16: Parameter collection¹ for the model 9 derived from table 3.15

Figure 3.17 shows a geometric interpretation of model 9 as a spherical 3[RPS] structure with perfectly aligned spherical prismatic joints (P). Instead of actuating the R-joints, the spherical P-joints could be actuated by driving the pantograph in a push-pull manner with a conical gear wheel on its second R-joint [Vischer 95]. This arrangement would correspond to a spherical "Stewart"-Platform with its planar complement shown in figure 3.2.

¹A more detailed definition of the different parameters of model 9 can be found in [Bubendorf 96]. However, the model presented here is slightly changed in order to remove a mathematical singularity.

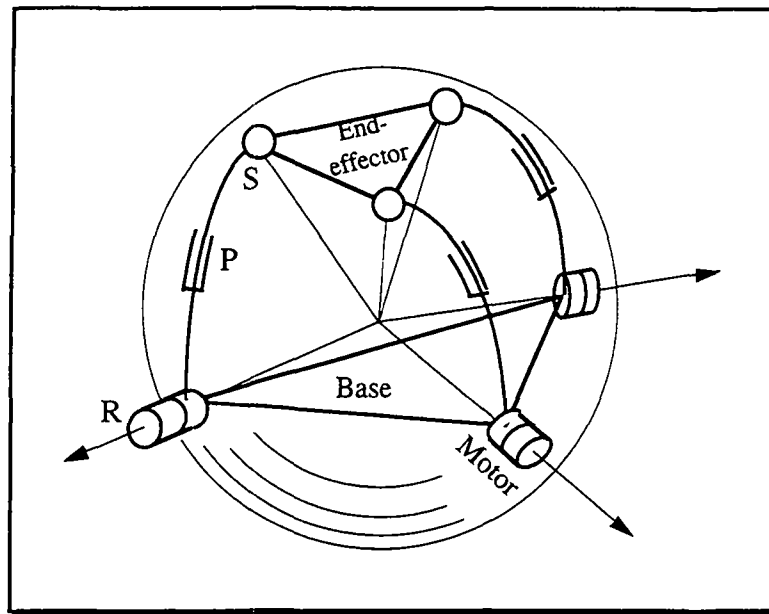


Fig. 3.17: Geometric interpretation of model 9 as a spherical 3[RPS] structure.

3.5.5 The nominal model

For the Argos mock-up the motor axes as well as the vector pointing to the S-joints are arranged orthogonally. Thus, six angles of the model 9 are fixed to 90 degrees to get the nominal model:

$$\begin{aligned}
 2. \text{ Chain: } \Phi_2 = \varphi_2 = \pi/2; \\
 3. \text{ Chain: } \Omega_3 = \Phi_3 = \omega_3 = \pi/2, \quad \varphi_3 = -\pi/2;
 \end{aligned}
 \tag{3.27}$$

The nominal model is given by the substitution of table 3.18 into eq. 3.26.

Rotation matrices containing the three joint coordinates	
$\bar{\bar{Q}}_1 = Rot3(x, \alpha_1), \quad \bar{\bar{Q}}_2 = Rot3(y, \alpha_2), \quad \bar{\bar{Q}}_3 = Rot3(z, \alpha_3)$	Rotation matrix containing the motor angle
Rotation matrices, vectors and scalars containing the 3 geometric parameters	
$\bar{u}_1 = \bar{\bar{Q}}_1 \cdot Rot3(x, \alpha_{01}) \cdot \{0, 1, 0\}^T$ $\bar{u}_2 = \bar{\bar{Q}}_2 \cdot Rot3(y, \alpha_{02}) \cdot \{-1, 0, 0\}^T$ $\bar{u}_3 = \bar{\bar{Q}}_3 \cdot Rot3(z, \alpha_{03}) \cdot \{-1, 0, 0\}^T$	Vector perpendicular to the twisted pantograph plane
$\bar{v}_1 = \{1, 0, 0\}^T; \quad \bar{v}_2 = \{0, 1, 0\}^T; \quad \bar{v}_3 = \{0, 0, -1\}^T;$	Unit vector pointing to the S-joint

Table 3.18: Parameters of the nominal model derived from table 3.16 with the aide of eq. 3.27

3.5.6 Conclusion

In this section two proportional and complete calibration models for the Argos structure were proposed. As for the calibration models of the Delta robot the first set G1 of model 27 (eq. 3.22) contains model 9 whereas the second set G2 degenerates to identity.

3.6 Error propagation in serial compared to parallel robots

3.6.1 Introduction

In this section an application of the developed calibration models is shown in order to simulate the pose errors of the end-effector caused by estimated tolerances of the mechanical parts of the robot. In figure 3.19, seven poses of Delta's end-effector following an elliptical trajectory are shown with its six forearms subjected to length deviations. As expected orientation errors at the end-effector result.

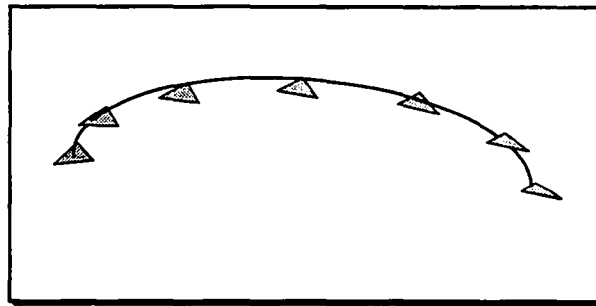


Fig. 3.19: Simulation based on model 54 of the end-effector's pose errors of the Delta robot subjected to geometric deviations in its structures.

The aim of this section is to compare by simulation the accuracy at the end-effector of a serial SCARA robot and a parallel Delta robot assuming the same tolerance field for the mechanical parts.

3.6.2 Results from simulation

The following criterion for equivalence of the two robots has been introduced by Clavel [91]:

$$a_0 + a_1 = L_a + L_b \quad (3.28)$$

The nominal length of the two arms of the SCARA must be equal to the sum of the nominal length of the forearm and the arm of a Delta robot. The arm of the Delta mock-

up has a length of 120 mm and the forearm of 240 mm. The arm segments of the SCARA were chosen to be 200 and 160 mm long, which satisfies eq. 3.28. The calibration model with 16 parameters for the SCARA robot is given in paragraph 3.2.3, whereas for the Delta robot model 54, presented in paragraph 3.4.3 is used. As tolerances equally distributed random values within an estimated tolerance field were chosen. The chosen field of tolerance is:

$$\begin{aligned} \text{linear: } & \pm 0.05 \text{ mm} \\ \text{angular: } & \pm 0.1^\circ \end{aligned} \quad (3.29)$$

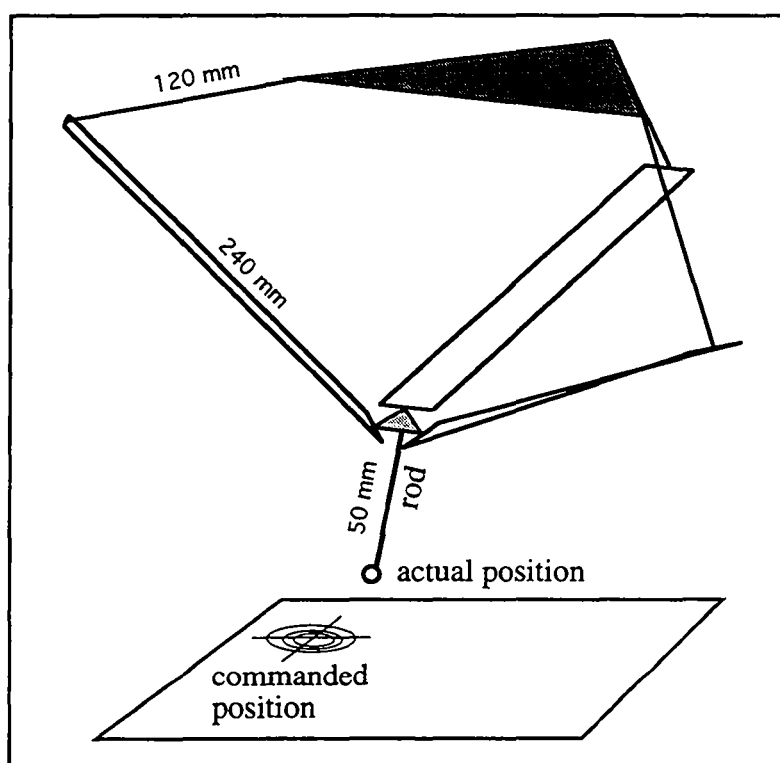


Fig. 3.20: Error between the commanded and actual position of the rod's tip

According to figure 3.20 a rod of 50 mm was fixed to the end-effector. With the tip of this rod the robot should follow a xy-plane located in the middle of its working volume. Using the nominal model the corresponding joint angles for a point in the plane were calculated. These joint angles were used to calculate, with the help of the calibration model, the real pose of the end-effector in space. The actual position of the rod's tip was then compared to the commanded position. This kind of simulation shows the ability of a robot, with design errors to follow a trajectory. A typical example for this kind of task is a CNC-milling center. Thus, the simulation was performed according the flow diagram represented in figure 3.21:

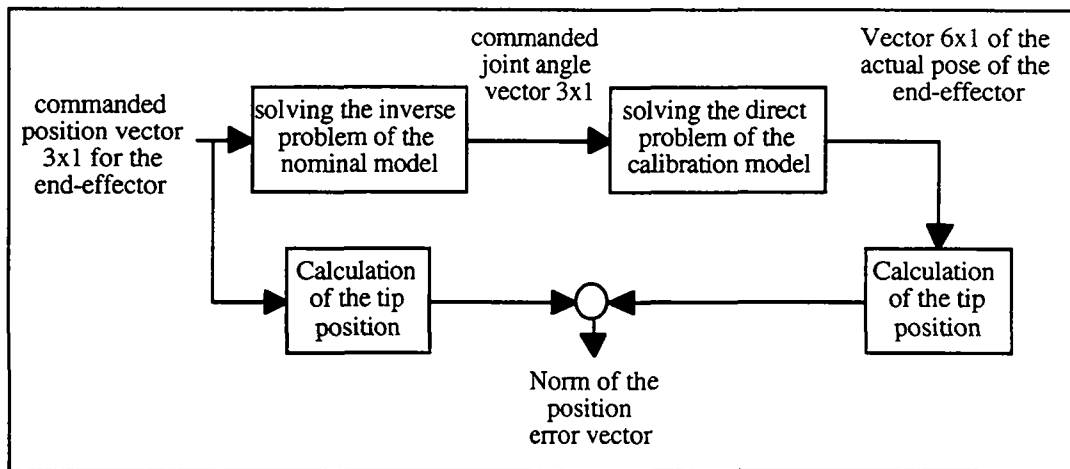


Fig. 3.21: Flow diagram for the simulation of a milling center

The resulting norm of the position error is shown in Figure 3.22:

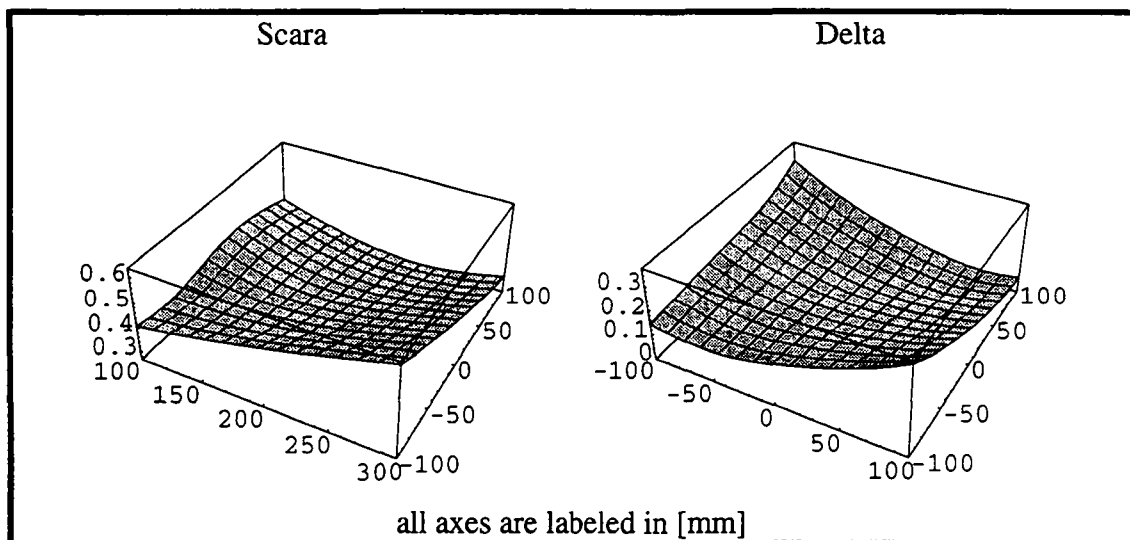


Fig. 3.22: Resulting position error of the rod's tip for a Scara and Delta robot scanning a plane.

Figure 3.22 shows that the variation is about 0.3 mm for both robots whereas the mean values are different, with 0.5 mm for the Scara and 0.2 mm for the Delta robot. These results confirm the conclusion that the accuracy of parallel structures is of the same order of magnitude as for comparable serial structures [Lin 89, Wang 92]. It seems however that the mean value for parallel structures is smaller than for serial ones, which is slightly advantageous.

The situation is different if a measurement robot should be simulated. In this case a piece to be measured, e.g. a cube, is lying in the workspace of the robot. The robot establishes physical contact with the cube via a touch probe. The position of the touch

probe is then calculated with the nominal model in order to know where the touched point on the surface of the cube is located in space. This situation is summarized in the flow diagram of figure 3.23.

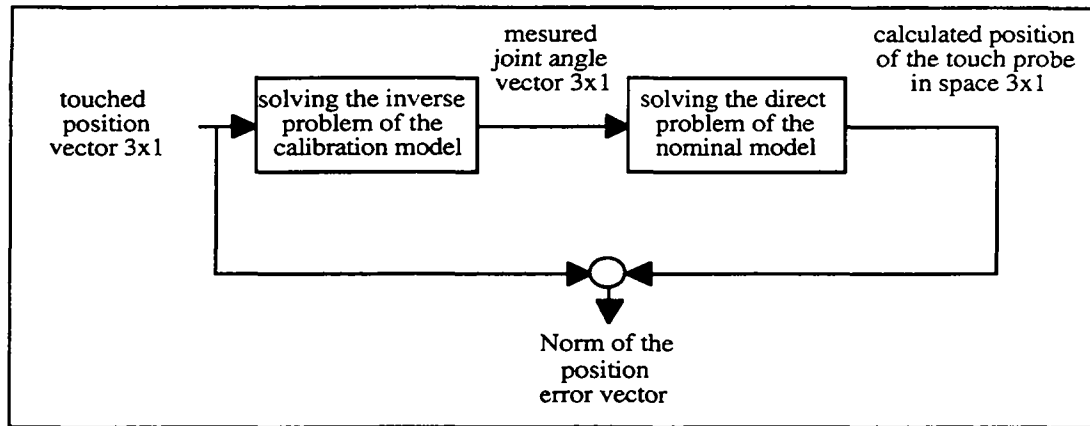


Fig. 3.23: Flow chart for the simulation of a measurement robot

3.7 Conclusion

In this chapter some fundamental properties for good parametric calibration models were discussed. These theoretical tools were used to establish models for the Scara, Stewart, Delta and Argos structures. The importance of the choice of parameters is not to be underestimated and modeling is therefore a very important step in calibration.

4. Measurement

4.1 Introduction

In this chapter experimental set-ups for the collection of measurement data for the calibration process will be proposed for both the Delta and the Argos structure. For identification some *redundant information* about the structure to be calibrated must be provided, which is the principal goal of the measurement step.

The measurement devices can be grouped into two classes such as internal and external transducers. Internal transducers are devices which belong to the robot such as its encoders for example, whereas external transducers are added only during the measurement phase such as for example a 3D-measuring machine. The redundant information is gained by establishing a loop between any link of the robot and the base. Particularly interesting is the closure between the end-effector and the base. This particular loop is referred to as the *measurement loop*.

The measurement loop can be established in an active way using for example a system of external theodolites or in a passive way using passive joints of high precision for examples. A further grouping into contact and non-contact measurement is possible. The advantage of non-contact measurement is that the measurement device doesn't transmit contact forces to the end-effector. Thus, its pose is not disturbed by the presence of the measurement system. This becomes particularly important if the measurement should be performed while the robot is in motion (dynamic measurement).

How much information can be gained for each measurement point depends on how many times the measurements are redundant. If all six degrees of freedom of the end-effector are actively measured the full pose of the end-effector is available, which leads to six error equations for each measured point. Thus, full pose information is six times redundant. Partial pose information is available for a robot sliding with the tip of its end-effector on a plane. Knowing that each point lies on this plane, a single error equation is gained per point taken, which corresponds to a single redundancy.

For the first group using external transducers many propositions can be found in literature. Further added to this group are set-ups using any kind of pre-measured fixtures [Tanner 90, Zobel 93, Hollerbach 93] such as for examples a block with holes of precisely known position [Veitschegger 87]. Van Brussel [90], Felder [90], Tradt [91] and Mooring [91] have given good overviews of different commercially available systems. Non-contact measurement devices such as the SMART-system of Leica [Markendorf 90, Kyle 93a] is very expensive (250'000 Swiss francs). The SMART system is based on a Laser interferometer, which can be deflected over two mirrors. It can track the position of a corner cube fixed on the robot's end-effector up to a speed of 4 m/s and accelerations of 2 g with a resolution of 0.7 arcseconds and 1.3 micrometers. The SMART system is comparable to the 3D-Laser tracker from Chesapeake. A system without interferometer called Lasertrace has been developed at the University of Surrey and is commercially available from Automatic System Laboratories (ca. 40'000 Swiss francs) [Parker 87, Mayer 94]. A system in addition capable of measuring the orientation of the end-effector without contact was developed by Prenninger [93]. Other non-contact systems are based on two motorized theodolites as for examples the ATMS or SPACE system by Leica [Kyle 93b]. The position measurement is based on triangulation of a target fixed on the end-effector whereas the tracking is performed with the help of a CCD-camera. This kind of system is about 10% - 20% more expensive than the already mentioned 3D-Laser tracker SMART. Selspot II is a system based on CCD cameras sold by the Swedish company Selspot. This small selection of propositions from literature for measurement devices shows already that there is no single low-cost device covering all the needs. Since this work is restricted to static calibration only, there is no need for non-contact measurement. This allows to use coordinate measuring machines and digital probes, which will be used for the experimental set-up of both the Delta and the Argos structures.

For the second group without external transducers [Wampler 92, Sayeh 94], Hornick [91] proposed to use a single flat plane to establish redundancy in the transducer readings of the robot [Zhong 95]. Using several planes for a serial robot, calibration was successfully simulated by Flury [94]. Khalil [95] proposed to lock the end-effector of a 6R robot to the base in the eight different, possible configurations. Benett's [91] method for the calibration of a redundant manipulator consists of fixing the robot's end-effector to the base and moving this closed-loop into different configurations. Driels [91] proposed to close the measurement loop to the base by a rod with ball-socket joints at both of its ends (ballbar). All these set-ups have in common their strong dependence on the robots topology and its degrees of freedom. The number of passive set-ups using no external measurement devices is proportional to the degrees of freedom of the robot.

Non-contact set-ups as for instance with a linear air bearing cancels partially the advantage of a low price. Two propositions, belonging to this class, were built for the Delta robot, namely the *plane/sphere set-up* and the *short-cut set-up*.

The accuracy for the different measurement units given in this chapter are calculated from the specifications given for the different transducers and under the assumption of a perfect geometry. In fact, it would be necessary to calibrate each measurement unit first and to measure the resulting accuracy. Thus, the given values correspond to the upper bound of accuracy.

4.2 Set-ups for the Delta mock-up

4.2.1 Introduction

In this section an external full-pose measurement system for the mock-up of the Delta will be introduced. The aim is to provide a measurement system which is able to measure large changes of the end-effector's position, typically of the length of the forearm and small changes of the end-effector's orientation ($< \pm 5^\circ$). The chosen solution consists of a 3D-measuring machine where the end-effector of the mock-up is attached to the z-axis by means of a ball-socket joint, still allowing small changes of the orientation. These changes are measured by three digital probes, fixed on the z-axis of the measuring machine. Furthermore two measurement set-ups without external sensors are shown.

4.2.2 Position unit

Instead of adapting the measuring machine to the mock-up, the dimensions of the mock-up were optimized to fit an available 3D-measuring machine as shown in figure 4.1. The machine, a Validator 10, is built by the Swiss company TESA (Brown&Sharp) and has the following specifications:

Measurement volume x,y,z	300x300x120 mm
Resolution	5 μm
Accuracy (on 300 mm)	$\pm 10 \mu m$

The height is too low for the Delta mock-up and therefore three different base frames {B} were fixed at different heights of the mock-up (fig. 4.3). The measurement volume of the machine can be manually varied in the z-direction. The interface to a PC was

established by a digital data acquisition board (PC-DIO-96) of National Instruments and within the programming environment LabView™ [Egloff 95].

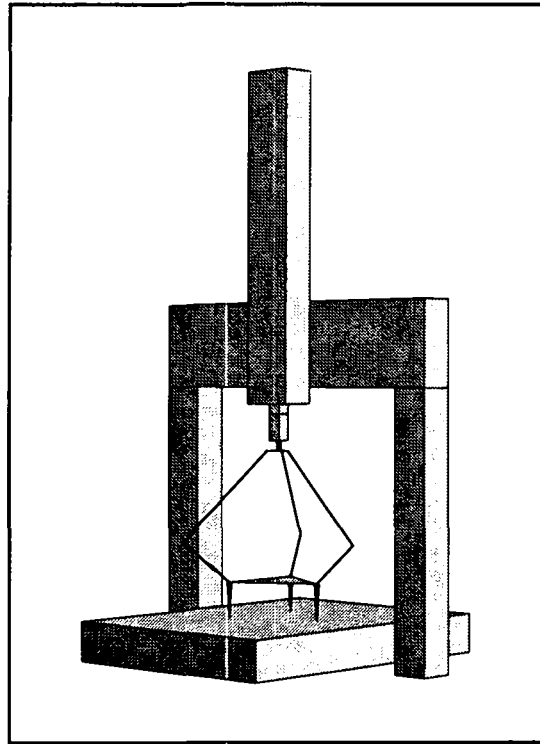


Fig. 4.1: Optimization of the dimensions of mock-up which has to fit underneath the measuring machine.

4.2.3 Orientation unit

Three linear digital touch probes TESA-GT22C orthogonally arranged to each other, are used to measure three fixed angles defining the orientation of the end-effector (fig. 4.2) with the following specifications:

Measurement volume roll, pitch, yaw	$\pm 4,7^\circ$ each
Resolution	1"
Accuracy (within $\pm 1.9^\circ$)	$\pm 15''$

The interface to the computer is established via an RS 232-link, which is supported by LabView™ [Egloff 95].

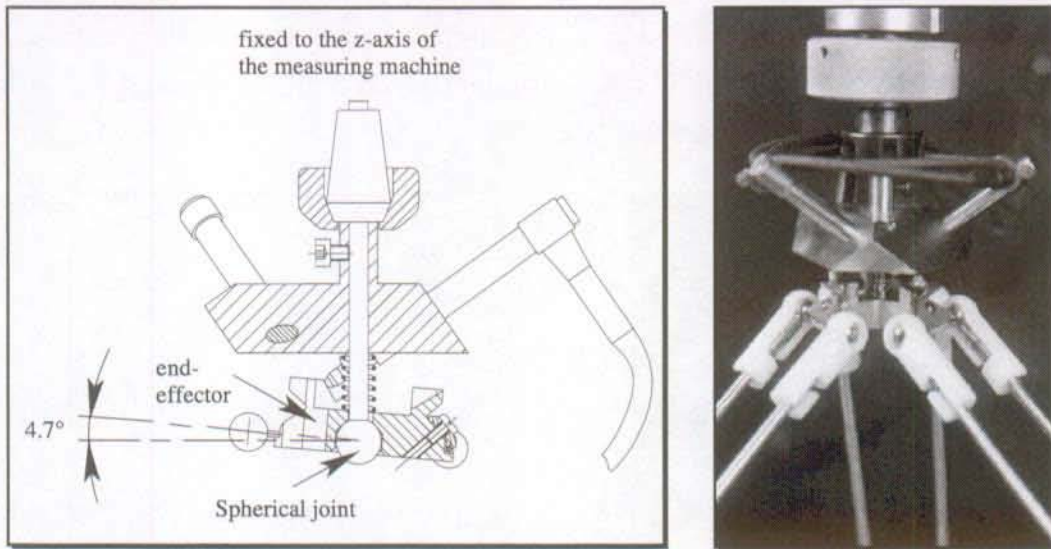


Fig. 4.2: Orientation unit for small changes

4.2.4 Full-pose measurement set-up

The mock-up is clamped on the cross table of the measuring machine by means of three solenoids as shown in figure 4.3

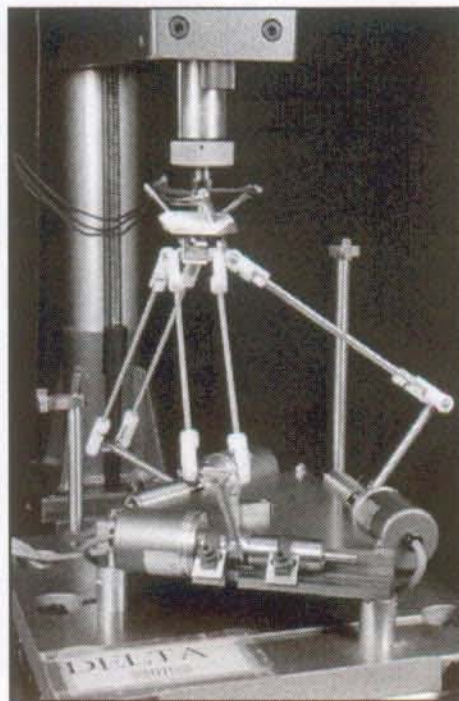


Fig. 4.3: Full-pose measurement set-up for the mock-up of the Delta robot

To align the measuring machine as well as the orientation unit with respect to the mock-up, the forearms must be removed and the z-axis of the measuring machine is

brought into physical contact with one of the three reference frames whereas the end-effector, which is still fixed to the z-axis, can be twisted until it fits into the V-groove and to the plane of the reference frame (fig. 4.4).

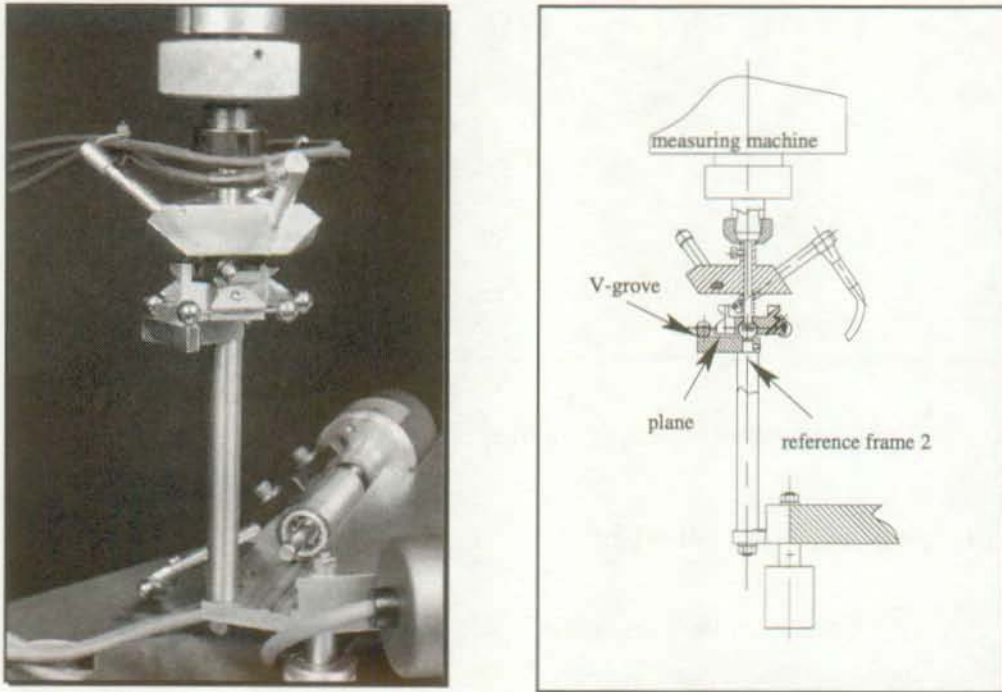


Fig. 4.4: Aligning the external measurement devices relative to the mock-up

4.2.5 Set-up in another configuration

Some design parameters cannot be observed (set P2, paragraph 3.4.6) using the configuration where the end-effector stays parallel to the base. As shown in figure 4.5 changing the configuration space allows the design parameters given in the P2-set to be observed. This set-up requires external measurement devices, too. In the chosen configuration the orientation of the end-effector is changing considerably and the measurement unit proposed in paragraph 4.2.3 has to be replaced by the unit proposed in paragraph 4.3.3.

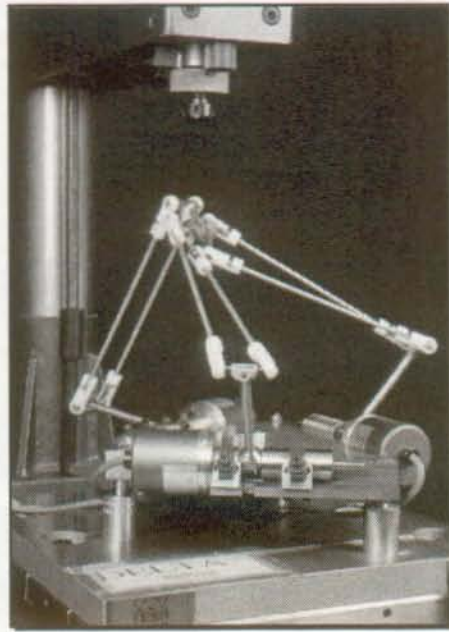


Fig. 4.5: The observability of the kinematic parameters changes for different configuration spaces. The configuration above is the same as solution with strongly varying orientation shown in figure 6.3.

4.2.6 Plane/Spheres set-up

The plane/spheres set-up uses no external measurement devices. During the data acquisition the end-effector is restricted to follow a plane and a sphere, respectively. This constraint reduces the degrees of freedom of the Delta's end-effector to two. Thus, the set-up gets redundant with respect to one encoder reading, which will be used to identify the kinematic parameters. According to Hornick [91] the plane can be posed arbitrarily in the working volume of the robot. However, due to the topology of the Delta robot and for practical reasons, the plane is chosen to be approximately parallel to the base. The pose of the robot's base frame $\{B\}$ relative to this plane is still not restricted with respect to the three remaining degrees of freedom (x, y, φ) . Thus, two spheres having their centers lying in the plane were added by means of two ballbars.

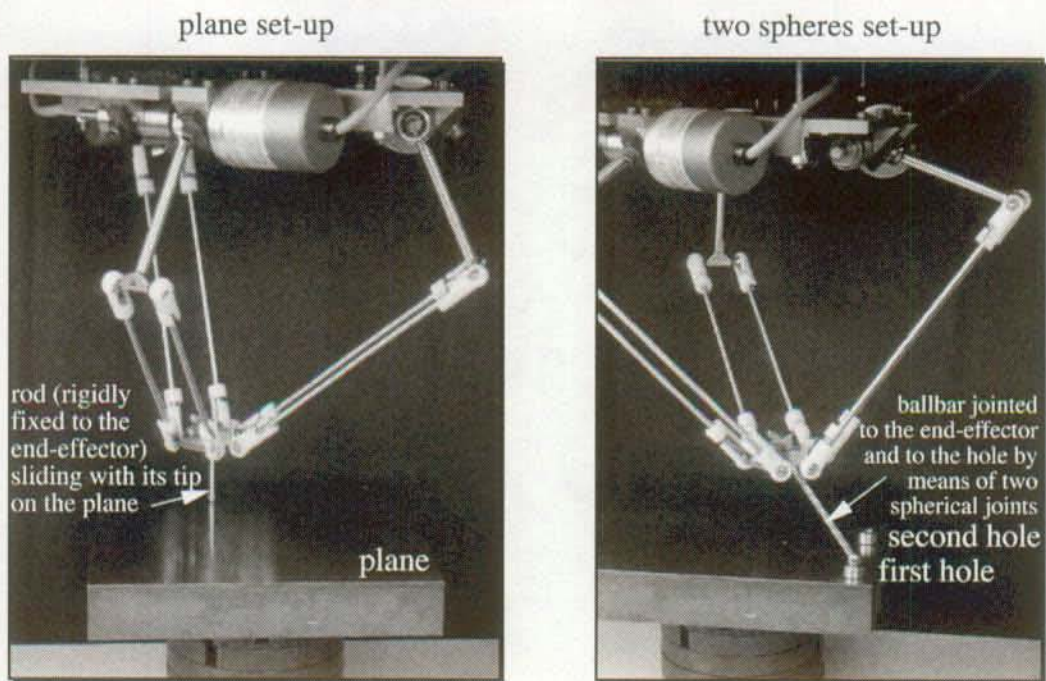


Fig. 4.6: The plane/spheres set-up

Instead of passive [Driels 91 Sayeh 94] also active ballbars can be used as proposed by Goswami [93] for partial pose measurement.

4.2.7 Short-cut set-up

The spherical joints of the Delta robot can easily be disassembled. Using the same mechanical parts an RSSR-loop with one degree of freedom can be built (fig. 4.7) [Hunt 78]. The two encoders measure the input and the output angle of such kind of a loop. This way all design parameters of the entire robot with the exception of the end-effector's dimensions can be identified successively without the use of external measurement devices.

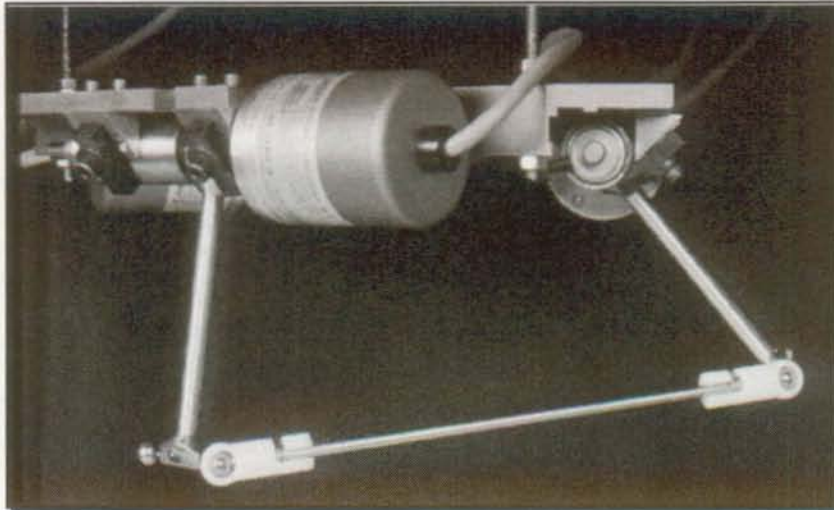


Fig. 4.7: The short-cut set-up

4.3 Set-up for the Argos mock-up

4.3.1 Introduction

The claims for an external full-pose measurement system for the mock-up of the Argos structure contrasts what was claimed for the full-pose measurement system for the Delta mock-up. Small displacements of the position of the virtual center ($< \pm 3$ mm) must be measured, whereas the orientation varies considerably ($< \pm 60^\circ$). The chosen solution is discussed in this section.

4.3.2 Position unit

Three linear digital touch probes TESA-GT130, which are orthogonally arranged to each other (Fig. 4.8), are used to measure the small displacement of the virtual center of the Argos mock-up with the following specifications:

Measurement volume x, y, z	6x6x6 mm
Resolution (limited by the electronic)	0.1 μm
Accuracy (within ± 0.8 mm)	$> \pm 1.9 \mu\text{m}$

The interface to the PC is established via an RS 232-link, which is supported by LabView™.

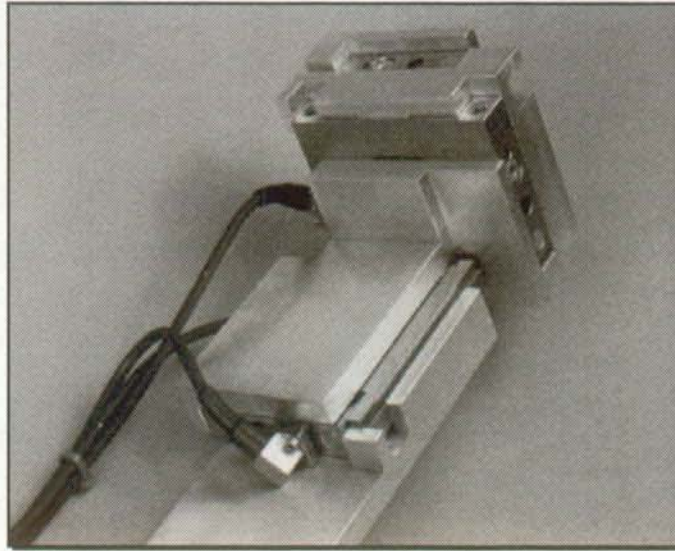


Fig. 4.8: Position unit for small changes

4.3.3 Orientation unit

Figure 4.9 shows three incremental encoders Pewatron-PEH-20 with 2000 physical increments which are orthogonally arranged (gimbal) in order to measure the large orientation changes of the Argos end-effector with the following specifications:

Measurement volume (limitations due to the cables)	$> \pm 70^\circ$ each
Resolution	2.7'
Accuracy	no specification given

Custom-made counter boards based on a HP-HCTL-2016 IC and with a bandwidth up to 12 MHz post-quadrature are used to read dynamically the three incremental encoders. A parallel link to the PC is established by means of a digital data acquisition board (PC-DIO-96) of National Instruments. This effort for dynamic data acquisition was necessary because the Argos mock-up cannot be fixed easily at any measurement point. Thus, the high resolution measurement devices are subjected to fast changes due to the trembling of the operator.

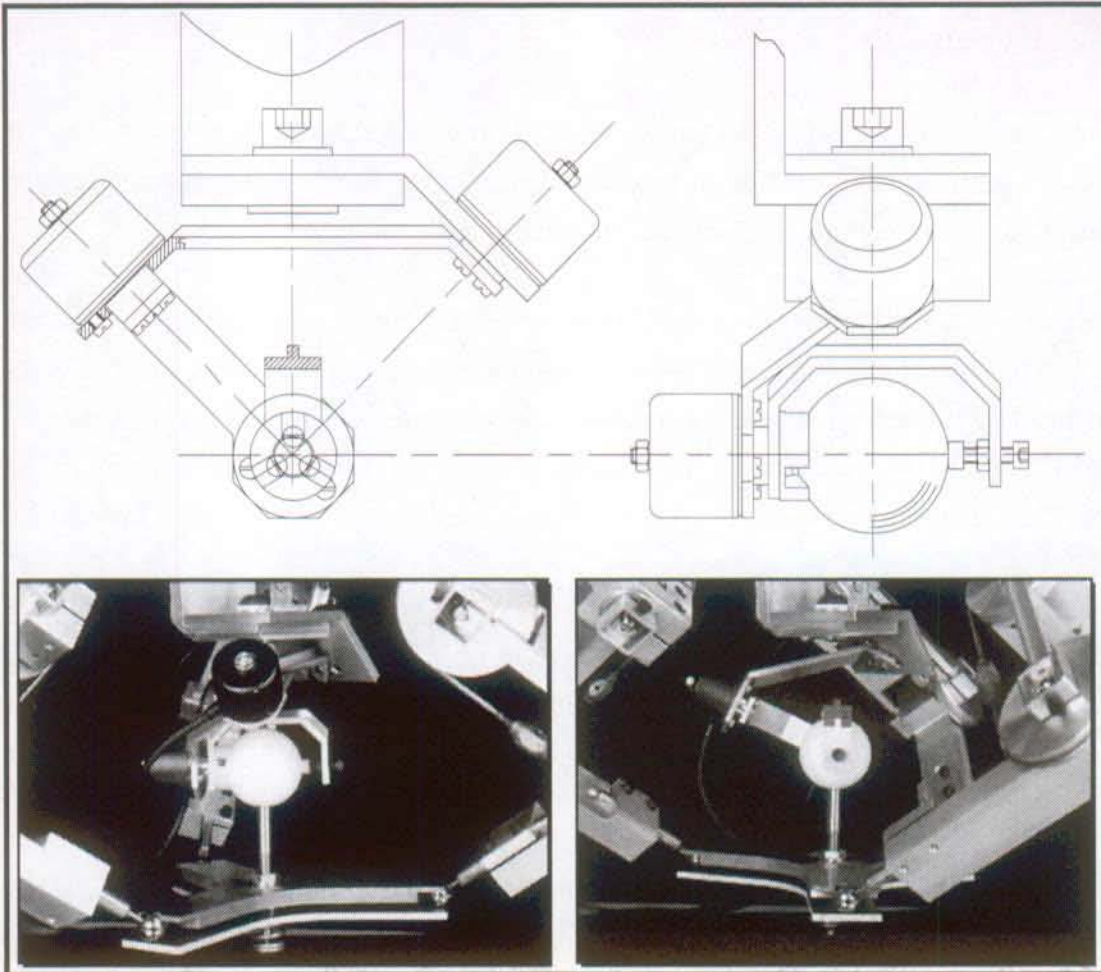


Fig. 4.9: Orientation unit for large changes

4.3.4 Full pose measurement set-up

The measurement device is pre-loaded by a spring in order to compensate its weight.

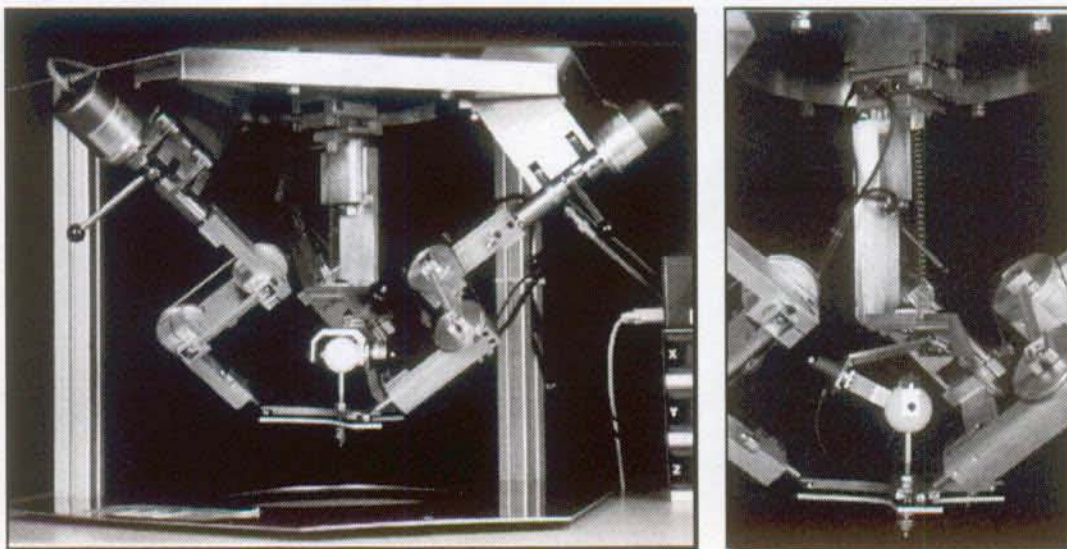


Fig. 4.10: Full pose measurement set-up

4.4 Conclusion

In this section the experimental part of calibration was addressed by entering the vast field of metrology. Four different external measurement units and two simple set-ups using no external measurement device were presented.

Even if the field of measurement devices is extended allowing small contact forces, the claim for cheap and accurate full pose measurement is difficult to reach. The reason is that most of the low-cost and accurate devices measure *only one degree of freedom* (DoF). Thus, intermediate mechanical parts of high precision have to be designed to locate these single DoF-devices with respect to each other, which increases the costs.

The design of two orientation measurement units was much more time-consuming than the design of the translational units. This is not only due to the increasing effort for imagination of orientation matters, but also to the more complicated shape of the intermediate mechanical parts as well as to the smaller number of basic principles for 3D-orientation measurement.

To conclude, the measurement step stays with its difficulty in no way behind the three theoretical steps of calibration (modeling, identification and implementation) and good know-how of this step is essential to develop powerful theoretical tools.

5. Identification

5.1 Introduction

In this chapter the kinematic parameters of the calibration models given in chapter 3 will be determined so that they match the measurement data obtained in chapter 4 most closely. This is referred to as *identification* step. The identification step is the central step in calibration. It deals with the actual storage phase of the robot's individual deviations where adjustable parameters in the calibration model are adapted to fit the real robot behavior as good as possible. A flow chart of the identification step is shown in the figure 5.1:

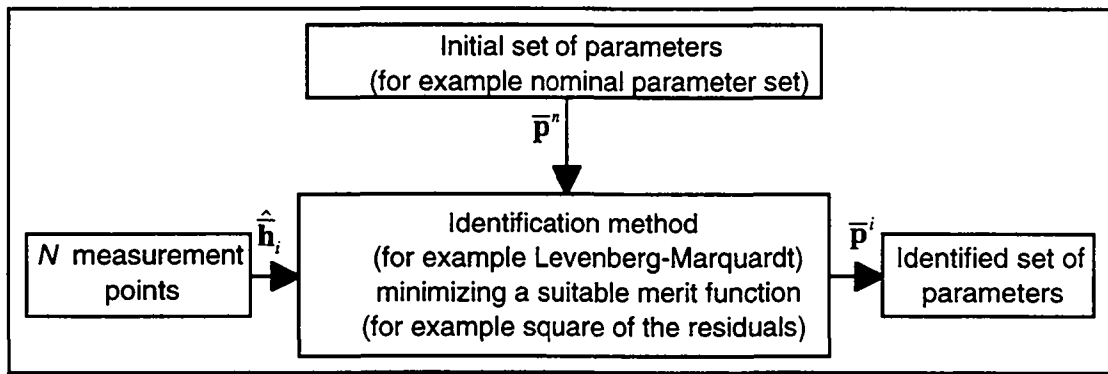


Fig. 5.1: The identification step

Some kinematic parameters as for example the length of the forearm of the Delta robot could be measured directly, whereas others as for example the length of a common normal between two joint axes causes severe problems for accurate direct measurement. Most of the kinematic parameters of the calibration models proposed in chapter 3 cannot be measured directly with a reasonable effort. This is the motivation to apply indirect procedures such as identification to determine the kinematic parameters.

Parameter identification has been studied extensively in the field of numerical mathematics and is widely applied in control theory and dynamic systems modeling. Many of the standard techniques (annex C) for parameter identification [Schwarz 86, Fletcher 87, Press 89] are directly applicable to the manipulator calibration problem [Mooring 91]. Different software libraries containing such standard methods for identification are available on the market. In this work the *optimization toolbox* of

MatLab™ will be used [Moler 92, Grace 92]. *MatLab™* is a software dedicated to fast matrix manipulation based on a library called *LINPACK* [Dongarra 79, Golub 83]. Advantages of the chosen software are the free access to the source code of the optimization toolbox and a high reliability.

In a following section the problem of parameter identification for parallel robots will be stated. In order to show how the standard method for least-squares minimization given in annex C can be adapted to the identification problem of parallel robots, a single-loop structure will be introduced and simulations of different calibration methods presented in a further section.

The rest of the chapter is dedicated to parameter identification of the two chosen examples, the Delta robot and the Argos structure.

5.2 The calibration problem

5.2.1 Introduction

Identification of kinematic parameters of serial structures is generally performed on the direct model, because it can be established very easily by multiplying homogeneous matrices [Kim 87, Mooring 91, Schröder 93]. According to Whitney [86] this is referred to as *forward calibration*.

For parallel robots the model is established in an *implicit form* given by closure equations. Thus, the easiest description of a parallel structure is its implicit model. Identification of parallel robots is therefore preferably done on the implicit model, referred to as *implicit calibration*. This method has recently been applied by Wampler [95].

Calibration using the solution of the inverse problem is referred to as *inverse calibration* [Shamma 87].

The advantages of implicit calibration of parallel robots is that the calibration procedure gets independent for each joint-link train if full pose measurement data are available and if each single parameter to identify appears only in one of the closure equations. The last claim is partially dependent on the type of parameterization chosen.

Taking the implicit model 42 of the Stewart platform (eq. 3.8) as an examples it is easy to see that the identification problem splits into the subproblem of identifying seven

kinematic parameters of a single SPS joint-link train looped back to the base by the measurement loop (fig. 5.2).

A further advantage is, that in most of cases the identification Jacobian (section C.1) of the subproblem can be derivated analytically. For model 42 this results in a $N \times 7$ matrix where N is the number of measurements. How much more complicated the forward calibration of model 42 is can be seen in Wang's work where the direct problem has to be solved numerically many times during identification and where the not analytically available identification Jacobian of the size $6N \times 42$ is established by finite differences [Wang 92].

To conclude, the implicit calibration of a multi-loop structure consists of performing implicit calibration on some sub-structures if full pose measurement data is available. In most of the practical cases, these sub-structures consist of a single loop composed of one joint-link train and the measurement device as shown in figure 5.2.

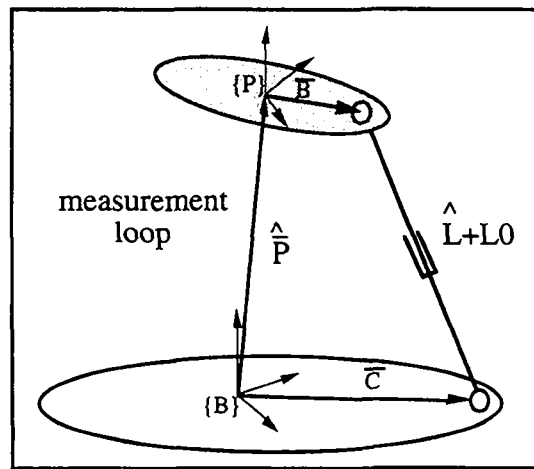


Fig. 5.2: Implicit calibration of a multi-loop structure may be reduced to the implicit calibration of several single-loop structures.

The calibration problem given in figure 5.2 is that of finding the unknown set of kinematic parameters $(\bar{\mathbf{B}}, \bar{\mathbf{C}}, L0)$ for a given set of N full pose measurement points and the corresponding transducer readings $(\hat{L}_j, \hat{\mathbf{P}}_j, \hat{\mathbf{B}}_j)^1$. Both sets will be collected in vectors:

$$\bar{\mathbf{p}} = \{B_x, B_y, B_z, C_x, C_y, C_z, L0\}^T$$

¹ The ^-sign indicates that these values are subject to *measurement errors*.

$$\hat{\mathbf{h}}_j = \{\hat{L}_j, \hat{x}_j, \hat{y}_j, \hat{z}_j, \hat{\alpha}_j, \hat{\beta}_j, \hat{\gamma}_j\} \quad j=1..N \quad (5.1)$$

The implicit model for calibration of the single-loop structure shown in figure 5.2 is given by the substitution of the measurement data (eq. 5.1) into the corresponding equation of model 42 (eq. 3.8):

$$f(\bar{\mathbf{p}}, \hat{\mathbf{h}}_j) = \left(\hat{\mathbf{P}}_j + \hat{\mathbf{R}}_j \cdot \bar{\mathbf{B}} - \bar{\mathbf{C}} \right)^T \cdot \left(\hat{\mathbf{P}}_j + \hat{\mathbf{R}}_j \cdot \bar{\mathbf{B}} - \bar{\mathbf{C}} \right) - (\hat{L}_j + L_0)^2 = 0 \quad j=1..N$$

or in vector notation:

$$\bar{\mathbf{f}}(\bar{\mathbf{p}}) = \bar{\mathbf{0}} \quad (5.2)$$

with the vector $\bar{\mathbf{f}} = \{f_1, f_2, \dots, f_j, \dots, f_N\}^T$

The system of equations 5.2 is non-linearly coupled in the unknown kinematic parameters $\bar{\mathbf{p}}$.

5.2.2 The merit function

According to eq. 5.1 the vector $\bar{\mathbf{p}}$ contains seven unknown kinematic parameters for the chosen example. More generally speaking, the vector $\bar{\mathbf{p}}$ is of dimension n :

$$\bar{\mathbf{p}} \in \mathcal{R}^n \quad (5.3)$$

The system of eqs. 5.2 is determined if the number of unknown parameters (n) is equal to the number of measurements (N) and could therefore exactly be solved for $\bar{\mathbf{p}}$ [Zobel 93]. However, since the measurement vector ($\hat{\mathbf{h}}_j$) is subject to measurement errors the number of measurements should be superior to the number of unknowns [Mooring 91, Schröder 93]:

$$N > n \quad (5.4)$$

The system of eqs. 5.2 becomes thus overdetermined and the problem to solve is that of minimizing the residuals according to a chosen merit function. The residuals are given by:

$$r_j \equiv f(\bar{\mathbf{p}}, \hat{\mathbf{h}}_j) = f_j(\bar{\mathbf{p}}) \quad j=1..N$$

and are collected in a vector:

$$\bar{\mathbf{r}} = \{r_1, r_2, \dots, r_j, \dots, r_N\}^T \equiv \bar{\mathbf{f}}(\bar{\mathbf{p}}) \quad (5.5)$$

According to Mooring [91] or Schröder [93] a least-squares merit function is well adapted to the problem of parameter identification:

$$Q(\bar{\mathbf{p}}) \equiv \bar{\mathbf{r}}^T \cdot \bar{\mathbf{r}} = \bar{\mathbf{f}}(\bar{\mathbf{p}})^T \cdot \bar{\mathbf{f}}(\bar{\mathbf{p}}) \quad (5.6)$$

The problem of solving the system of eqs. 5.2 for $\bar{\mathbf{p}}$ is now converted to the calibration problem of finding a set of parameters $\bar{\mathbf{p}}$ that minimizes the merit function (eq. 5.6).

5.2.3 Unconstrained non-linear least-squares estimation

Taking the square of the Euclidean norm of the residual vector as a merit function simplifies the identification process considerably since the *Hessian matrix* of the merit function is partially composed of first order derivatives (identification Jacobian) of the calibration model. Due to this fact, only *gradient based methods* were reviewed in annex C, which are generally much faster than *direct search methods* where no information on the gradient of the merit function is provided [Press 89].

According to Mooring [91] for most of the practical problems the constraints on the kinematic parameter errors, especially their tolerance field, can be ignored. Therefore, the stated problem is that of *unconstrained non-linear least-squares estimation*.

5.2.4 Conclusion

For the identification problem of parallel robots, two of the methods reviewed in annex C are chosen, due to their *robustness*:

- For weakly non-linear problems and *small residuals* -> Gauss-Newton method based on singular value decomposition (SVD)
- For more non-linear problems and *large residuals* -> Levenberg-Marquardt (LM) method.

When no information about the derivatives is available, the identification Jacobian is replaced by finite difference approximation, which is obtained by a perturbation approach [Wang 92, Grace 92]. The LM-method with integrated finite difference approximation of the identification Jacobian is supported by the *optimization toolbox* of MatLab™ [Grace 92]. In Mathematica™ [Wolfram 91] the LM-method is implemented in a supplementary package called Statistics`NonlinearFit [Mathematica 93].

Whenever possible the implicit calibration model or variations of it will be identified. The only difference compared to forward and inverse calibration is, where in the loop¹

¹Loop built of the measurement device and the structure.

the residuals to be minimized are defined. Simulations have shown that all three methods end up with about the same remaining errors of the end-effector's pose as will be shown in the next section.

5.3 Simulation of the identification step

5.3.1 Introduction

The aim of this section is to compare different calibration methods by simulation. The advantage of simulation is that the original, accurate set of parameters $\bar{\mathbf{p}}^\circ$, which was used to generate the measurement data, is known. The identified set of parameters can therefore be compared to the original one, which allows to get information on the quality of the simulated identification.

As pointed out in the last section, the identification of multi-loop structures using the implicit model may be split into the identification of several single-loop structures if full pose measurement data is available.

This is the main reason why the simulations discussed in this section are based on an example of a single-loop structure. A further advantage of the chosen single-loop structure is that the direct as well as the inverse solution is simple and of closed form, which allows to simulate forward and inverse calibration for comparison with the other calibration methods.

5.3.2 A planar, slider-crank structure

A planar slider-crank mechanism [Denavit 55, Dijkman 76] having one degree of freedom is shown in figure 5.3 .

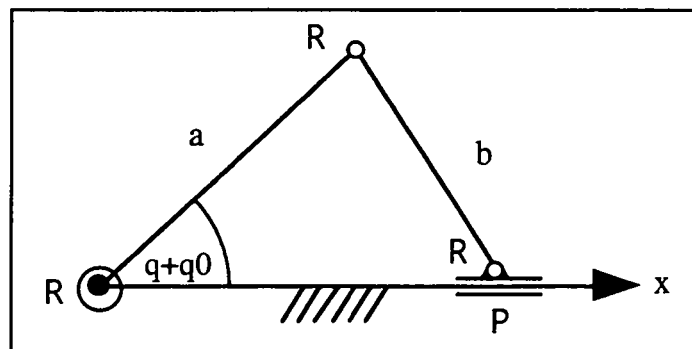


Fig. 5.3: Planar, single-loop slider-crank mechanism

According to eq. 3.4, four independent parameters are required to describe a planar 3RP-loop with no arbitrarily located base and moving frame $\{b\}$ and $\{p\}$. However, assuming that the sensed R-joint is located on the axis of the P-joint and that the last R-joint lies coincident with the P-joint reduces the number of independent parameters to three, which will be collected in a vector according to eq. 5.1:

$$\bar{\mathbf{p}} = \{a, b, q0\}^T \quad (5.7)$$

The following nominal values are assigned to the lengths of the two arms:

$$\begin{aligned} a^n &= 80 \text{ [mm]} & \bar{\mathbf{p}}^n &= \{80, 50, 0\}^T \\ b^n &= 50 \text{ [mm]} \end{aligned} \quad (5.8)$$

The slider of the P-joint is selected as end-effector. A position measurement of the end-effector can be provided by a linear transducer, which measures the displacement of the slider in the x -direction. The two measured values, namely the input angle q and the output displacement x will also be collected in a vector assuming measurements in N different points:

$$\hat{\mathbf{h}}_j = \{\hat{q}_j, \hat{x}_j\} \quad j=1..N \quad (5.9)$$

The implicit model is given by the closure equation over the loop:

$$\boxed{a^2 + x^2 - b^2 - 2*a*x*\cos(q + q0) = 0} \quad (5.10)$$

The solution of the direct problem is:

$$x_{1,2} = a*\cos(q + q0) \pm \sqrt{b^2 - a^2*\sin^2(q + q0)} \quad (5.11)$$

The solution of the inverse problem is:

$$q_{1,2} = \pm \arccos\left(\frac{a^2 + x^2 - b^2}{2*x*a}\right) - q0 \quad (5.12)$$

For the simulation of measurement data the vector $\hat{\mathbf{h}}_j$ is generated by the inverse solution (eq. 5.12) and a Gaussian distributed measurement noise is added according the flow chart of figure 5.4:

¹For identification the dimensional values are normalized by a characteristic length. For the slider-crank structure the first arm length a^n is chosen as characteristic length. However, for clarity dimensional values are kept in the text.

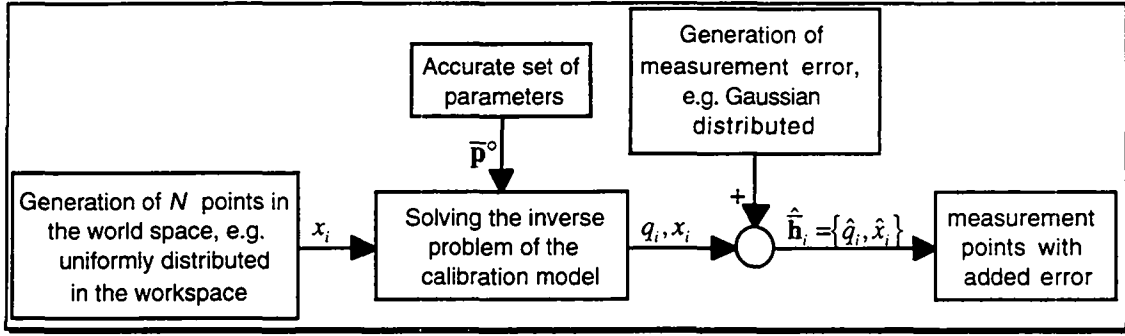


Fig. 5.4: Generation of the measurement points for simulation.

It is of course also possible to use the solution of the direct problem, but using the solution of the inverse problem has the advantage that it is easier to create a uniformly distributed set of measurement points in the world space and that the inverse problem may be easier to solve for multi-loop parallel robots than the direct one (chapter 6).

The following numerical values are assigned to the accurate parameter set \bar{p}^o :

$$\begin{aligned}
 a^o &= a^n + \Delta a = 80 + 0.2 \text{ [mm]} = 80.2 \text{ [mm]} \\
 b^o &= b^n + \Delta b = 50 + 0.1 \text{ [mm]} = 50.1 \text{ [mm]} \\
 q0^o &= 1 \text{ [}^\circ\text{]}
 \end{aligned}
 \quad \bar{p}^o = \{80.2, 50.1, 1\}^T \quad (5.13)$$

The measurement errors are Gaussian distributed with standard deviations¹ of :

$$\begin{aligned}
 \sigma_x &= 0.02 \text{ [mm]} \\
 \sigma_q &= 2/60 \text{ [}^\circ\text{]} = 2 \text{ [']}
 \end{aligned} \quad (5.14)$$

According to the flowchart shown in figure 5.4 a set of 30 measurement points \hat{h}_j was generated shown in figure 5.5:

¹This statistical property has to be chosen realistically. A calibration method may work well when simulated without measurement noise but may fail when applied to measurement data (e.g. "cascaded calibration" paragraph 5.3.7). An angular measurement error of 10 arcseconds introduced by Lin [89] or an assumed measurement error of 1 micrometer per 100 millimeters [Innocenti 95] is difficult to reach in practice as shown in chapter 4. In reality much less is known about the quality of the measurement device and its improvements is very time consuming and expensive.

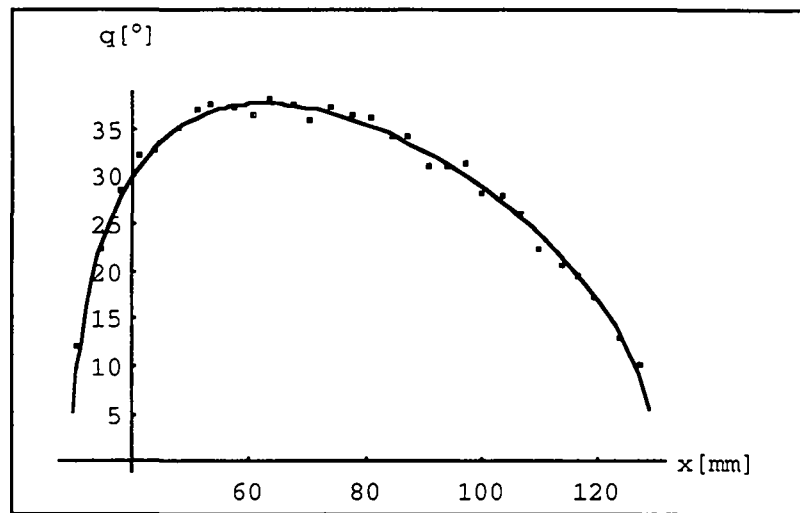


Fig. 5.5: Set of 30 measurement points (measurement errors amplified by a factor of 20 for visibility). The graph shows the calibration model with the accurate parameters $\bar{\mathbf{p}}^0$.

Since the parameterization is chosen with care and no dependent measurement points are introduced, the slider-crank mechanism doesn't represent a *pathological case* for identification. In other words, any of the *non-linear least square estimation* techniques reviewed in annex C works well for this example. Therefore, the *question of robustness* of the different least square estimation techniques can't be tested using the proposed example. For more detailed information about robustness the reader is referred to Fletcher [87] and Press [89].

Now, the measurement data are prepared and the different calibration method can be tested, which is subject of the remaining paragraphs of this section.

5.3.3 Implicit calibration

The first and to the author's opinion most important calibration method for parallel robots is implicit calibration. It is given by substitution of the measurement data shown in figure 5.5 into the implicit model (eq. 5.10):

$$f(\bar{\mathbf{p}}, \hat{\mathbf{h}}_j) = f_j(\bar{\mathbf{p}}) = a^2 + \hat{x}_j^2 - b^2 - 2*a*\hat{x}_j*\cos(\hat{q}_j + q_0) = 0 \quad j=1..N \quad (5.15)$$

For least-squares estimation techniques for parameter identification one element of the residual vector $\bar{\mathbf{r}}$ (eq. 5.5) for the iteration step k can be calculated as:

$$r_j \equiv f_j = a_k^2 + \hat{x}_j^2 - b_k^2 - 2*a_k*\hat{x}_j*\cos(\hat{q}_j + q_{0k}) \quad j=1..N \quad (5.16)$$

One row of the identification Jacobian $\bar{\mathbf{J}}$ (eq. C.5) can be written as:

$$\bar{\mathbf{J}}_j = \{2*a_k - 2*\hat{x}_j*\cos(\hat{q}_j + q_{0k}), \quad -2*b_k, \quad 2*a_k*\hat{x}_j*\sin(\hat{q}_j + q_{0k})\} \quad j=1..N \quad (5.17)$$

Finally, one layer of the identification Hessian $\bar{\mathbf{H}}$ (eq. C.8) is given by:

$$\bar{\mathbf{H}}_j = \begin{bmatrix} 2 & 0 & 2*\hat{x}_j*\sin(\hat{q}_j + q_{0k}) \\ 0 & -2 & 0 \\ 2*\hat{x}_j*\sin(\hat{q}_j + q_{0k}) & 0 & 2*a_k*\hat{x}_j*\cos(\hat{q}_j + q_{0k}) \end{bmatrix} \quad j=1..N \quad (5.18)$$

The identification of the parameter set $\bar{\mathbf{p}}^i$ to fit the measurement data is performed according to the flow chart of figure 5.6:

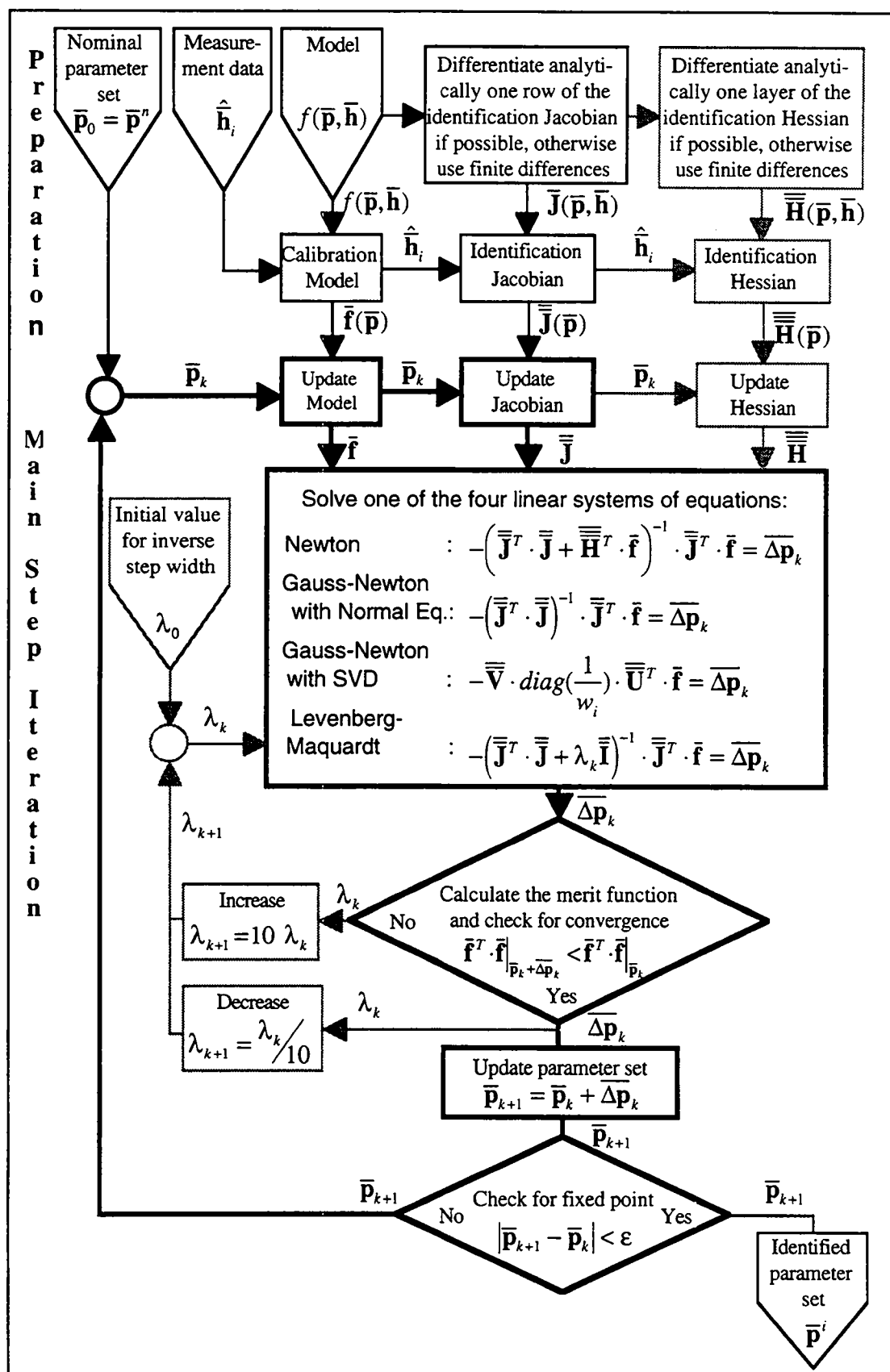


Fig. 5.6: Flow chart of parameter identification with optional use of one of the four methods for least-squares estimation reviewed in annex C.

All of the four different methods converge to the same set of parameters, but require a different number of iterations and therefore a varying calculation time:

$$\bar{\mathbf{p}}' = \{80.196 \text{ [mm]} \quad 50.107 \text{ [mm]} \quad 1.010 \text{ [}^\circ\text{]}\}^T \quad (5.19)$$

Table 5.7 shows the number of iterations and the calculation times of the different methods given in annex C and figure 5.6. The simulations were done using Mathematica™ on a Macintosh having a Motorola 68030-processor with a mathematical 68882-coprocessor. The calculation time depends strongly on the use of build-in functions or not.

Method	Number of iterations	Calculation time [s]	function
Newton	4	59	programmed
Gauss-Newton based on Normal Equations	3	26	programmed
Gauss-Newton based on singular value decomposition (SVD)	3	7	programmed / build-in (SVD)
Levenberg-Marquardt (LM)	3	18	built-in
Steepest Decent	72	57	built-in

Table. 5.7: Number of iterations and calculation time of the different methods for parameter identification (fig. 5.6) of the slider-crank mechanism.

From table 5.7 it can be seen that including the identification Hessian isn't advantageous for the chosen examples (Newton method) since more iterations and more calculation time is needed than for the Gauss-Newton method. This is somewhat contradictory to the expected quadratic convergence rate of Newton's method as compared to the only linear convergence rate of Gauss-Newton method. This may be due to the weak non-linearity of the treated problem. Newton's method will not be further used in this work.

The difference in the calculation time between the Gauss-Newton method using normal equations and singular value decomposition (SVD) can only be explained by the use of built-in functions in the case of SVD. Their iteration steps are identical, of course. The Gauss-Newton method is the fastest method requiring the lowest number of iterations.

The Levenberg-Marquardt (LM) method lies as expected between the Gauss-Newton and the steepest decent method. The poor convergence rate of the steepest descent method is well known in numerical mathematics [Press 89]. This method is therefore also abandoned.

Table 5.7 shows also a clear difference in calculation time between built-in and self-programmed functions, which supports the decision to use standard software libraries such as the optimization toolbox of MatLab™.

It remains to discuss the improvement of accuracy gained by calibration. This will be done in two parts:

The results will firstly be qualified using the nominal parameter set $\bar{\mathbf{p}}^n$ and the identified parameter set $\bar{\mathbf{p}}^i$ as in the experimental calibration carried out in section 5.4 and 5.5.

In the second part, the identified parameter set $\bar{\mathbf{p}}^i$ will be compared to the accurate parameter set $\bar{\mathbf{p}}^o$, which had been used to generate the measurement data for simulation (fig. 5.4).

Figure 5.8 shows the residuals defined in eq. 5.16 plotted versus the measurement points before and after calibration:

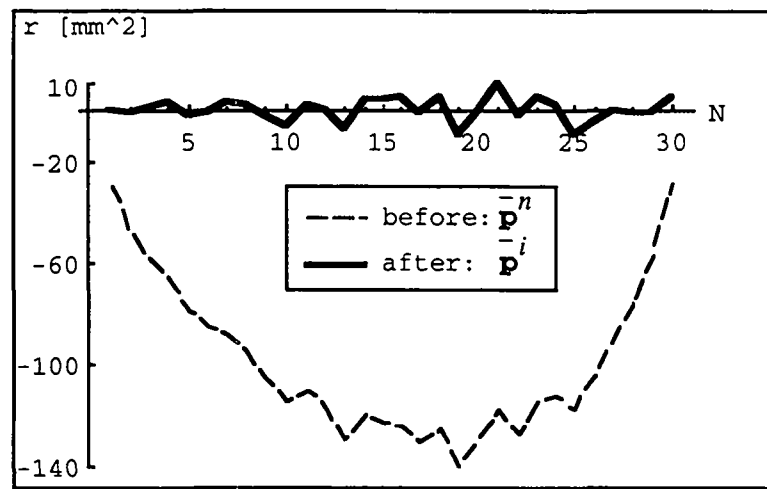


Fig 5.8: Residuals of the implicit model before and after calibration. The merit function Q defined as $\bar{\mathbf{F}}^T \cdot \bar{\mathbf{F}}$ decreases from 319'829 to 614 [mm⁴].

The residuals of the implicit model are rather abstract. Thus, table 5.9 shows on the left hand side how much the accuracy of the end-effector (slider) position was actually improved by calibration. This may be important to know when controlling a milling center or other machine-tool of parallel topology. On the right hand side of the table the error of the encoder readings is shown. It may be required for a measurement machine with a parallel structure if the end-effector is brought into contact with the piece to be measured. In addition different factors of improvement are defined for quantification of the gain of accuracy by calibration.

Position error	Joint error
<p>parameter set or $\bar{\mathbf{p}}^n$ or $\bar{\mathbf{p}}^i$</p> <p>measurement data $\hat{\mathbf{h}}_i$</p> <p>Joint space measurement data $\hat{\mathbf{q}}_i$</p> <p>World space measurement data $\hat{\mathbf{x}}_i$</p> <p>Direct solution of the implicit model $\bar{\mathbf{x}}_i$</p> <p>remaining position error $\Delta\bar{\mathbf{x}}_i$</p>	<p>parameter set or $\bar{\mathbf{p}}^n$ or $\bar{\mathbf{p}}^i$</p> <p>measurement data $\hat{\mathbf{h}}_i$</p> <p>World space measurement data $\hat{\mathbf{x}}_i$</p> <p>Joint space measurement data $\hat{\mathbf{q}}_i$</p> <p>Inverse solution of the implicit model $\bar{\mathbf{q}}_i$</p> <p>remaining joint error $\Delta\bar{\mathbf{q}}_i$</p>
<p>$\Delta\mathbf{x}$ [mm]</p> <p>$\bar{\mathbf{p}}^n$: before calibration -> dashed line $\bar{\mathbf{p}}^i$: after calibration -> bold line</p>	<p>$\Delta\mathbf{q}$ [°]</p>
<p>before calibration:</p> <p>mean: -1 [mm]</p> <p>deviation: 3.7 [mm]</p>	<p>before calibration:</p> <p>mean: -1 [°]</p> <p>deviation: 0.2 [°]</p>
<p>after calibration:</p> <p>mean: 0.08 [mm]</p> <p>deviation: 0.34 [mm]</p>	<p>after calibration:</p> <p>mean: -0.0002 [°]</p> <p>deviation: 0.04 [°]</p>
<p>Improvement factor of the position</p> $F^{Pos} = \frac{\sum_{i=1}^N \Delta\bar{\mathbf{x}}_i(\bar{\mathbf{p}}^n) }{\sum_{i=1}^N \Delta\bar{\mathbf{x}}_i(\bar{\mathbf{p}}^i) } = 17.9$	<p>Improvement factor of the joint angles</p> $F^{Joint} = \frac{\sum_{i=1}^N \Delta\bar{\mathbf{q}}_i(\bar{\mathbf{p}}^n) }{\sum_{i=1}^N \Delta\bar{\mathbf{q}}_i(\bar{\mathbf{p}}^i) } = 29.3$

Table 5.9: Summary of the results of the calibration performed on the implicit model.

If the orientation of the end-effector is included in the calibration model a third factor of improvement must be defined:

$$F^{Ori} = \frac{\sum_{i=1}^N |\Delta \bar{e}_i(\bar{p}^n)|}{\sum_{i=1}^N |\Delta \bar{e}_i(\bar{p}^i)|} \quad (5.20)$$

The right hand side of table 5.9 shows that the position error of the slider becomes very high when the mechanism is in its direct singularity (bifurcation of two solutions -> gain of one degree of freedom). For that reason close approach of singularities must be avoided during the data acquisition phase (chapter 4).

The identified set \bar{p}^i (eq. 5.19) used to generate the measurement data will now be compared with the accurate set \bar{p}^o (eq. 5.13). Their difference is given by:

$$\bar{p}^o - \bar{p}^i = \{4 [\mu\text{m}] \quad -7 [\mu\text{m}] \quad -35 [^\circ]\}^T \quad (5.21)$$

In figure 5.10 the difference between the inverse solution of the accurate parameter set and the identified parameter set is plotted over the workspace ($\Delta q = q(\bar{p}^o, x) - q(\bar{p}^i, x)$):

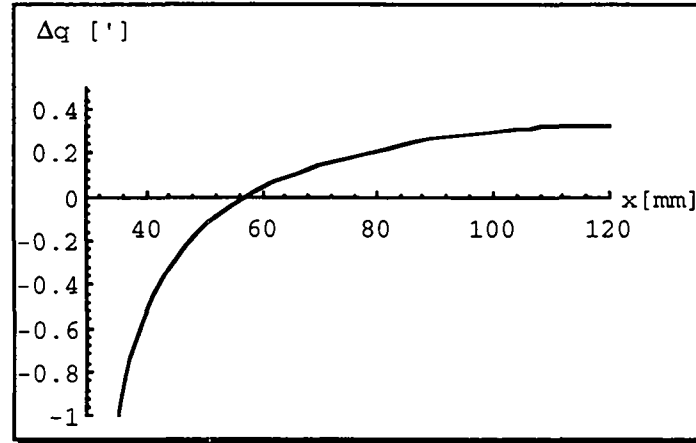


Fig. 5.10: Plot over the work space of the difference between the inverse solutions of \bar{p}^o and \bar{p}^i .

Figure 5.10 as well as eq. 5.21 show that the identified parameter set \bar{p}^i is very close to the original data set \bar{p}^o . Implicit calibration performs well for this kind of problem in spite of the measurement errors introduced (eq. 5.14).

5.3.4 Forward calibration

The term "*forward calibration*" was introduced by Whitney [86] for calibration of serial robots. Forward calibration uses the direct solution (eq. 5.11) of the model. Measurement data are thereby substituted into the direct solution. The residuals of the minimization problem are given by:

$$r_j = a_k * \cos(\hat{q}_j + q0_k) \pm \sqrt{b_k^2 - a_k^2 * \sin(\hat{q}_j + q0_k)^2} - \hat{x}_{1,2,j} \quad j=1..N \quad (5.22)$$

Forward calibration (eq. 5.22) was performed using Gauss-Newton method based on singular value decomposition (SVD). To avoid diverging of the algorithm measurement points (point 8-13 in table 5.9) closest to the direct singularity [Gosselin 88] at point $\{38.7^\circ, 62.4 \text{ mm}\}$ were not used for identification. The following set of parameters was identified within 19 seconds and four iterations:

$$\bar{\mathbf{p}}^i = \{80.166 [\text{mm}] \quad 50.060 [\text{mm}] \quad 0.961 [^\circ]\}^T \quad j=1..N \quad (5.23)$$

The two factors of improvement as defined in table 5.9 are:

$$F^{Pos} = 18.4 \quad F^{Joint} = 28.4 \quad (5.24)$$

Comparison with table 5.9 shows that the F^{Pos} -factor is slightly superior to the same factor calculated for implicit calibration. Such small differences are not significant. It can therefore be concluded that both methods work about equally well.

However, forward calibration is not well suited for the calibration of parallel robots due to the following reasons:

- The direct problem of parallel robots has always *several solutions* (e.g. eq. 5.22). The bifurcation points of the different branches of solutions are located at direct singularities (paragraph 6.2.4). For forward calibration the solution which corresponds to the real, measured robot configuration must be selected. This *selection problem* may become severe during identification when the parameters of the model are varied, what changes also the bifurcation points of direct solutions. A measurement point which was originally taken at a certain distance away from a direct singularity of the nominal model may get close to this wandering bifurcation during identification forcing the identification algorithm to fail (as could be seen in this paragraph). Forward calibration has therefore two problems to face when approaching a direct singularity: First, as for all other methods, measurement values are subjected to large errors due to physical breakdown of the structure. Second, the

right solution has to be chosen which is difficult closed to the mathematical branching of the direct solution (mathematical breakdown).

- Comparison of eq. 5.22 with the corresponding eq. 5.15 for implicit calibration shows a more complicated mathematical structure for forward calibration, which increases the *complexity* of the *identification Jacobian*. Unlike the identification Jacobian of implicit calibration, the identification Jacobian for forward calibration of multi-loop structures can generally *not* be supported analytically due to its enormous complexity .
- As shown in chapter 6, the solution of the *direct problem* of multi-loop structures can be already *very complicated*. Due to the fact that a system of closure equations representing a multi-loop structure is *coupled in the unknown world coordinates*, the analytical solution -if its exists- will contain all kinematic parameters in each of its equations. The forward calibration¹ can therefore *not be split* into subproblems as in the case of implicit calibration.

Forward calibration was treated in this paragraph in order to shown that this *standard method* for the calibration of *serial robots* is not suitable for parallel structures.

5.3.5 Inverse calibration

Inverse calibration is performed on the inverse solution of the kinematic model (eq. 5.12), where the residuals are given by:

$$r_j = \pm \arccos\left(\frac{a_k^2 + \hat{x}_j^2 - b_k^2}{2 * \hat{x}_i * a_k}\right) - q_0 - \hat{q}_{1,2,j} \quad j=1..N \quad (5.25)$$

Again, the problem of multiple solutions arises. However, it may be more easily to overcome, since the bifurcation of solutions takes place at inverse singularities (paragraph 6.2.4), which is generally located at the border of the workspace. The identification of the parameters using Gauss-Newton method based on singular value decomposition leads within three iterations and 13 seconds of calculation time to the following set of identified parameters:

$$\bar{\mathbf{p}}^i = \{80.208 \text{ [mm]} \quad 50.116 \text{ [mm]} \quad 1.015 \text{ [}^\circ\text{]}\}^T \quad (5.26)$$

¹Simulations of forward calibration of a Stewart Platform were made by Wang [92], showing the enormous complexity of applying this method to multi-loop structures.

Calculation of the two factors of improvement defined in table 5.9 yields:

$$F^{Pos} = 17.1 \quad F^{Joint} = 29.3 \quad (5.27)$$

The F^{Joint} -factor is the same as received by implicit calibration whereas the F^{Pos} -factor is slightly below the one given in table 5.9. However, such slight differences may again not be trusted and it can be said that both of the methods work with equal quality on such kind of problem. The solution of the inverse problem is generally simple for parallel robots if the model is not too complicated (i.e.: simple for model 24 and very complicated for model 54).

5.3.6 Linear calibration

Linear calibration is based on a linearization of the implicit model at the nominal parameter set. For the slider-crank mechanism the following parameter substitution may be introduced:

$$a = a^n + \Delta a, \quad b = b^n + \Delta b, \quad q_0 = \Delta q_0 \quad (5.28)$$

Substitution into the implicit model (eq. 5.10) and neglecting quadratic terms yields:

$$\begin{Bmatrix} 2a^n - 2\hat{x}_j \cos(\hat{q}_j) \\ -2b^n \\ 2\hat{x}_j a^n \sin(\hat{q}_j) \end{Bmatrix}^T \cdot \begin{Bmatrix} \Delta a \\ \Delta b \\ \Delta q_0 \end{Bmatrix} = (b^n)^2 - (a^n)^2 - (\hat{x}_j)^2 + 2\hat{x}_j a^n \cos(\hat{q}_j) \quad j=1..N$$

or in matrix notation:

$$\bar{\mathbf{J}}|_{\bar{\mathbf{p}}^n} \cdot \Delta \bar{\mathbf{p}} = -\bar{\mathbf{f}}|_{\bar{\mathbf{p}}^n} \quad (5.29)$$

This is an overdetermined linear system of equations, which can be solved with singular value decomposition (SVD) leading to the following identified parameter set:

$$\bar{\mathbf{p}}^i = \{80.184 \text{ [mm]} \quad 50.087 \text{ [mm]} \quad 1.012 \text{ [}^\circ\text{]}\}^T \quad (5.30)$$

The two factors of improvement (table 5.9) are:

$$F^{Pos} = 12.06 \quad F^{Joint} = 26.17 \quad (5.31)$$

Both factors are below the factors of implicit calibration given in table 5.9. This is not further astonishing: Comparison of the two calibration methods given in figure 5.6 and eq. 5.29 shows that the first step of iteration of the implicit calibration is identical to the linear calibration. In other words, if the iterative algorithm of the implicit calibration is converging, than the identified parameter set will always be of better quality than the one found by linear calibration.

5.3.7 Cascaded calibration

Cascaded calibration of parallel robots was proposed by Zhuang [91]. The method may be explained as the splitting of the implicit model into sub-models, which are as linear as possible in the unknown parameters. The creation of this sub-models is based on manipulations of the N error equations, which may be added to or subtracted from one another. Therefore it may be useful to acquire special measurement sets, where some joint or world coordinates are hold constant. This will make terms depending on these coordinates vanishing by subtraction of two error equations of this special measurement set. Zhuang for example eliminated the quadratically involved transducer offset by collecting measurement data of the Stewart Platform where the corresponding leg was kept at a constant length. Further work has been done using this concept [Geng 94, Innocenti 95].

Cascaded calibration is applied here to the slider-crank mechanisms. The implicit model (5.10) is first rewritten as linearly as possible with respect to its kinematic parameters. Applying trigonometric transformations yields:

$$a_x^2 + a_y^2 - b^2 + \hat{x}_j^2 - 2\hat{x}_j a_x \cos(\hat{q}_j) - 2\hat{x}_j a_y \sin(\hat{q}_j) = r_j \quad j=1..N \quad (5.32)$$

with

$$a_x = a \cos(q_0), \quad a_y = -a \sin(q_0)$$

The power of two terms of the unknown parameters $\{a_x, a_y, b\}^T$ appearing in eq. 5.32 are isolated from the measurement data. Any subtraction of two error equations will therefore be linear with respect to the remaining unknowns. Thus, $N-1$ error equations will be subtracted from the last one:

$$\hat{x}_N^2 - \hat{x}_j^2 - 2(\hat{x}_N - \hat{x}_j)a_x \cos(\hat{q}_j) - 2(\hat{x}_N - \hat{x}_j)a_y \sin(\hat{q}_j) = (r_N - r_j) \quad j=1..N-1 \quad (5.33)$$

This is the first linear sub-model used for the *first step* of the cascaded calibration in order to identify linearly the two unknown parameters $a_x, a_y \rightarrow a_x^i, a_y^i$: Once these two parameters are identified, they can be back-substituted into eq. 5.32. The latter can be linearly solved for the remaining unknown b^2 , which leads to the *second step* of cascaded calibration:

$$a_x^{i2} + a_y^{i2} - b^2 + \hat{x}_j^2 - 2\hat{x}_j a_x^i \cos(\hat{q}_j) - 2\hat{x}_j a_y^i \sin(\hat{q}_j) = r_j \quad j=1..N \quad (5.34)$$

Finally, the identified parameter set can be calculated as:

$$a^i = \sqrt{(a_x^i)^2 + (a_y^i)^2}, \quad b^i = \sqrt{(b^i)^2}, \quad q\theta^i = \arctan 2 \left(\frac{-a_y^i}{a^i}, \frac{a_x^i}{a^i} \right) \quad (5.35)$$

Note that cascaded calibration doesn't need an initial guess of the parameter set if all sub-models are linear.

Cascaded calibration leads within 0.8 seconds to the following set of identified parameters using singular value decomposition (SVD) in order to solve the two over-determined sub-models given by eq. 5.33 and eq. 5.34:

$$\bar{\mathbf{p}}^i = \{80.224 \text{ [mm]} \quad 50.156 \text{ [mm]} \quad 1.050 \text{ [}^\circ\text{]}\}^T \quad (5.36)$$

where the two factors of improvement (table 5.9) are:

$$F^{Pos} = 12.12 \quad F^{Joint} = 26.16 \quad (5.37)$$

The improvement is about the same as reached by linear calibration (eq. 5.31), but below the improvement gained by implicit calibration (table 5.9). It should be pointed out that without measurement noise cascaded calibration finds the correct parameter set. However, it seems less suitable for identification if there's measurement noise.

5.3.8 Semiparametric calibration

Semiparametric calibration is situated between parametric and nonparametric modeling as discussed in section 3.1. It is a modification of a calibration concept for serial robots proposed by Sayeh [94] and independently by Flury [94] based on a work of LeeS [91].

The concept is to separate joint and world coordinates of the kinematic parameters in the implicit model by means of expanding trigonometric functions into sums of trigonometric functions. Once separated as many *independent linear factors* are introduced as combinations of joint and/or world coordinates exist plus an additional linear factor for the constant part in order to replace the original kinematic parameters. This is more than a parameter substitution since in general more independent factors have to be introduced than kinematic parameters were required. Taking for examples model 24 of the Delta robot 36 independent linear factors are needed. Thus, a *part of the geometric constraints* of the underlying system were *skipped* and replaced by independent factors, which corresponds to a replacement of the parametric model by a *semiparametric model*.

As indicated the starting point is the implicit parametric model given by eq. 5.10:

$$a^2 + x^2 - b^2 - 2*a*x*\cos(q + q0) = 0 \quad (5.38)$$

The factor, where joint as well as world coordinates are not yet separated from the kinematic parameters, is the last one on the left hand side. Thus, this trigonometric function will be written as a sum of trigonometric functions:

$$a^2 + x^2 - b^2 - 2*a*x*\cos(q) * \cos(q0) + 2*a*x*\sin(q) * \sin(q0) = 0 \quad (5.39)$$

which corresponds to eq. 5.32 in the cascaded calibration.

Coordinates and kinematic parameters are now separated. Two combinations of joint and world coordinates exist. Thus, three linearly independent factors (a_i) are required in order to establish the semiparametric model for the slider crank mechanism

$$a_1 + a_2 * x * \cos(q) + a_3 * x * \sin(q) + x^2 = 0 \quad (5.40)$$

Substitution of the N measurement points (fig. 5.5) yields the model for semi-parametric calibration:

$$a_1 + a_2 * \hat{x}_j * \cos(\hat{q}_j) + a_3 * \hat{x}_j * \sin(\hat{q}_j) + \hat{x}_j^2 = 0 \quad j=1..N \quad (5.41)$$

Semiparametric calibration leads after 0.2 seconds to the following set of identified linear factors (a_i) using singular value decomposition (SVD) to solve the over-determined system of equations 5.41:

$$a_1^i = 3921 \text{ [mm}^2\text{]}, a_2^i = -160 \text{ [mm]}, a_3^i = -2.83 \text{ [mm]} \quad (5.42)$$

Substitution of eq. 5.42 into the semiparametric model (eq. 5.40) and solving the direct and inverse problem leads to the two improvement factors as defined in table 5.9:

$$F^{Pos} = 17.9 \quad F^{Joint} = 29.3 \quad (5.43)$$

Comparison with implicit calibration shows that the gained improvement is exactly the same (table 5.9). For the slider crank example the number of kinematic parameters is equal to the numbers of linear, independent factors needed for the semiparametric model. This allows *exceptionally* to extract the kinematic parameters from the identified linear factors¹, which shows in fact that both parameter sets are the same.

¹ $a_1 = a^2 - b^2, a_2 = -2*a*\cos(q0), a_3 = 2*a*\sin(q0)$

Semiparametric calibration needs no initial guess and is about *35 times faster* than implicit calibration leading to the same factors of improvement. It is decoupled for each joint-link train, as all methods which are based on the implicit model. Major drawback is that generally the *kinematic parameters cannot be extracted* from the independent, linear factors¹. This cancels the advantage of parametric modeling, allowing a quality control of the robot's mechanical parts by calibration. This gets particularly important for the two chosen practical examples, the Delta and the Argos structure, where the orientation of the end-effector and the position of the virtual center cannot be influenced otherwise than by exchanging mechanical parts exceeding tolerances. This was the reason for semiparametric calibration not using.

5.3.9 Optimal selection of measurement set

For the slider-crank mechanisms the 30 measurement points were equally distributed over the whole workspace. All measurement points together build a measurement set. In praxis, each additional measurement point increases the costs of the calibration. In order to minimize the number of measurement points for a reliable calibration, points have to be found in the workspace where small variations of some of the kinematic parameters can be well *observed*. In other words, such a point must be *sensitive* to changes of these kinematic parameters. However, minimizing the calibration cost is not the only argument for optimizing the measurement set. Bad measurement points (e.g. close to a singularity) can even force the identification algorithm to diverge as could be seen in the example of forward calibration in paragraph 5.3.4.

The observability of a kinematic parameter varies in the workspace and each parameter has its maximal observability at a different point. The observability of a kinematic parameter does not only depend on the location in the workspace, it also depends on all the other parameters. The manifold of possible sets of measurement points makes the optimal choice to an *optimization problem of large scale* [Press 89].

¹The analytical relation between kinematic parameters and independent, linear factors is violated by the introduction of the factors, which are claimed to be independent. Solution of the resulting overdetermined system of equations in a least square sense generally yields a badly fitting kinematic parameter set. The parametric model 42 of the Stewart platform for examples would be converted to a semiparametric model having 174 independent linear factors, which corresponds to a 132 times overdetermined system of equations to extract the 42 kinematic parameters from.

Apart from numerical search of the optimal measurement set some simple rules may be established such as:

- Stay away of singularities, particularly of direct ones (paragraph 5.3.4).
- Avoid approaching the border of the workspace too closely (inverse singularities).
- Choose points in the entire volume where the calibration should be valid.
- If the parallel robot is symmetrical, the acquisition should be performed symmetrically, too (equal treatment of each joint link train).

This approach is used for the experimental part of this chapter. However, numerical techniques exist to find an optimal measurement set. One approach is based on checking of the condition number (κ) of the identification Jacobian, which depends on the measurement set [Mooring 91]. The condition number was defined in section C.3 as the ratio of the highest and the lowest singular value. The bigger the κ -number the more ill-conditioned is the identification Jacobian. As already mentioned then, there are two reasons for an ill-conditioned identification Jacobian. The model may contain redundant parameters or the measurement points are badly chosen. If the model is for sure not redundant¹ the κ -number can be used to search for the optimal measurement set by varying the measurement points and calculating the resulting κ -number. The set with the lowest κ -number will be the optimal one. An similar approach based on an observability measurement where all singular values are taken into account is given in Borm [91]. A *noise amplification index* was defined by Nahvi [96] as ratio of the lowest singular value to the κ -number.

Another numerical approach was proposed by Zhuang [94] using *simulated annealing* (SA). According to Press [89] the SA-method allows to find the optimum of large scale problems and is therefore well suited for the search of optimal set. It is based on the stochastic simulation of a cooling down process, where the molecules loose more and more energy and the probability for an up-hill step gets smaller and smaller, but it remains possible. SA-method is therefore capable to escape from a local minimum once trapped in it. According to Zhuang SA-method works well on the problem of optimal selection of the measurement points.

Investigations on the observability of model 54 of the Delta robot have shown that 12 parameters (set P2 given in eq. 3.15) describing the size of the end-effector and the

¹ This should be guaranteed by the eq. 3.3 proposed in section 3.2.1

distance between the forearms are nearly unobservable in the *entire* workspace. Searching for possibilities to identify these 12 parameters, one of the forearms could be replaced by a much longer one (e. g. 10 mm). In that case, the orientation of the end-effector changes reasonably within the workspace and the 12 parameters become observable. Another possibility is to change for calibration the configuration space shown in paragraph 4.2.5. Both methods need a measurement device for *large changes* in orientation (paragraph 4.3.3), which is difficult to support and therefore a drawback.

However, as pointed out earlier, *small observability is caused by small sensitivity*. Small changes of the tolerance field of these 12 parameters nearly don't affect the pose of the end-effector within the entire workspace. For that reason they are set to the nominal values during identification. Only the remaining 42 *parameters* have thus to be *identified*.

5.3.10 Conclusion

In table 5.11, the results are summarized from implicit, forward, inverse, linear, cascaded, and semiparametric calibration of the slider-crank mechanism:

Calibration	Identified parameters			Difference to the accurate parameter set			Quality of improvement Suitable for quality control				Speed		Comment
Variable	$\bar{\mathbf{p}}^i$			$\bar{\mathbf{p}}^o - \bar{\mathbf{p}}^i$			$Q(\bar{\mathbf{p}}^i)$	F^{Pos}	F^{Joint}	-	N_{itr}	t_{cal}	
Units	[mm]		[°]	[μm]		["]	[mm ⁴]	-	-	-	-	[s]	
$\bar{\mathbf{p}}^o$	80.200	50.100	1.00										
Implicit	80.196	50.107	1.01	4	-7	-35	614	17.9	29.3	y	3	7	-
Forward	80.166	50.060	0.96	34	40	-141	745	18.4	28.4	y	4	19	Difficult in case of multi-loop
Inverse	80.208	50.116	1.02	-8	-15	-54	625	17.1	29.3	y	3	13	-
Linear	80.184	50.087	1.01	16	13	-43	749	12.1	26.2	y	0	0.2	First step of implicit calibration
Cascaded	80.224	50.146	1.05	-24	-56	-181	688	12.1	26.2	y	0/0	0.8	No initial guess needed
Semiparametric	80.196	50.107	1.01	4	-7	-35	614	17.9	29.3	n	0	0.2	No initial guess needed

Table 5.11: Summary of results from different calibration methods. Bold face letters give the best values.

Table 5.11 shows that implicit calibration is a good compromise as *standard method* for the *calibration problem of parallel robots*. It is not only one of the best method in terms of gained improvement, but it has also identified the parameter set closest to the accurate parameter set. This is important since the identified parameters should not only fit the measurement data as well as possible (*local validity*), but it should also be valid for other measurement data generated by the accurate parameter set (*global validity*). Furthermore, the identification problem may be split into subproblems and the identification Jacobian differentiated analytically. Also, a quality control is possible, which is especially important when the robot has less than six degrees of freedom.

***Implicit calibration is proposed as standard method
for the calibration problem of parallel robots.***

The factors of improvement in table 5.11 represent the upper boundary for gain of accuracy by calibration. Even if a realistic variance of measurement noise is introduced the distribution of the measurement noise is perfectly Gaussian. In reality however, the measurement points will be subject to outliers and nothing is known about the distribution of the measurement noise.

5.4 Parameter identification for the Delta robot

5.4.1 Introduction

The nominal parameters $\bar{\mathbf{p}}^n$ of the mock-up built of the Delta robot (section 2.3) are given in table 5.12. They correspond to the parameterization given in figure 3.9a&b and table 3.10 (paragraph 3.4.2):

model 24										
main chain	D_x	D_y	D_z	ϑ	α	La_x	La_y	Lb		
	[mm]	[mm]	[mm]	[°]	[°]	[mm]	[mm]	[mm]		
1	76	-16.5	0	0	90	119.963	-3	240		
2	76	-16.5	0	120	90	119.963	-3	240		
3	76	-16.5	0	240	90	119.963	-3	240		
additional 30 parameters for model 54										
main chain	d_z	$\Delta\alpha$	$\Delta\beta$	ΔC_x	ΔC_y	ΔC_z	ΔLb	b_x	b_y	b_z
	[mm]	[°]	[°]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]
1	20	0	0	0	0	0	0	24	0	0
2	20	0	0	0	0	0	0	24	0	0
3	20	0	0	0	0	0	0	24	0	0

Table 5.12 Nominal parameter set $\bar{\mathbf{p}}^n$ of the Delta mock-up.

A set of 74 measurement points was collected with the full-pose measurement set-up introduced in paragraph 4.2.4. The data are about uniformly distributed within a cube located in the center of the workspace¹ taking into account the rules defined in paragraph 5.3.9. This set of measurement data is used in the first two paragraphs of this section.

For the least-squares minimization the Levenberg-Marquardt (LM) method (section C.4) implemented in the optimization toolbox of MatLab™ was used [Moler 92, Grace 92]. All calculations were done on a PC with an Intel Pentium processor running at 90 MHz.

¹The point where arms and forearms are perpendicular to each other is considered to be the center of the workspace.

5.4.2 Full position measurement and model 24

For the identification of the parameters of model 24 (eq. 3.16) only the position of the end-effector and three joint angles are needed from the 74 measurement points collected.

Identification of the 24 parameters was performed for the implicit model 24 (eq. 3.16), which corresponds to implicit calibration (paragraph 5.3.3). This has the advantage that the identification can be split into the identification problem of three single-loop structures (fig. 5.2 and fig. 3.11) consisting of one joint-link train and the measurement device having eight parameters each. For one of these single-loops one row of the 74×8 identification Jacobian (eq. C.5) can be analytically differentiated. The eight partial derivatives consist of only 61 different factors.

With 3x9 iterations the LM-method has identified the following parameter set $\bar{\mathbf{p}}^i$ in 3x29 seconds using the nominal parameter set ($\bar{\mathbf{p}}^n$ -> table 5.12) as an initial guess:

model 24								
main chain	D_x	D_y	D_z	ϑ	α	La_x	La_y	Lb
	[mm]	[mm]	[mm]	[°]	[°]	[mm]	[mm]	[mm]
1	75.949	-16.41	0.404	0.064	89.881	119.960	-3.702	240.130
2	76.141	-16.709	0.182	120.008	90.044	119.933	-3.204	240.067
3	76.023	-16.624	0.062	239.998	90.010	119.990	-2.672	239.949

Table 5.13: Identified parameter set $\bar{\mathbf{p}}^i$ of model 24 based on implicit calibration .

The position error of the end-effector before and after calibration, calculated according to the flow chart given on the left hand side of table 5.9, is shown in figure 5.14:

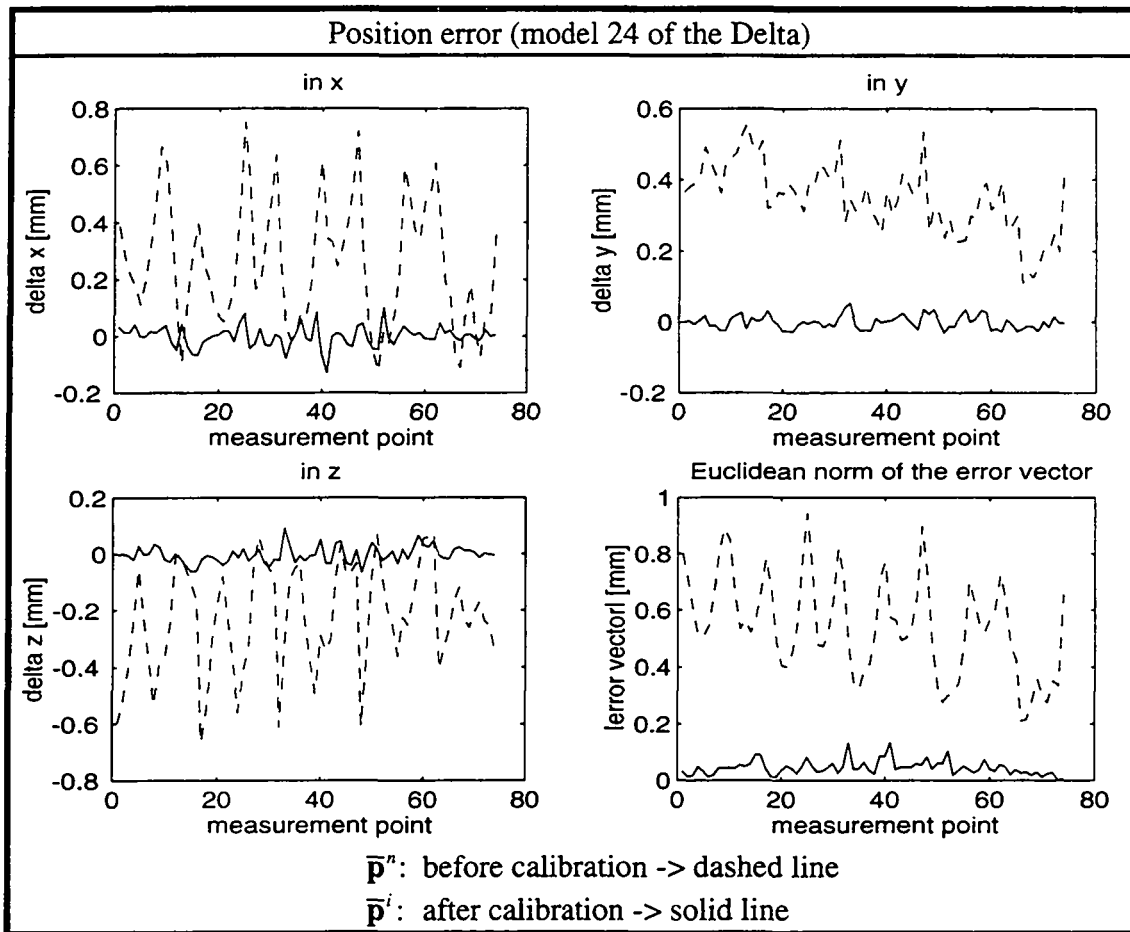


Fig. 5.14: Position error of the Delta end-effector before and after calibration.

Table 5.15 presents mean value and standard deviation of the differences between measured and calculated values. Further the improvement factors (table 5.9) are given:

direct solution (fig. 5.9 left hand side)					inverse solution (fig. 5.9 right hand side)				
Position error [μm]					Joint error [']				
	Δx	Δy	Δz	$ \Delta \bar{\mathbf{x}} $		$\Delta \theta_1$	$\Delta \theta_2$	$\Delta \theta_3$	$ \Delta \bar{\mathbf{q}} $
before calibration									
mean	260	350	-230	550	mean	12	11	-4	18
deviation	220	99	190	180	deviation	3.5	3.3	2.7	3.1
after calibration									
mean	-0.7	-0.1	-3.6	44	mean	-0.02	-0.02	-0.003	1.5
deviation	37	19	29	26	deviation	1.2	0.9	0.8	0.9
Factor of improvement for the position					Factor of improvement for the joint angles				
$F^{Pos} = 12.3$					$F^{Joint} = 11.9$				

Table 5.15: Position error of the end-effector and the joint angle errors before and after calibration based on model 24.

Implicit calibration was performed using the analytical identification Jacobian of model 24. Approximation of the identification Jacobian by finite differences leads to a less accurate result and requires generally more time and iterations. The calculation time (t_{cal}), the number of iterations (N_{itr}) and the number of times the function (model 24) had to be evaluated are shown in table 5.16. Besides, forward and inverse calibration were executed on model 24 leading to slightly different sets of parameters than the one found by implicit calibration (table 5.13). Semiparametric calibration needs no Jacobian for the identification of the 36 linear factors. The solution of the direct and inverse problem of the semiparametric model is similar to the solutions of model 24 given in section 6.3. Again it should be pointed out, that semiparametric calibration is only a good fitting calibration, if the 36 linear factors are used consequently.

Calibration	Number of the main chain	Calculation speed						Quality of improvement		
		Number of Iterations			Number of function evaluations			Suitable for quality control		
		without Jacobian			with Jacobian					
Variable		t_{cal}	N_{itr}	\bar{f}_{Cont}	t_{cal}	N_{itr}	\bar{f}_{Cont}	F^{Pos}	F^{Joint}	-
Units		[s]	-	-	[s]	-	-	-	-	
Implicit	1	186	17	256	35	9	32			y
"	2	129	13	177	29	10	31			y
"	3	141	13	186	23	8	24			y
"	total	456	43	619	87	271	87			y
"	1,2,3	1782 ¹	28	825	-	-	-	12.3	11.9	y
Forward	1,2,3	506	7	189	-	-	-	12.6	11.7	y
Inverse	1,2,3	318	6	161	-	-	-	12.0	12.2	y
Semi-parametric	1,2,3	0.5	0	1	-	-	-	15.2	15.0	n

Table 5.16: Gain in calculation speed of implicit calibration with and without the use of the analytical identification Jacobian. The quality of improvement of implicit, forward and inverse calibration is also given. Bold face numbers indicate the best values excluding semiparametric calibration.

¹The high calculation time is due to the numerical calculation of the identification Jacobian. The latter contains six 8x8 zero blocks (indicating that the main chains are decoupled for implicit calibration) provoking useless calculations.

Semiparametric calibration may be successfully applied, if there is no need for quality control. However table 5.16 shows only the *local validity* of a identified model, which means, that the same measurement points were used for both, identification and check for improvement. It may be therefore less surprising that a model with 36 independent parameter will better fit then a model with only 24 parameters. To check for *global validity* some authors [Zhuang 95] proposed to split the measurement points into two sets, one for identification and the other one for the improvement check. It was expected that due to the partial skipped geometric constraints, the global validity of the calibrated semiparametric model would be worse than the one received by implicit calibration. However, splitting the set of the 74 measurement points into two sets showed for both calibration methods *equal global validity*.

The question of the *minimal number of measurement points required for a reliable identification* is closely related to the question of finding the *optimal measurement set* discussed in paragraph 5.3.9. It is obvious that a sub-optimal set of measurement points will need more measurement points for a reliable calibration than the optimal set. In practice, a measurement set can be checked for the minimal number of measurement points required for a reliable calibration by successively reducing the number of points taken for calibration.

This is done in figure 5.17 for model 24 and the already used set of 74 measurement points which are equally distributed in the workspace of the Delta robot. This set was divided into equally distributed subsets of 0,8,12,16,24,32,40,48,56,64 and 74 points¹. With each subset a parameter identification of model 24 was performed leading to 11 parameter sets. Taking the direct solution (paragraph 6.3.1) the remaining error of the position (Euclidean norm) of the end-effector on the entire 74 points (table 5.9) was calculated. The mean value of these errors are plotted in figure 5.17 versus the number of points used for calibration.

¹ Taking zero points for calibration corresponds to the uncalibrated state, where the parameter set is given by the nominal values. Eight points are the minimal number required for the calibration based on model 24 when full position measurement is available (three redundant measurements). It leads to a *determined* non-linear system of equations, which can be solved exactly, i.e. with zero residuals.

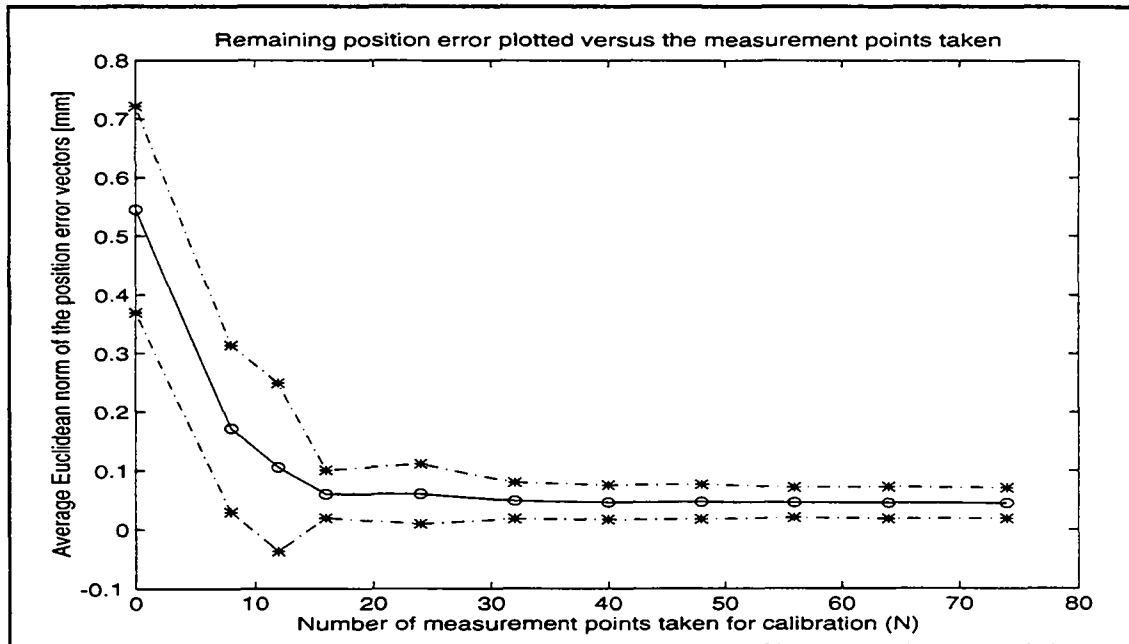


Fig. 5.17: Mean value and standard deviation plotted versus the number of measurement points used for implicit calibration which is based on model 24 of the Delta mock-up.

The experimental results of figure 5.17 show that an implicit calibration based on 2×8 measurement points yields already a quite reliable calibration. This *rule of thumb* of taking *twice as much measurement points as parameters of the model* is also supported by the work of Zhuang [91]. However, increase of the number of measurement points leads to a calibration with higher reliability.

5.4.3 Full pose measurement and model 54

For the identification of the 42 sensitive parameters of model 54 (eq. 3.12) the entire measured pose of the 74 collected measurement points is needed, including the measured deviations in the parallelism between the end-effector and the base.

The insensitive and nearly unobservable parameter set P2 (eq. 3.15) was set to its nominal value. Implicit calibration was performed by splitting the whole problem into three subproblems. Thus, each main joint-link train forming together with the measurement device a double-loop structure was identified separately. In 3x13 iterations using 3x740 seconds of calculation time, the LM-method has identified the following set of parameters ($\bar{\mathbf{p}}^i$) using the nominal parameter set ($\bar{\mathbf{p}}^n$ -> table 5.12) as an initial guess:

model 54										
24 parameters influencing the position										
main chain	D_x	D_y	D_z	ϑ	α	La_x	La_y	Lb		
	[mm]	[mm]	[mm]	[°]	[°]	[mm]	[mm]	[mm]		
1	75.902	-16.318	-0.642	-0.517	89.513	119.967	-3.693	240.074		
2	76.116	-16.716	-0.793	119.372	90.268	119.943	-3.236	240.108		
3	76.063	-16.792	-0.658	239.573	90.093	119.976	-2.673	240.009		
additional 30 parameters affecting the orientation & the position										
main chain	d_z	$\Delta\alpha$	$\Delta\beta$	ΔC_x	ΔC_y	ΔC_z	ΔLb	b_x	b_y	b_z
	[mm]	[°]	[°]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]
1	20	0.343	0.582	-0.023	-0.005	-0.006	0.092	24	0	0
2	20	-0.235	0.610	-0.049	0.069	0.023	0.009	24	0	0
3	20	-0.080	0.404	-0.049	0.076	-0.012	-0.062	24	0	0

Table 5.18: Identified parameter set $\bar{\mathbf{p}}^i$ of model 54 based on implicit calibration.

The position and orientation errors of the end-effector before and after calibration were calculated according to the flow chart given on the left hand side of table 5.9. The position error is not plotted in figure 5.19, since it differs only slightly from figure 5.14.

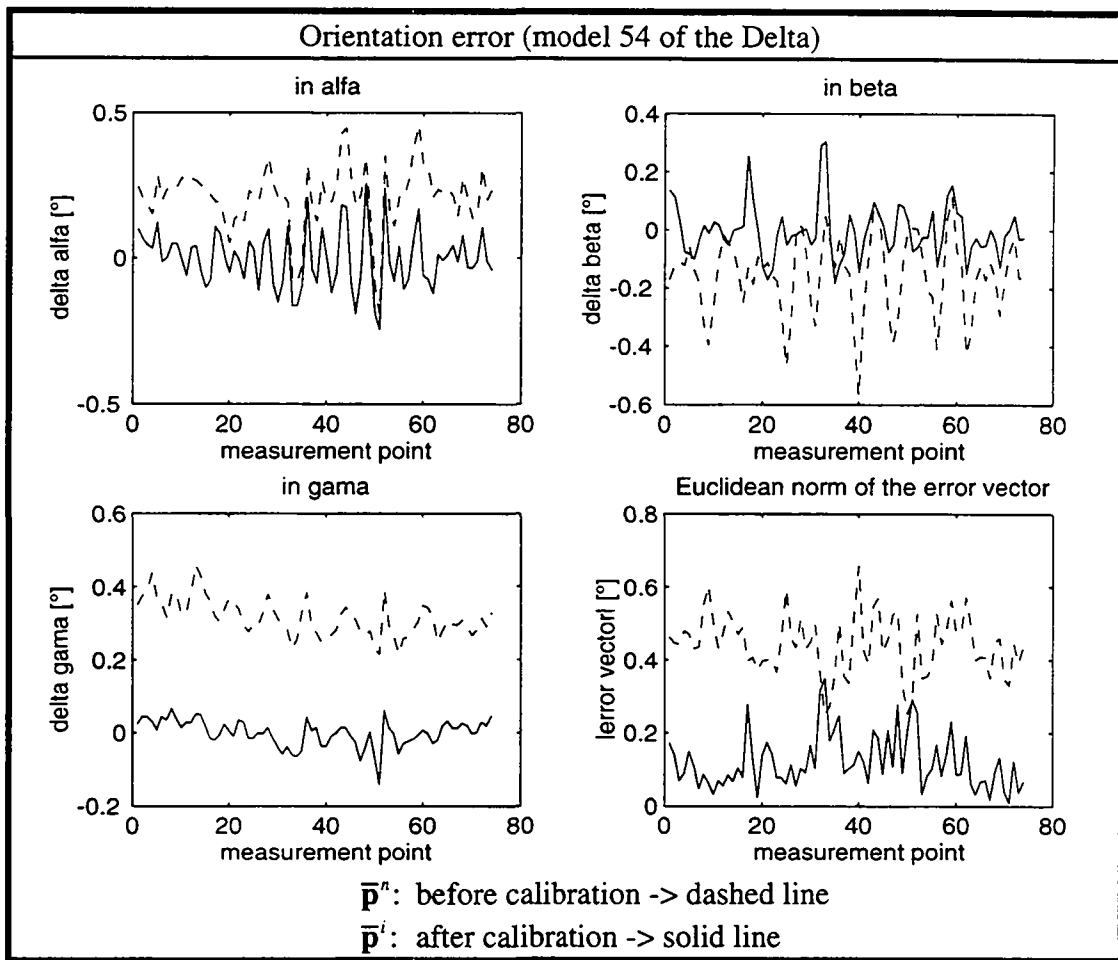


Fig. 5.19: Orientation error of the Delta end-effector before and after calibration.

Table 5.20 shows the improvement factors (table 5.9 & eq. 5.20) as well as mean value and standard deviation of the difference between measured and calculated values.

direct solution (left hand side of fig. 5.9)												
Position error [μm]					Orientation error [']				Joint error [']			
Δx	Δy	Δz	$ \Delta \bar{\mathbf{x}} $	$\Delta \alpha$	$\Delta \beta$	$\Delta \gamma$	$ \Delta \bar{\mathbf{e}} $	$\Delta \theta_1$	$\Delta \theta_2$	$\Delta \theta_3$	$ \Delta \bar{\mathbf{q}} $	
before calibration												
mean	260	350	-230	550	13	-9.4	18	27	12	11	-4	18
deviation	220	99	190	180	6.8	8.2	3.1	4.9	3.5	3.3	2.7	3.1
	after calibration:											
mean	-1.4	1.0	6.7	53	-0.10	-0.25	-0.04	7.2	-0.05	-0.07	-0.13	1.9
deviation	42	32	34	32	6.1	5.6	2.2	4.5	1.5	1.2	1.2	1.3
Improve- ment	$F^{Pos} = 10.1$				$F^{Ori} = 3.7$				$F^{Joint} = 9.2$			
					inverse solution (left hand side of fig. 5.9)							

Table 5.20: Factors of improvement and statistical values

5.4.4 Plane/Spheres set-up and model 24

Hornick [91] proposed to use a simple precision plane for calibration, which is arbitrarily located in the workspace of the robot. For such an arrangement (left hand side of figure 4.6) measurement data were collected. Identification of the parameters of model 24 was attempted applying the LM-method but the algorithm was *diverging*.

This is not surprising since some of the parameters in model 24 are unobservable. The robot is not constraint in the three degrees of freedom of the plane. During identification the not fully constraint parameters are able to twist around the z-axis of the {B}-frame and to translate laterally in x- and y- direction.

Driels [91] proposed to use a simple rod equipped with two S-joints at each end (ballbar) to connect the end-effector to its base (right hand side of figure 4.6). Combination of such a ballbar and the plane is referred to as *plane/spheres set-up* (paragraph 4.2.6). The idea is to introduce three additional constraints. A ballbar will guide the end-effector on a sphere. Combining the plane with a sphere will omit the translation in x- and y-direction. A second sphere is necessary to avoid the robot twisting around the z-axis.

During the measurement step three different kinds of data sets have to be acquired:

- The first set is acquired while the end-effector is sliding on the plane.
- The second set is acquired while the end-effector is connected to a first hole in this plane by a ballbar.
- The third set is acquired by connecting the end-effector to a second hole in the plane by a ballbar.

To choose the holes within the added precision plane was induced by the topology of the industrial version of the Delta robot [Clavel 91]. The current construction of the Delta robot is that of a portal robot having the end-effector below its base (fig. 1.9 and 4.6). Below the end-effector a plate is needed to support the work piece, which is referred to as *interface plane*. This interface plane with two additional holes offers several advantages for the calibration: Firstly, the robot will be calibrated relative to the plane and the holes, which can be used to locate the work piece in the workspace of the robot. Secondly, it is not very convenient to locate a work piece relative to a {B}-frame, which is attached to the base plate of the hanging robot. Alignment of the work piece is much easier if the {B}-frame is physically present on the interface plane. This means that the first hole defines the origin of the {B}-frame, the second hole fixes the direction

of the x-axis whereas the z-axis stays perpendicular to the plane. Thirdly, fast and easy recalibration becomes possible.

Simulations of the identification step using the plane/spheres set-up have shown a serious mathematical drawback. By applying the method of simulated annealing [Press 89] it was shown that many (hundreds) other minima are located around the minimum created by the accurate parameter set used to generate the measurement data in presents of measurement noise. Choosing the right minimum among the many minima is an open research topic [Merlet 93b]. Simple rules such as taking the parameter set closest to the nominal one fail. It is also *not a solution to search for the global minimum*, since *some of the parameter sets will fit the noisy measurement data even better than the accurate set!*

The problem discussed above is the one of *global validity* of the identified data set. A measurement set is collected in a part of the workspace in the hope that the identified parameter set will be valid in the entire workspace. Simple convergence of an algorithm is not a proof of its ability to find the parameter set with best global validity. Here, simulation offers the advantage that this best global valid set of parameters is known as the accurate set of parameters used to generate the data set. Simulating a different method (e.g. implicit, forward or inverse) accordingly the previous section (5.3) is a numerical approach to solve the problem of multiple minima by searching for the method which approaches the accurate set closest.

Following intuitive rule may be of any use: The more complex (nonlinear) the mathematical structure of the underlying model is, the more minima will be generated and the further away of the accurate parameter set the search algorithm will be trapped in a minimum, which fits well the measurement data but is unsuitable for the rest of the workspace.

For the plane/spheres set-up the solution of the direct problem must be substituted into the equation of a plane and a sphere, respectively. Each measurement point leads to one equation with a highly complex mathematical structure, what explains the great number of multiple minima.

To conclude, the plane/spheres set-up seeks for an algorithm capable of identifying a parameter set which is *globally valid*. For the plane/sphere set-up the global validity of the identified parameter set gets particularly important for the following reason: From measurement data collected on an x-y plane a set of parameters should be identified which is valid in x-, y-, and z-direction. The z-axis is the direction of insertion for the

Delta robot and therefore crucial. High accuracy is thus needed. Today's algorithms fail at solving this problem, since they get trapped by one of the minima around the searched minima.

5.4.5 Short-cut set-up and model 54

The short-cut set-up (paragraph 4.2.7) consists of building a single-loop structure, which is redundant in the actuator readings with the mechanical parts of a multi-loop structure. For the Delta robot this is easily done since its S-joints are separable. This allows to build a RSSR-loop (fig. 4.7).

The number of independent parameters required to describe such a loop can be calculated according to eq. 3.3. For a not arbitrarily located base $\{b\}$ and moving $\{p\}$ frame it results in 9 parameters including the two encoder offsets. Using the same parameterization as for model 54 (eq. 3.9 and table 3.10) would lead to 32 parameter for the RSSR-loop, which is highly redundant. This may be explained by the arbitrarily located $\{B\}$ -frame in model 54 becoming unobservable and by the used concept of model 54 representing a main joint-link train by the difference and the sum of its two closure equations.

Thus, a new parameterization based on 9 parameters must be introduced. The identification of such a single loop, where redundant measurement is provided by the two encoders, is not a problem. With the six forearms and the six proximal S-joints, 72 different combinations exists to built a RSSR-loop. Each time, 9 parameters can be identified, which depend on each other. Out of this at most 72×9 parameters some of the 54 parameters can be extracted, such as for example the arm and forearm lengths and the location of the motor axes relative to each other. Parameters which cannot be identified this way are the position of the distal S-joints located on the end-effector because this mechanical part isn't included in the RSSR-loops. However, as could be seen in paragraph 5.4.3 the sizes of the end-effector are insensitive parameters in the model 54 and can therefore be set to their nominal values.

The short-cut set-up strongly depends on the topology of the structure to be calibrated. However, since one of the goals of this work is to search for *standard* calibration procedures for parallel topologies, the short-cut approach was not further investigated.

5.4.6 Conclusion

By implicit calibration based on model 24 and full position measurement an accuracy improvement of about a factor of 10 could be gained on the mean value whereas the standard deviation is 7 times improved. For the size of the Delta mock-up (table 5.12) this corresponds to a improvement of 500 to 50 micrometers for the mean value and for the standard deviation of 200 to 30 micrometers (table 5.15).

By implicit calibration based on model 54 and full pose measurement an improvement of the *prediction* of the orientation of a factor four could be gained, whereas the improvement of the position is slightly inferior¹ as compared to model 24. The standard deviation for the orientation decreased only slightly. For the size of the Delta mock-up, this corresponds to a improvement of $1/2$ to $1/10$ of a degree for the mean value and for the standard deviation to a slight improvement of $1/12$ to $1/13$ of a degree. To estimate the improvement gained including angular deviations of the end-effector a rod of 120 millimeter length may be fixed at the center of the end-effector with its longitudinal axis coincident with the z-axis of the {P}-frame. The mean value of the position error of the distal rod tip improves from 3.7 to 0.25 millimeter, which corresponds to a 15fold improvement in prediction of the position of the distal tip.

The plane/sphere set-up may also be successfully applied if the problem generated by the multiple minima can be solved. Measurement sets could therefore be split into several subsets. Each data set could be screened for minima leading each time to a different set of minima. Overlying these different sets will show a density distribution of minima. The searched set of parameters may be chosen in the region of the parameter space with the highest density of minima.

Identification using the short-cut set-up seems to be straight forward. However, it wasn't experimentally verified. Treatment of the mechanical parts not included in the single loop is a problem to be solved. Besides, it has to be shown that such kind of multi-stage calibration based on single-loop structures is suitable to provide a set of parameters which leads to an accuracy improvement of the multi-loop structure.

¹Model 54 is much more non-linear than model 24. Identification of a reliable parameter set becomes therefore more difficult (see discussion in paragraph 5.4.4).

5.5 Parameter identification for the Argos structure

5.5.1 Introduction

Table 5.21 contains the nominal parameters ($\bar{\mathbf{p}}^n$) of the Argos mock-up (section 2.4) according to the parameterization given in figure 3.14a&b and table 3.15 (paragraph 3.5.2):

model 9						
main chain	α_0	Ω	Φ	ω	φ	
	[°]	[°]	[°]	[°]	[°]	
1	-45	-	-	-	-	
2	45	-	90	-	90	
3	-45	90	90	90	-90	
additional 18 parameters for model 27						
main chain	ΔA	ΔC	Δa	Δc	γ	R
	[mm]	[mm]	[mm]	[mm]	[°]	[mm]
1	-	-	0	0	90	110
2	-	0	0	0	90	110
3	0	0	0	0	90	110

Table 5.21 Nominal parameter set $\bar{\mathbf{p}}^n$ of Argos mock-up.

A set of 47 measurement points was collected with the full-pose measurement set-up proposed in paragraph 4.3.4. The parameter identification of the two models 9 and 27 are both based on this set. However, experiments have shown that parameterization based on the assumption of not arbitrarily located $\{b\}$ and $\{p\}$ is not suitable for identification. There are two reasons for this:

- 1) Referencing the external measurement devices perfectly with respect to the *not arbitrarily located frames* is impossible.
- 2) Each of the three mechanically identical chains has a *different number* of kinematic parameters, e.g. for model 9 (table 5.21) one, three and five. Such a parameterization is called *asymmetric* in contrast to a symmetrical parameterization having the same number of parameters for each of joint-link train. The chain with the fewest number of parameters becomes too "stiff" during identification resulting in a poor accuracy improvement.

The first problem can be solved by adding two arbitrary frames $\{B\}$ and $\{P\}$. The $\{B\}$ -frame will be automatically adapted to the base frame of the measurement device, whereas the $\{P\}$ -frame will be automatically adapted to the moving frame of the measurement device during identification. Unfortunately, decoupling of the identification for each joint-link train during implicit calibration is no longer possible.

Once these arbitrary frames are added, the second problem can be solved by introducing a new symmetrical parameterization leading to a decoupling of each joint-link train for implicit calibration. Unfortunately, the direct problem of the perfect spherical Argos structure can no longer be reduced to a univariate polynomial without parameter transformation.

Here only the first problem was solved by adding two arbitrary frames whereas the parameterization was left asymmetrical in order to keep the direct problem solvable. According to equation 3.4 a perfect spherical Argos structure needs 15 independent parameters. Therefore three angles for the transformation between the $\{b\}$ and the $\{B\}$ -frame and another three for the transformation between the $\{p\}$ and $\{P\}$ -frame were added. Due to the construction of the orientation unit for the Argos structure (4.3.3) its encoders could not be referenced with respect to the device itself and thus three offsets of its encoders were added, too.

In summary, *nine additional parameters* for the alignment and calibration of the external measurement devices were added to each of the two calibration models 9 and 27¹. 18 parameters were thus identified for the model 9 and 36 for model 27. These additional 9 parameters are interpreted to belong to the measurement device and are therefore not presented in this work.

As for the Delta robot the least-squares minimization was performed using the Levenberg-Marquardt (LM) method (section C.4) implemented in the optimization toolbox of MatLab™. A Pentium based PC running at 90 MHz was again the hardware platform.

¹According to equation 3.3, a complete spatial model with arbitrarily located base $\{B\}$ and moving frame $\{P\}$ requires additional 6 parameters leading to a total of 42 parameters including the encoder offsets of the orientation measurement unit.

5.5.2 Full orientation measurement and model 9

For the identification of the parameters of model 9 (eq. 3.26) only the orientation of the end-effector pose and the three joint angles of the measurement set are needed.

Within 6 iterations and 420 seconds calculation time the LM-method has identified the following parameter set $\bar{\mathbf{p}}^i$ using the nominal parameter set ($\bar{\mathbf{p}}^n$ -> table 5.21) as an initial guess:

model 9					
main chain	α_0	Ω	Φ	ω	φ
	[°]	[°]	[°]	[°]	[°]
1	-44.02	-	-	-	-
2	47.89	-	91.27	-	89.82
3	-42.17	90.45	91.85	91.18	-87.56

Table 5.22 Identified parameter set $\bar{\mathbf{p}}^n$ of model 9 based on implicit calibration.

The orientation error of the end-effector before and after calibration is calculated according the flow chart given on the left hand side of table 5.9 and presented in figure 5.23.

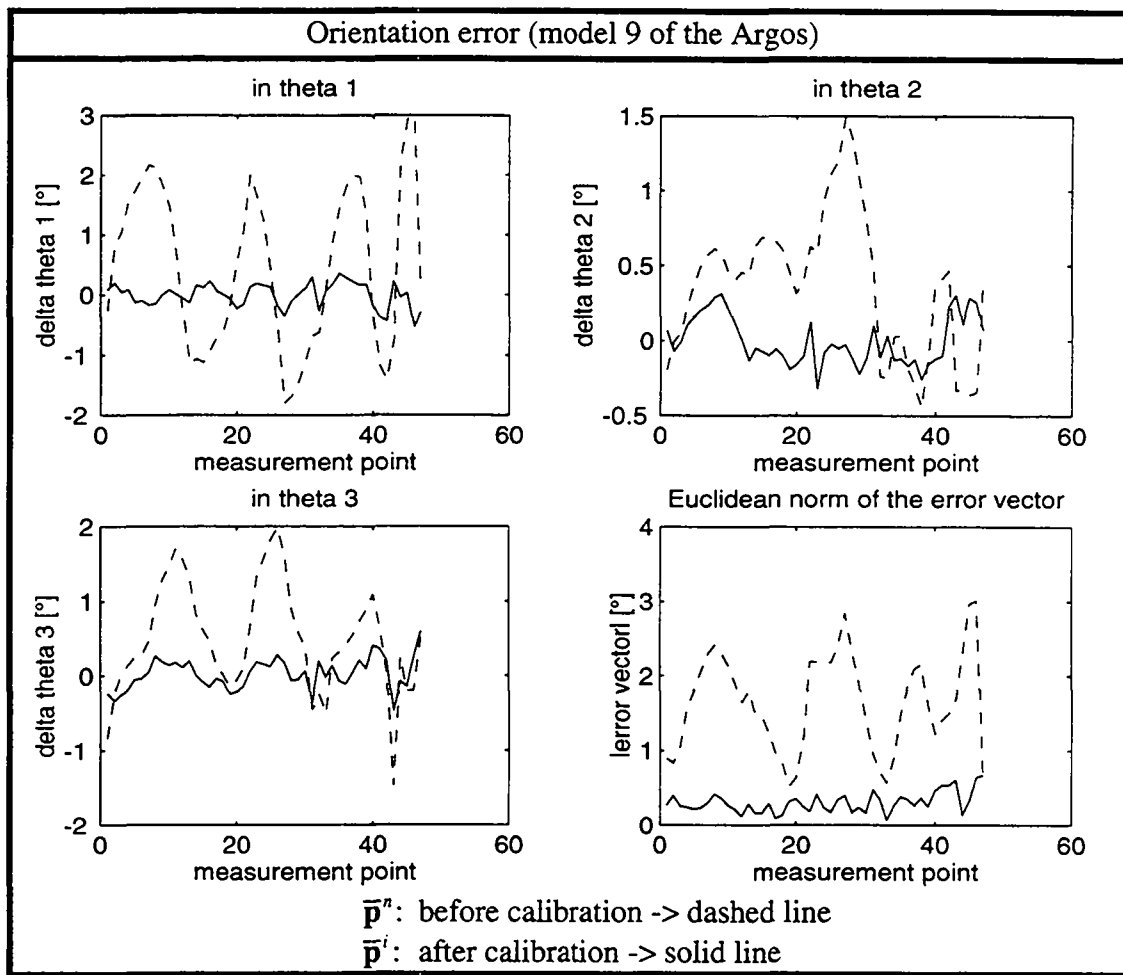


Fig. 5.23: Orientation error of the Argos end-effector before and after calibration.

Table 5.24 shows the improvement factors (table 5.9) as well as the mean value and the standard deviation of the difference between measured and calculated values.

direct solution (fig. 5.9 left hand side)					inverse solution (fig. 5.9 right hand side)				
Orientation error [°]					Joint error [°]				
	$\Delta\vartheta_1$	$\Delta\vartheta_2$	$\Delta\vartheta_3$	$ \Delta\bar{e} $		$\Delta\alpha_1$	$\Delta\alpha_2$	$\Delta\alpha_3$	$ \Delta\bar{q} $
before calibration									
mean	28	22	30	99	mean	28	-6.2	28	95
deviation	81	29	44	40	deviation	81	28	40	37
after calibration									
mean	-0.32	-0.17	1.5	18	mean	-0.13	-0.03	0.16	18
deviation	12	9.7	13	8.5	deviation	12	12	12	10
Improvement factor of the orientation					Improvement factor for the joint angles				
$F^{Ori} = 5.4$					$F^{Joint} = 5.2$				

Table 5.24: Improvement factors and statistical values

5.5.3 Full pose measurement and model 27

For identification of the parameters of model 27 (eq. 3.22) the entire pose of the 47 collected measurement points is required including the deviation of the position of the virtual rotation center.

The LM-method identified within 19 iterations and 2395 seconds¹ calculation time the following set of parameters ($\bar{\mathbf{p}}^i$) using the nominal parameters ($\bar{\mathbf{p}}^n$ -> table 5.21) as an initial guess:

model 27							
first 9 parameters							
main chain	α_0	Ω	Φ	ω	φ		
	[°]	[°]	[°]	[°]	[°]		
1	-39.33	-	-	-	-		
2	44.17	-	89.57	-	90.43		
3	-40.37	92.44	85.71	90.72	-87.53		
and additional 18 parameters							
main chain	ΔA	ΔC	Δa	Δc	γ	ΔR	R
	[mm]	[mm]	[mm]	[mm]	[°]	[mm]	[mm]
1	-	-	2.39	-3.20	94.85	-4.95	105.86
2	-	1.00	4.65	-10.14	92.47	13.56 ²	98.31
3	0.32	-0.05	2.19	1.79	91.89	-0.86	107.85

Table 5.25: Identified parameter set $\bar{\mathbf{p}}^i$ of model 27 based on implicit calibration.

The orientation and position error of the end-effector before and after calibration is calculated according the flow chart given on the left hand side of table 5.9. The orientation error is not plotted in figure 5.26 since it differs only slightly from figure 5.23, where the orientation error vector of model 9 was shown.

¹The increased calculation time is due to the couplings introduced by the nine additional parameters (see paragraph 5.5.1).

²Such a high value indicates that the second chain contains significant manufacturing and alignment errors which may be reduced by refabrication and readjustment of this chain.

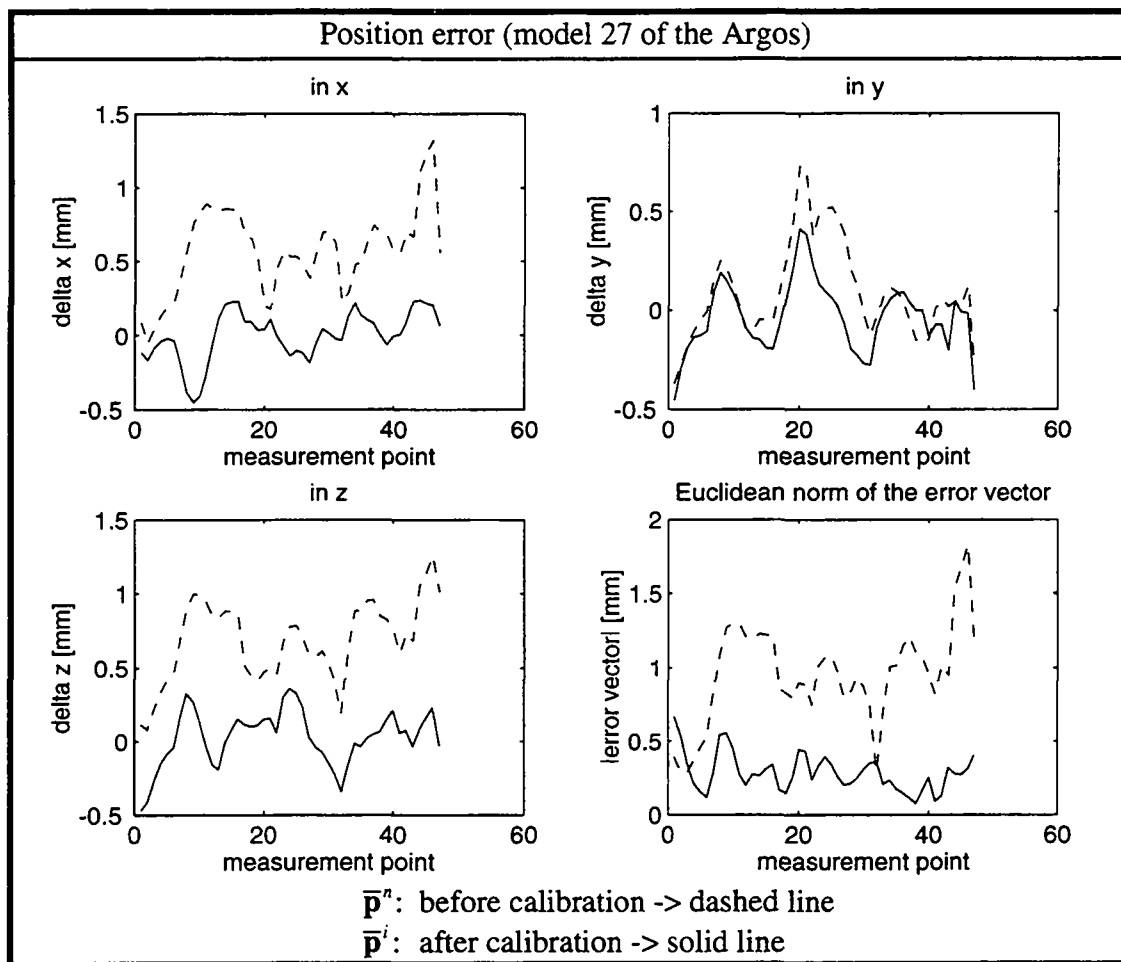


Fig. 5.26 Position error of the Argos end-effector before and after calibration.

The improvement factors (table 5.9 & eq. 5.20) as well as mean value and standard deviation are given in table 5.27:

direct solution (fig. 5.9 left hand side)												
Orientation error [']					Position error [μm]				Joint error [']			
$\Delta\vartheta_1$	$\Delta\vartheta_2$	$\Delta\vartheta_3$	$ \Delta\bar{\mathbf{e}} $	Δx	Δy	Δz	$ \Delta\bar{\mathbf{x}} $	$\Delta\alpha_1$	$\Delta\alpha_2$	$\Delta\alpha_3$	$ \Delta\bar{\mathbf{q}} $	
before calibration												
mean	28	22	30	99	574	94.4	688	958	28	-6.2	28	95
deviation	81	29	44	40	294	251	271	338	81	28	40	37
	after calibration:											
mean	-0.04	0.91	0.21	19	-3.0	-36	26	285	-0.14	0.90	1.1	18
deviation	11	13	11	8.7	167	179	189	130	11	14	10	10
Improve- ment	$F^{Ori} = 5.3$				$F^{Pos} = 3.4$				$F^{Joint} = 5.3$			
inverse solution (fig. 5.9 right hand side)												

Table 5.27: Improvement factors and statistical values

5.5.4 Conclusion

By calibration based on model 9 and full orientation measurement a accuracy improvement of a factor 5.4 for the Argos structure could be gained on the mean value whereas the standard deviation decrease by a factor of 4.7. This corresponds to an absolute improvement of 1.7 to 0.3 degrees for the mean value and 0.67 to 0.14 degrees for the standard deviation.

By calibration based on model 27 and full pose measurement the prediction of the position of the virtual rotation center improved by about a factor of 3.4 for the mean value and by a factor of 2.6 for the standard deviation. For the chosen size of the Argos mock-up this corresponds to an absolute quantitative improvement of 960 to 285 micrometers for the mean value, whereas the standard deviation drops from 338 to 130 micrometers.

Asymmetrical parameterization and non-arbitrarily located base and moving frames should be avoided for reliable identification.

5.6 Conclusion

In this chapter identification of kinematic parameters was addressed. In a first section the calibration problem was stated as a unconstrained non-linear least-squares problem. Based on a review of standard estimation method in annex C, the Gauss-Newton method based on singular value decomposition (SVD) as well as the Levenberg-Marquardt method were proposed due to their robustness.

In a further section it was shown by simulation that implicit calibration works well for closed-loop structures offering several advantages: Firstly, the identification problem of multi-loop structures using implicit calibration may be split into identification of the kinematic parameters of some sub-structures. Secondly, the identification Jacobian may be differentiated analytically and thirdly, the identified parameter set is the closest to the accurate parameter set, which is globally valid. Thus, implicit calibration was proposed as standard calibration method for parallel robots. Semiparametric calibration may be applied if there is no need for quality control of the mechanical parts of the robot.

In the last two sections the theoretical tools introduced were experimentally tested on the two examples namely the Delta and the Argos structure showing a gain in accuracy of 3.4 up to 12.3. This experimental verification proves that the theoretical tools work well. The two approaches using no external measurement system couldn't be treated in depth. Simplifications of the measurement step seems to lead to a considerably higher complexity in the identification step. This observation is also supported by Hollerbach [93].

The work presented in this chapter shows that *improving the accuracy of parallel robots by calibration is possible.*

6. Implementation

6.1 Introduction

In this chapter the question of how to solve the direct and the inverse problem of the identified calibration models is addressed. In addition the solution for the nominal model of the Argos structure is presented. These solutions can either be used to simulate a robot in a virtual environment such as a CAD system for off-line programming [Mooring 91] or for an implementation of the identified model in the robot controller. The second task leads to the question of how to generate these solutions in real-time.

If the nominal models can be solved in a closed form, it is in general due to some assumptions made about the robot's geometry. A very popular example is the Pieper's solution of the inverse problem of a serial robot with six degrees of freedom and three consecutive axes intersecting at a point [Craig 89]. For higher accuracy however modeling of the slight misalignment of these three axes is necessary. Without the assumption of three consecutive intersecting axes the problem is no more solvable in a closed form. An example of a robot with non-intersecting axes is a general serial 6R robot, where the solution of the inverse problem leads to a polynomial of 16th order [LeeH 91].

Model 24 of the Delta robot and model 9 of the Argos structure are quite easily reducible to a univariate polynomial for the direct as well as for the inverse problem. Model 54 and model 27 however are more difficult to solve. Their direct problem can also be expressed as univariate polynomial of 40th and 16th degree respectively, but these polynomials are not suitable for real-time applications, because of their high calculation time. In order to get real-time solutions for these two models, three different approaches are of interest:

Aiming at numerical solutions, for example the Newton-Raphson algorithm could be applied. This algorithm requires an initial guess of the solution, which can be supplied by solving the nominal model. The algorithm is very efficient for this kind of problem and needs generally only few iterations (typically three) to converge. In case of multiple solutions the algorithm has the advantage that the right solution can be found by

starting the iteration very closely to the solution of interest. However, the convergence to the right solution is only highly probable but not guaranteed. Commercially available controllers are generally less powerful than for example the transputer-based system used to control the direct drive version of the Delta robot [Codourey 91]. Therefore numerical solutions may be too time consuming for a real-time application.

A second possibility to obtain real time solutions is the numerical precalculation of solutions of the direct and inverse problem, which are then stored in a look-up table of the robot controller and used for real-time interpolation. In order to facilitate the interpolation between the precalculated points, only the differences between the nominal and the calibrated solution can be stored, which varies less than the solution of the entire calibration model. A disadvantage of this method might be the limitation of storage place available and the access time to the storage medium.

A third possibility is to simplify the calibration models leading to low computational effort. Two contradictory criteria are encountered thereby: The easier to solve the modified model, the faster it is, but the bigger is also the resulting error of its solution compared to the original model. Finding such simplified models can be quite laborious, but has an economical advantage over the other two propositions: If the complexity of the simplified is comparable to the nominal solution, the controller hardware does not have to be modified resulting in low manufacturing costs.

To solve model 54 and model 27 possibility one and three will be investigated, whereas for the other models polynomial solutions will be presented.

6.2 The four basic problems

6.2.1 Introduction

In this section three different direct and one inverse kinematic problem will be solved. Their solutions will be employed to solve the different models of the Delta and Argos structure. To illustrate the kinematic problems encountered, a position and orientation decoupled robot is shown in figure 6.1. It consists of a 6[R2S] structure, where three of the S-joints located on the end-effector are coincident [Innocenti 92, Patarinski 93].

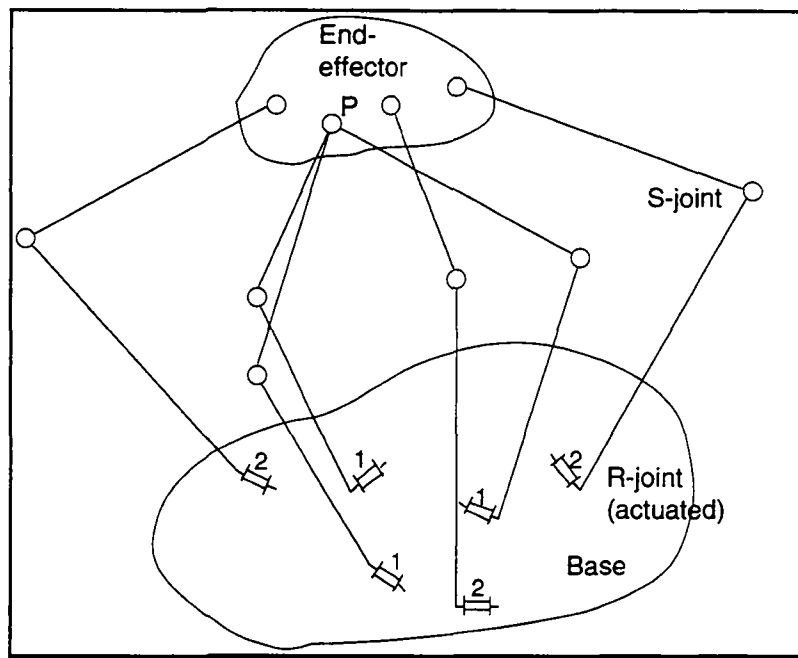


Fig. 6.1: Position / Orientation decoupled parallel structure with six degrees of freedom

The first problem is here referred to as "*general position problem*" and consists of calculating the position of the point P in figure 6.1 for a given set of the first three joint angles. Once the position of point P is known, it remains to calculate the orientation of the end-effector, which is called the "*general orientation problem*". This cascaded calculation of the pose of the end-effector is only possible because of the very particular structure of the robot given in figure 6.1. Not taking into account this decoupling effect would mean to calculate the pose of the end-effector for a given set of six joint angles, which is referred to as "*general pose problem*".

The fourth and last problem is the calculation of the resulting joint angle for a given pose of the end-effector, which will be referred to as the "*general joint angle problem*".

Table 6.2 shows a small selection of Literature on these four problems:

problem	Solved by:	possible geometric interpretation	examples of robots (nominal models)
direct problems			
position	[Sternheim 87] [Clavel 91] [Pierrot 91] [Innocenti 92] [Wohlhart 94]	- intersection point of three spheres - intersection of two small circles	[Clavel 85] Delta [Thornton 88] GEC Tetrabot [Parikian 95] modified Pollard robot [Vischer 95] PantoScope (annex A.1)
orientation	[Gosselin 92] [Innocenti 92] [Innocenti 93] [Wohlhart 94]	- fitting a triangle to three small circles	[Pollard 42] Pollard robot [Gosselin 94] Agile eye [Vischer 95] Argos (annex A.1)
pose	[Raghavan 93] [Husty 94]	- fitting a irregular hexahedron to six spheres	[Fichter 86] Stewart Platform [Pierrot 91] Hexa
inverse problem			
joint angle	[Sternheim 87] [Pierrot 90] [Clavel 91]	- intersection of a circle with a sphere - intersection of two small circles	[Clavel 85] Delta [Gosselin 94] Agile eye

Table 6.2: Selection of literature on the "general position, orientation, pose and joint angle problem"

The first three problems are represented by a set of non-linear equations coupled with respect to the unknown variables, which is typical for the direct problem of parallel robots. This is in contrast with the inverse problem of parallel robots, which is decoupled with respect to the unknown variables as long as the number of world coordinates is equal to the number of joint coordinates as can be seen by examination of the "general joint angle problem".

6.2.2 The general position problem

The "general position problem" was for instance solved by Clavel [91] (table 6.2). A summary is given in this paragraph since the results will later be used in the sections 6.3, 6.4 and 6.7. According to Clavel the "general position problem" of finding the world coordinates x, y, z of point P in figure 6.1 can always be expressed as:

$$\{E_i, F_i, G_i\} \cdot \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} + H_i = -(x^2 + y^2 + z^2) \quad \text{or} \quad \bar{\mathbf{V}}_i \cdot \bar{\mathbf{P}} + H_i = -\bar{\mathbf{P}}^T \cdot \bar{\mathbf{P}} \quad i=1..3 \quad (6.1)$$

The system of eqs. 6.1 represents the problem of finding the intersection point of three spheres. The factors E_i, F_i, G_i and H_i contains the known kinematic parameters as well as the joint coordinates of the robot.

The right hand side of this nonlinear system of three coupled equations (eq. 6.1) is free of design parameters and will therefore vanish for each difference of two equations. Thus, a linear system of equations can be established, which has linearly dependent rows:

$$\begin{bmatrix} E_1 - E_2 & F_1 - F_2 & G_1 - G_2 & H_1 - H_2 \\ E_2 - E_3 & F_2 - F_3 & G_2 - G_3 & H_2 - H_3 \\ E_3 - E_1 & F_3 - F_1 & G_3 - G_1 & H_3 - H_1 \end{bmatrix} \cdot \begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix} = 0$$

and solved for x and y as a function of z :

$$x = \frac{N_2}{N_1}z + \frac{N_3}{N_1} \quad \text{and} \quad y = \frac{N_4}{N_1}z + \frac{N_5}{N_1} \quad (6.2)$$

with the following determinants as factors:

$$\begin{aligned} N_1 &= \begin{vmatrix} E_1 - E_2 & F_1 - F_2 & 0 \\ E_2 - E_3 & F_2 - F_3 & 0 \\ E_3 - E_1 & F_3 - F_1 & 1 \end{vmatrix} = E_1F_2 - E_1F_3 + E_2F_3 - E_2F_1 + E_3F_1 - E_3F_2 \\ N_2 &= \begin{vmatrix} 0 & F_1 - F_2 & G_1 - G_2 \\ 0 & F_2 - F_3 & G_2 - G_3 \\ 1 & F_3 - F_1 & G_3 - G_1 \end{vmatrix} = F_1G_2 - F_1G_3 + F_2G_3 - F_2G_1 + F_3G_1 - F_3G_2 \\ N_3 &= \begin{vmatrix} 0 & F_1 - F_2 & H_1 - H_2 \\ 0 & F_2 - F_3 & H_2 - H_3 \\ 1 & F_3 - F_1 & H_3 - H_1 \end{vmatrix} = F_1H_2 - F_1H_3 + F_2H_3 - F_2H_1 + F_3H_1 - F_3H_2 \\ N_4 &= \begin{vmatrix} E_1 - E_2 & 0 & G_1 - G_2 \\ E_2 - E_3 & 0 & G_2 - G_3 \\ E_3 - E_1 & 1 & G_3 - G_1 \end{vmatrix} = -E_1G_2 + E_1G_3 - E_2G_3 + E_2G_1 - E_3G_1 + E_3G_2 \\ N_5 &= \begin{vmatrix} E_1 - E_2 & 0 & H_1 - H_2 \\ E_2 - E_3 & 0 & H_2 - H_3 \\ E_3 - E_1 & 1 & H_3 - H_1 \end{vmatrix} = -E_1H_2 + E_1H_3 - E_2H_3 + E_2H_1 - E_3H_1 + E_3H_2 \end{aligned} \quad (6.3)$$

The last step consists of substituting eq. 6.2 into the first equation of 6.1 and solving the resulting quadratic equation for z as:

$$z_{1,2} = \frac{-M \pm \sqrt{M^2 + 4LQ}}{2L} \quad (6.4)$$

with the factors:

$$\begin{aligned}
 L &= N_1^2 + N_2^2 + N_4^2 \\
 M &= N_1(E_1N_2 + F_1N_4 + G_1N_1) + 2N_2N_3 + 2N_4N_5 \\
 Q &= N_1(E_1N_3 + F_1N_5 + H_1N_1) + N_3^2 + N_5^2
 \end{aligned} \tag{6.5}$$

The "general position problem" is now solved by eqs. 6.2 and 6.4 with the factors 6.3 and 6.5:

$$\begin{aligned}
 &\boxed{z_{1,2} = \frac{-M \pm \sqrt{M^2 + 4LQ}}{2L}} \\
 \rightarrow &\boxed{\begin{aligned} x &= \frac{N_2}{N_1}z + \frac{N_3}{N_1} \\ y &= \frac{N_4}{N_1}z + \frac{N_5}{N_1} \end{aligned}} \quad \text{with the factors} \quad \boxed{\begin{aligned} N_{1..5} &\rightarrow \text{eq. 6.3} \\ L, M, Q &\rightarrow \text{eq. 6.5} \end{aligned}} \tag{6.6}
 \end{aligned}$$

It has either no or two sets of solutions (x, y, z) . As already mentioned a geometrical interpretation of the system of eqs. 6.1 is finding the intersection of three spheres, which has not more than two points.

6.2.3 The general orientation problem

Gosselin [92] solved the "general orientation problem" by means of the trigonometric identity. A similar problem is treated in annex D.2.1. Innocenti [92] employed the method of resultants as explained in annex D.2.2. In this section the solution of Innocenti [93] is reviewed and an enhancement proposed for the symbolical representation of the factors of the univariate polynomial. The solution of the "general orientation problem" will be used in section 6.6 and 6.7. According to Innocenti the "general orientation problem" can always be established with only two unknown angles describing the orientation of the end-effector in figure 6.1 if point P is assumed to stay fixed with respect to the base:

$$\begin{aligned}
 &\{\cos(\vartheta_2), \sin(\vartheta_2), 1\} \cdot \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \cdot \begin{Bmatrix} \cos(\vartheta_3) \\ \sin(\vartheta_3) \\ 1 \end{Bmatrix} = 0 \\
 &\{\cos(\vartheta_2), \sin(\vartheta_2), 1\} \cdot \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \cdot \begin{Bmatrix} \cos(\vartheta_3) \\ \sin(\vartheta_3) \\ 1 \end{Bmatrix} = 0
 \end{aligned} \quad \text{or} \quad \begin{aligned} &\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{m}} \cdot \bar{\mathbf{q}}_3 = 0 \\ &\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{k}} \cdot \bar{\mathbf{q}}_3 = 0 \end{aligned} \tag{6.7}$$

The system of eqs. 6.7 represents the problem of fitting a triangle to three small circles. The factors m_{ij} and k_{ij} contains the known kinematic parameters as well as the joint coordinates of the robot.

Eq. 6.7 is a non-linear system of equations coupled with respect to the two unknown angles ϑ_2, ϑ_3 containing transcendental functions. In order to eliminate these functions, a half tangent substitution (annex D.2.2) is performed on eq. 6.7:

$$\begin{aligned} \{u_2^2, u_2, 1\} \cdot \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \cdot \begin{bmatrix} u_3^2 \\ u_3 \\ 1 \end{bmatrix} &= 0 \\ \{u_2^2, u_2, 1\} \cdot \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} \cdot \begin{bmatrix} u_3^2 \\ u_3 \\ 1 \end{bmatrix} &= 0 \end{aligned} \quad \text{or} \quad \begin{aligned} \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}} \cdot \bar{\mathbf{u}}_3 &= 0 \\ \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}} \cdot \bar{\mathbf{u}}_3 &= 0 \end{aligned} \quad (6.8)$$

with the factors:

$$\begin{aligned} M_{11} &= m_{11} + m_{33} - (m_{31} + m_{13}); & M_{13} &= m_{31} + m_{33} - (m_{11} + m_{13}); \\ M_{33} &= m_{11} + m_{33} + (m_{31} + m_{13}); & M_{31} &= m_{13} + m_{33} - (m_{11} + m_{31}); \\ M_{12} &= 2(m_{32} - m_{12}); & M_{21} &= 2(m_{23} - m_{21}); \\ M_{32} &= 2(m_{32} + m_{12}); & M_{23} &= 2(m_{23} + m_{21}); \\ M_{22} &= 4m_{22} \end{aligned} \quad (6.9)$$

for the K -factors: $M \rightarrow K$ and $m \rightarrow k$

To reduce the polynomial system of eqs. 6.8 to a univariate polynomial in u_2 , the method of resultant [Annex D.1] is applied:

$$\text{Res}(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}} \cdot \bar{\mathbf{u}}_3, \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}} \cdot \bar{\mathbf{u}}_3) = \begin{vmatrix} \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i1} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i2} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i3} & 0 \\ 0 & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i1} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i2} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i3} \\ \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i1} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i2} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i3} & 0 \\ 0 & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i1} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i2} & \bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i3} \end{vmatrix} = 0 \quad (6.10)$$

where $\bar{\mathbf{M}}_{i1,3}$ and $\bar{\mathbf{K}}_{i1,3}$ are the column vectors of the matrices $\bar{\mathbf{M}}$ and $\bar{\mathbf{K}}$.

Expansion of eq. 6.10 leads to:

$$\begin{aligned} &(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{M}}_{i1}^T \cdot \bar{\mathbf{u}}_2)(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i3} \cdot \bar{\mathbf{K}}_{i3}^T \cdot \bar{\mathbf{u}}_2) + (\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i2} \cdot \bar{\mathbf{M}}_{i2}^T \cdot \bar{\mathbf{u}}_2)(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i1}^T \cdot \bar{\mathbf{u}}_2) + \\ &(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i3} \cdot \bar{\mathbf{M}}_{i3}^T \cdot \bar{\mathbf{u}}_2)(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i1}^T \cdot \bar{\mathbf{u}}_2) - (\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{M}}_{i2}^T \cdot \bar{\mathbf{u}}_2)(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i2} \cdot \bar{\mathbf{K}}_{i3}^T \cdot \bar{\mathbf{u}}_2) - \\ &(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i2} \cdot \bar{\mathbf{M}}_{i3}^T \cdot \bar{\mathbf{u}}_2)(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i2}^T \cdot \bar{\mathbf{u}}_2) + \\ &(\bar{\mathbf{u}}_2^T \cdot \bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{M}}_{i3}^T \cdot \bar{\mathbf{u}}_2)(\bar{\mathbf{u}}_2^T \cdot (\bar{\mathbf{K}}_{i2} \cdot \bar{\mathbf{K}}_{i2}^T - 2\bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i3}^T) \cdot \bar{\mathbf{u}}_2) = 0 \end{aligned} \quad (6.11)$$

Further expansion would lead to an 8th order polynomial in u_2 :

$$P_8 u_2^8 + P_7 u_2^7 + P_6 u_2^6 + P_5 u_2^5 + P_4 u_2^4 + P_3 u_2^3 + P_2 u_2^2 + P_1 u_2 + P_0 = 0 \quad (6.12)$$

The symbolical derivation of the P-factors is cumbersome and can hardly be found without a software for symbolic calculations. The whole polynomial consists of 423 summands, each of which is composed of a multiplication of four elements of the matrices $\overline{\overline{\mathbf{M}}}$ and $\overline{\overline{\mathbf{K}}}$. This explosion of the factors is well known when trying to reduce polynomial systems of equations to a univariate polynomial [Nanua 90, Innocenti 93 Husty 94]. The factors in the "general position problem" given in eq. 6.3 and 6.5 confirm this tendency to explode.

Here, a different method will be developed to symbolically derive the factors of the 8th order polynomial. Eq. 6.11 is composed of seven summands having the general form $(\overline{\mathbf{u}}^T \cdot \overline{\overline{\mathbf{G}}} \cdot \overline{\mathbf{u}})(\overline{\mathbf{u}}^T \cdot \overline{\overline{\mathbf{H}}} \cdot \overline{\mathbf{u}})$. Each of the two factors represents a 4th order polynomial written in matrix form. These matrices are pre- and postmultiplied by the *same* vector. Therefore, the matrix can be converted to a vector:

$$(\overline{\mathbf{u}}^T \cdot \overline{\overline{\mathbf{G}}} \cdot \overline{\mathbf{u}}) = \{u^2, u, 1\} \cdot \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} \cdot \begin{Bmatrix} u^2 \\ u \\ 1 \end{Bmatrix} = \{u^4, u^3, u^2, u, 1\} \cdot \begin{Bmatrix} P_4 \\ P_3 \\ P_2 \\ P_1 \\ P_0 \end{Bmatrix} = \overline{\mathbf{u}}^T \cdot \overline{\mathbf{P}}_G \quad (6.13)$$

where i of the vector $\overline{\mathbf{u}}i$ indicates the highest power of u .

The relation between the G- and the P-factors is given by the addition of the matrix elements on lines parallel to the secondary diagonal, yielding:

$$\overline{\overline{\mathbf{G}}} = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} \rightarrow \left\{ \begin{array}{c} G_{11} \\ \hline G_{21} + G_{12} \\ \hline G_{31} + G_{22} + G_{13} \\ \hline G_{32} + G_{23} \\ \hline G_{33} \end{array} \right\} = \begin{Bmatrix} P_4 \\ P_3 \\ P_2 \\ P_1 \\ P_0 \end{Bmatrix} = \overline{\mathbf{P}}_G \quad (6.14)$$

Using eq. 6.13 and 6.14 the following conversion is performed:

$$(\overline{\mathbf{u}}^T \cdot \overline{\overline{\mathbf{G}}} \cdot \overline{\mathbf{u}})(\overline{\mathbf{u}}^T \cdot \overline{\overline{\mathbf{H}}} \cdot \overline{\mathbf{u}}) = \overline{\mathbf{u}}^T \cdot \overline{\mathbf{P}}_G \cdot \overline{\mathbf{P}}_H^T \cdot \overline{\mathbf{u}} \quad (6.15)$$

The right hand side of eq. 6.15 has again a matrix form and will be converted to a vector form using a generalized equation similar to eq. 6.14, which can be considered a function applied to a matrix to extract a vector. It is therefore referred to as a one-

argument *Sh-function* capable to shrink (Sh) a matrix to a vector according to the definition given in annex B.3. Rewriting eq. 6.15 and applying two-times the Sh-function yields:

$$(\bar{\mathbf{u}}2^T \cdot \bar{\mathbf{G}} \cdot \bar{\mathbf{u}}2)(\bar{\mathbf{u}}2^T \cdot \bar{\mathbf{H}} \cdot \bar{\mathbf{u}}2) = \bar{\mathbf{u}}4^T \cdot \text{Sh}(\bar{\mathbf{G}}) \cdot \text{Sh}(\bar{\mathbf{H}})^T \cdot \bar{\mathbf{u}}4 = \bar{\mathbf{u}}8^T \cdot \text{Sh}(\text{Sh}(\bar{\mathbf{G}}) \cdot \text{Sh}(\bar{\mathbf{H}})^T) \quad (6.16)$$

The right hand side contains the factors of the 8th order polynomial.

With the help of the new Sh-function the gap between eq. 6.11 and 6.12 can now be closed by repetitive application of this blow up and shrink action:

$$\begin{pmatrix} P_8 \\ P_7 \\ P_6 \\ P_5 \\ P_4 \\ P_3 \\ P_2 \\ P_1 \\ P_0 \end{pmatrix} = \text{Sh} \begin{pmatrix} \text{Sh}(\bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{M}}_{i1}^T) \cdot \text{Sh}(\bar{\mathbf{K}}_{i3} \cdot \bar{\mathbf{K}}_{i3}^T)^T + \text{Sh}(\bar{\mathbf{M}}_{i2} \cdot \bar{\mathbf{M}}_{i2}^T) \cdot \text{Sh}(\bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i3}^T)^T + \\ \text{Sh}(\bar{\mathbf{M}}_{i3} \cdot \bar{\mathbf{M}}_{i3}^T) \cdot \text{Sh}(\bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i1}^T)^T - \text{Sh}(\bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{M}}_{i2}^T) \cdot \text{Sh}(\bar{\mathbf{K}}_{i2} \cdot \bar{\mathbf{K}}_{i3}^T)^T - \\ \text{Sh}(\bar{\mathbf{M}}_{i2} \cdot \bar{\mathbf{M}}_{i3}^T) \cdot \text{Sh}(\bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i2}^T)^T + \\ \text{Sh}(\bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{M}}_{i3}^T) \cdot \text{Sh}((\bar{\mathbf{K}}_{i2} \cdot \bar{\mathbf{K}}_{i2}^T - 2\bar{\mathbf{K}}_{i1} \cdot \bar{\mathbf{K}}_{i3}^T))^T \end{pmatrix} \quad (6.17)$$

The introduction of this new function was initiated by observing a repetitive appearance of the same pattern when trying to reduce 6.11 to eq. 6.17. Expansion of eq. 6.17 would lead to the same 423 summands already mentioned. The question arises about the fastest way to calculate the P-factors. A fully expanded version of eq. 6.17 consists of 1269 multiplications and 414 additions. By introducing the Sh(Shrink)-function the number of operations could be reduced to 261 multiplications and 198 additions. However, the fastest possible way to calculate the P-factors is by directly substituting the numerical values into eq. 6.10 and collecting the terms for all powers of u_2 . Then, the speed of calculation is mainly determined by the collecting operation.

After calculating the P-factors (eq. 6.17), the 8th order polynomial of eq. 6.11 can be solved numerically for u_2 . Some of the solutions may be complex. Only the real solutions (n) have physical meaning. It remains the back-substitution in eq. 6.8 and to solve for u_3 :

The solution of the "general orientation problem" can be summarized as:

$$\begin{aligned}
& \sum_{j=0}^8 P_j u_2^j = 0 \rightarrow u_{2_k} \in \mathcal{R} ; \text{for } k=0..n \leq 8 \\
\rightarrow & \vartheta_{2_k} = 2 \arctan(u_{2_k}); \\
& \vartheta_{3_k} = 2 \arctan \left(\frac{\bar{\mathbf{u}}_{2_k}^T \cdot (\bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{K}}_{i3}^T - \bar{\mathbf{M}}_{i3} \cdot \bar{\mathbf{K}}_{i1}^T) \cdot \bar{\mathbf{u}}_{2_k}}{\bar{\mathbf{u}}_{2_k}^T \cdot (\bar{\mathbf{M}}_{i2} \cdot \bar{\mathbf{K}}_{i1}^T - \bar{\mathbf{M}}_{i1} \cdot \bar{\mathbf{K}}_{i2}^T) \cdot \bar{\mathbf{u}}_{2_k}} \right)
\end{aligned}
\quad
\begin{aligned}
& M_{1..3,1..3}, K_{1..3,1..3} \rightarrow 6.9 \\
& P_{0..8} \rightarrow 6.17
\end{aligned}
\quad (6.18)$$

It has eight sets of solution $(\vartheta_2, \vartheta_3)$ at most. As already mentioned, its geometric interpretation is a problem generally encountered in spherical kinematics [Chiang 92] of fitting a spherical triangle to three small circles. It is interesting to compare to the equivalent planar problem, which corresponds to fitting a plane triangle to three circles or three lines. Both cases are very well explored and have six and two solutions, respectively [Merlet 90, Hunt 78, Dijksman 76]. Another remark concerns the geometric interpretation of the "general position problem", that was shown in paragraph 6.2.2 to be a problem of intersection of three spheres in space, which is a symmetrical interpretation. The orientation problem could only be solved with an asymmetrical parameterization (paragraph 3.5.2). For an asymmetrical interpretation of the position problem one of the three spheres will be selected and the remaining two represented by their intersection circles with this sphere. Thus, the position problem can also be interpreted as a problem of spherical kinematics to find the intersection point of two small circles. This problem can therefore also be parameterized using the two coordinates needed to describe the unknown intersection point on the sphere.

6.2.4 The general pose problem

According to Wampler [Husty 94] the "general pose problem" can always be stated with the help of eight homogeneous world coordinates describing the pose of the end-effector $(\bar{\mathbf{X}})$ in figure 6.1 as:

$$\bar{\mathbf{X}}^T \cdot \bar{\mathbf{M}}_i \cdot \bar{\mathbf{X}} = 0$$

with

$$\bar{\mathbf{X}} = \{X_1, X_2, X_3, X_4, \tilde{X}_1, \tilde{X}_2, \tilde{X}_3, \tilde{X}_4\}^T \quad i=1..7 \quad (6.19)$$

$$\bar{\mathbf{M}}_i = \begin{bmatrix} \bar{\mathbf{A}}_i & \bar{\mathbf{B}}_i \\ \bar{\mathbf{B}}_i^T & \bar{\mathbf{C}}_i \end{bmatrix}$$

where the 4x4 blocks $\bar{\mathbf{A}}_i, \bar{\mathbf{B}}_i$ are symmetrical and skew-symmetric, respectively and contain the design parameters of the robot as well as the joint coordinates. The $\bar{\mathbf{C}}_i$ -

block is the 4x4 identity matrix. The seventh equation is free of design parameters and the following relation is valid:

$$\begin{aligned}\overline{\overline{\mathbf{A}}_7} = \overline{\overline{\mathbf{C}}_7} = \overline{\overline{\mathbf{0}}}: & \quad 4 \times 4 \text{ Zero matrix} \\ \overline{\overline{\mathbf{B}}_7} = 1/2 \overline{\overline{\mathbf{I}}}: & \quad 4 \times 4 \text{ Identity matrix}\end{aligned}\tag{6.20}$$

The system of equations 6.19 represents the problem of fitting a irregular hexahedron to six spheres. Wampler [96] has shown that the number of solutions drops from 128 for the general case to 40 due to the special structure of the matrices $\overline{\overline{\mathbf{M}}_i}$. This result was also found by Raghavan [93] using numerical techniques.

Wampler and Husty found simultaneously, that the "general pose problem" can be described with seven coupled quadratic equations as given in eq. 6.19. Its derivation is based on the mapping of spatial movements in a higher dimensional space. A movement is the set of poses which a moving body attains. For planar movements this image space is three dimensional and Euclidean [Bottema 79, Ravani 83]. For spatial movement Study introduced a seven dimensional, quasi-elliptical image space called "Soma"-space [Ravani 84, Husty 94]. A point in the Soma space has eight homogeneous coordinates given by a dual quaternion. A dual quaternion [Castelain 86, Milos 95] is composed of quaternions and dual-numbers. Quaternions are an extension of the complex numbers and therefore sometimes also called hypercomplex numbers. They were introduced by Hamilton to describe rigid body rotations in space with the help of Euler parameters [Branets 73, Funda 88, de Casteljau 87]. Dual-numbers were introduced by Clifford and are well suited to describe a rotation about and a translation along a vector, which results in a movement of a screw along this vector when combined [Dimentberg 65]. This leads to a different point of view for eq. 6.19:

Each possible pose in space is represented by a unique point on a six-dimensional hypersurface embedded in the seven dimensional Soma space. According to Husty this hypersurface is called "Study's quadric". Study's quadric is given by the seventh quadratic equation in the system of eqs. 6.19 together with the definition in eq. 6.20.

The other six equations are six-dimensional hypersurfaces, too. They represent a movement of a constraint rigid body. The constraint results from a point of the moving body being restricted to travel on the surface of a sphere. This corresponds to cutting off five of the six connecting rods between the base and the end-effector (fig. 6.1). Wampler and Husty found a way to represent a constraint hypersurface fulfilling this spherical condition. In planar kinematics this corresponds to a movement of a rigid

body constraint, where one of its points stays on a circle. For this case Bottema [79] has given the constraint surface in the three dimensional image space.

It remains to explain the exact meaning of the vector of unknowns $\bar{\mathbf{X}}$ in eq. 6.20. This vector is composed of the eight parameters defining a dual quaternion. This dual quaternion contains the coordinates of the unknown intersection point of the six hypersurfaces located on Study's quadric.

In paragraph 3.3.2 a much simpler parameterization for the same geometrical problem was chosen, leading to a system of six equations for six unknowns. Unfortunately, this system of equations is only useful for numerical solving. To extract a univariate polynomial it seems to be a dead end which misguided researcher over many years. It's a very characteristic difficulty in kinematics that the choice of the parameterization creates a system of closure equations which is either reducible to a univariate polynomial of minimal degree or not. This was also the motivation for Wampler and Husty to derive eq. 6.19, because it can be reduced to a univariate polynomial as shown by Husty. He introduced the following substitution of variables as:

$$\bar{\mathbf{X}} = \{k, m, p, 1\}^T = \left\{ \frac{X_1}{X_4}, \frac{X_2}{X_4}, \frac{X_3}{X_4}, 1 \right\}^T \quad (6.21)$$

The dual part will also be collected in a vector $\bar{\tilde{\mathbf{X}}} = \{\tilde{X}_1, \tilde{X}_2, \tilde{X}_3, \tilde{X}_4\}^T$. Substitution of both vectors into eq. 6.19 leads to:

$$\left\{ 2\bar{\mathbf{X}}^T \cdot \bar{\mathbf{B}}_i, \bar{\mathbf{X}}^T \cdot \bar{\mathbf{A}}_i \bar{\mathbf{X}} \right\} \cdot \begin{Bmatrix} \bar{\tilde{\mathbf{X}}} \\ X_4 \end{Bmatrix} = -\frac{\bar{\tilde{\mathbf{X}}}^T \cdot \bar{\mathbf{C}}_i \cdot \bar{\tilde{\mathbf{X}}}}{X_4} \quad i=1..7 \quad (6.22)$$

The right hand side is free of design parameters and any difference between the first six equations will therefore be linear in the five variables collected in the column vector on the left hand side. The situation is similar to that of the "general position problem". The seventh equation is inherently linear in these five variables because its right hand side is equal to zero. The resulting system of five differences plus the seventh equation is overdetermined and must be made linearly dependent in order to obtain non-trivial solutions for the five variables. On the one hand side this leads to the condition that the determinants of two 5x5 subsystems of differences plus the seventh equation must be equal zero. On the other hand, one of these two subsystems of equations can be linearly solved for the five variables and back-substituted into one of the constraint surfaces given in 6.19. Both steps lead to polynomials containing only the three unknowns given in $\bar{\mathbf{X}}$. These polynomials can further be treated with the method of resultants (Annex

D.1) yielding two univariate polynomials of 200th order. Their greatest common divisor is a polynomial of 40th order. Due to the already mentioned symbolic explosion (paragraph 6.2.3) the factors of this polynomial are only available in a numerical form.

Husty's algorithm was applied to solve the direct problem of the calibration model 54 in order to investigate how close other solutions approach the two solutions, which are parallel to the base (fig. 6.3). This is an important question if the direct solution should be taken for identification (chapter 5) as for serial robots. Numerical methods are employed to obtain the world coordinates of model 54 using the direct solution of model 24 as starting values. If another solution approaches closely the searched solution, the numerical algorithm will once find the one solution, but slight variation of the design parameters will lead to the other closely located solution. This causes severe problems during identification due to the introduction of this solution jumping. However, it could be shown that other solutions approach closely the parallel solution only in a very small region around the geometric singularities of the parallel structure. Otherwise they stay in orientation far away of the searched solution. Far from singularities the convergence of the numerical algorithms to the searched solution can therefore be guaranteed.

Figure 6.3 shows six poses of the end-effector of a Delta robot without geometric deviations, following a linear trajectory approaching a singularity located in the lower right corner. It shows two solutions obtained with Husty's algorithm. One is the searched solution, where the end-effector remains parallel. The other one possibly disturbs the numerical convergence to the parallel solution. In the lower right corner the robot is very close to a type c) singularity one of four different types of direct singularities [Fichter 86], which occurs for the Delta robot [Clavel 91]. A direct singularity is a bifurcation of solutions of the direct problem, thus for the same set of joint angles two or more coincident solutions exist. At this singularity the structure will gain as many degrees of freedom as solutions join the parallel solution. Moving out of the singularity these solutions - for the chosen singularity type c) there are two - will start to separate. The development of both solutions was followed in order to see how fast they separate. This shows that for the last set of joint angles the disturbing solutions has already left its working volume. At singularities the mechanisms can be forced to change from one configuration space into another one. It would be of interest to find a parallel structure with physically interesting configurations on both sides of a direct singularity, since this robot could work task dependent in its first as well as in its second configuration. This would be a powerful sales argument for such kind of parallel robot like selling one robot for two.

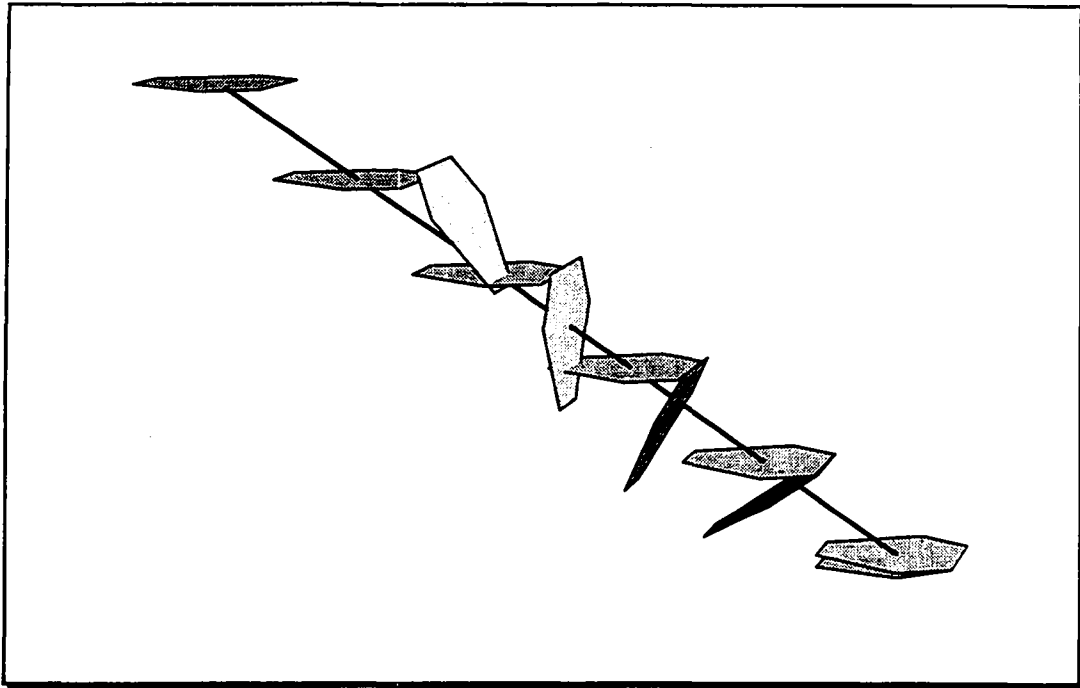


Fig. 6.3: Movement of the end-effector for a simulation of the Delta robot approaching a singularity of type c). Solutions of two configuration spaces are shown.

For the sake of completeness it should be mentioned that a second type of singularities occurs if solutions of the inverse problem join each other [Reboulet 88, Gosselin 88]. In this case the mechanism loses as many degrees of freedom as solutions coincide with the searched solution of the inverse problem. This is therefore referred to as inverse singularity [Guglielmetti 94]. Figure 6.3 shows only two of the solutions obtained with Husty's algorithm for the Delta robot. In fact, between 8 and 16 real solutions were found forming clusters around the two solutions obtained by solving the "general position problem" for a perfect remaining spatial parallelogram (model 24). In annex A.2 an entire set of such solutions (model 54) is given for a single set of joint angles.

6.2.5 The general joint angle problem

According to Pierrot [90] the "general joint angle problem" can be described separately for each R2S joint-link train:

$$a_i \cos(\theta_i) + b_i \sin(\theta_i) = c_i \quad (6.23)$$

with the factor a, b, c depending only on known parameters (geometry, pose of the end-effector, encoder offset) of the robot.

Eq. 6.23 represents the problem of finding the intersections of a circle with a sphere. The transcendental eq. 6.23 in the unknown joint angle θ_i is of a well known type [Paul 81, Craig 89] and its solution can either be found by a half-tangent substitution or by a circular substitution. Care has to be taken not to introduce mathematical singularities [Codourey 91, Clavel 91]. For that reason it is recommended to use the circular substitution [Vischer 95] leading to the following result:

$$\theta_{i,2} = \arctan 2 \left(c_i, \pm \sqrt{b_i^2 + a_i^2 - c_i^2} \right) - \arctan 2(a_i, b_i) \quad i=1..3 \quad (6.24)$$

The "general joint angle problem" is solved by eq. 6.24. It has no or two solutions for each joint-link train.

6.2.6 Conclusions

In this section the solutions for three direct problems, the "general position, orientation and pose problem", has been presented. Note the fast increasing difficulties upon going from the first to the last of these problems. The solution of the "general pose problem" can be interesting to explore a model as done for model 54 of the Delta robot, but it is much too slow for a real-time application. The inverse "general joint angle problem" is quite simple compared to the solution of any of the direct models. Both, the "general joint angle problem" as well as the "general position problem", can be interpreted as the intersection problem of a circle with a sphere. This means that the underlying equation of both problems is given by eq. 6.23.

6.3 Solving model 24 of the Delta robot

6.3.1 The direct problem

The solution of the direct problem of model 24 provides for a set of joint angles no, one or several sets of world coordinates: $\theta_1, \theta_2, \theta_3 \rightarrow x, y, z$. By comparing figure 3.11 with figure 6.1 it's easy to see that the direct problem of model 24 given in eq. 3.16 is a "general position problem" (paragraph 6.2.2). Thus, eq. 3.16 will be rewritten as:

$$\begin{aligned} \bar{\mathbf{P}}^T \cdot \bar{\mathbf{P}} - 2\bar{\mathbf{C}}_i^T \cdot \bar{\mathbf{P}} + \bar{\mathbf{C}}_i^T \cdot \bar{\mathbf{C}}_i &= Lb^2 \\ \text{with} & \\ \bar{\mathbf{C}}_i &= \bar{\mathbf{T}}_i \cdot (\bar{\mathbf{D}}_i + \bar{\mathbf{Q}}_i \cdot \bar{\mathbf{L}}\mathbf{a}_i) \end{aligned} \quad i=1..3 \quad (6.25)$$

Collecting factors E, F, G, H given in the definition of the "general position problem" (eq. 6.1) yields:

$$\begin{aligned} \{E_i, F_i, G_i\} &= -2 \left(\overline{\overline{\mathbf{T}}}_i \cdot \left(\overline{\mathbf{D}}_i + \overline{\overline{\mathbf{Q}}}_i \cdot \overline{\mathbf{L}\mathbf{a}}_i \right) \right)^T, \\ H_i &= \overline{\mathbf{D}}_i^T \cdot \overline{\mathbf{D}}_i + 2 \overline{\mathbf{D}}_i^T \cdot \overline{\overline{\mathbf{Q}}}_i \cdot \overline{\mathbf{L}\mathbf{a}}_i + \overline{\mathbf{L}\mathbf{a}}_i^T \cdot \overline{\mathbf{L}\mathbf{a}}_i - Lb^2 \end{aligned} \quad i=1..3 \quad (6.26)$$

Substituting these factors into eq. 6.1 the solution of the direct problem of model 24 is given by eq. 6.6 having two sets of solutions at most.

6.3.2 The inverse problem

The solution of the inverse problem provides for a set of world coordinates no, one or several sets of joint angles: $x, y, z \rightarrow \theta_1, \theta_2, \theta_3$. Model 24 given by eq. 3.16 is decoupled in the unknown variables $\theta_1, \theta_2, \theta_3$, which is typical for parallel robots. This simplifies the solution enormously since each of the non-linear eq. of 3.16 representing a R2S joint-link train can be treated separately. This is exactly the "general joint angle problem" treated in the last section. Expanding eq. 3.16 and collecting the rotation matrix $\overline{\overline{\mathbf{Q}}}_i$ containing the unknown joint angle leads to:

$$2 \left(\overline{\mathbf{P}}^T \cdot \overline{\overline{\mathbf{T}}}_i + \overline{\mathbf{D}}_i^T \right) \cdot \overline{\overline{\mathbf{Q}}}_i \cdot \overline{\mathbf{L}\mathbf{a}}_i + \overline{\mathbf{P}}^T \cdot \overline{\mathbf{P}} - 2 \overline{\mathbf{P}}^T \cdot \overline{\overline{\mathbf{T}}}_i \cdot \overline{\mathbf{D}}_i + \overline{\mathbf{D}}_i^T \cdot \overline{\mathbf{D}}_i + \overline{\mathbf{L}\mathbf{a}}_i^T \cdot \overline{\mathbf{L}\mathbf{a}}_i = Lb^2 \quad i=1..3 \quad (6.27)$$

Vector $\overline{\overline{\mathbf{Q}}}_i \cdot \overline{\mathbf{L}\mathbf{a}}_i$ can be rewritten as:

$$\overline{\overline{\mathbf{Q}}}_i \cdot \overline{\mathbf{L}\mathbf{a}}_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 \\ s\theta_i & c\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} L\mathbf{a}_{ix} \\ L\mathbf{a}_{iy} \\ 0 \end{bmatrix} = \begin{bmatrix} L\mathbf{a}_{ix} & -L\mathbf{a}_{iy} & 0 \\ L\mathbf{a}_{iy} & L\mathbf{a}_{ix} & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} c\theta_i \\ s\theta_i \\ 1 \end{bmatrix} = \overline{\mathbf{L}\mathbf{a}}_i \cdot \overline{\mathbf{Q}}_i \quad i=1..3 \quad (6.28)$$

This procedure of swapping between a rotation matrix and a vector is repeatedly used in this work and has therefore been defined as a two-argument "Sw"(Swap)-function (cf. annex B.2). It contains the unit-vector on the rotation axis as a first argument and the vector to be swapped as second argument. Eq. 6.28 can therefore be rewritten as:

$$\overline{\overline{\mathbf{Q}}}_i \cdot \overline{\mathbf{L}\mathbf{a}}_i = \text{Sw}(z, \overline{\mathbf{L}\mathbf{a}}_i) \cdot \overline{\mathbf{Q}}_i \quad i=1..3 \quad (6.29)$$

Collecting factors a,b,c given in the definition of the "joint angle problem" in eq. 6.23 yields:

$$\begin{aligned} \{a_i, b_i, c_i\} &= 2 \left(\overline{\mathbf{P}}^T \cdot \overline{\overline{\mathbf{T}}}_i + \overline{\mathbf{D}}_i^T \right) \cdot \text{Sw}(z, \overline{\mathbf{L}\mathbf{a}}_i) \\ c_i &= Lb^2 - e_i + 2 \overline{\mathbf{P}}^T \cdot \overline{\overline{\mathbf{T}}}_i \cdot \overline{\mathbf{D}}_i - \overline{\mathbf{P}}^T \cdot \overline{\mathbf{P}} - \overline{\mathbf{D}}_i^T \cdot \overline{\mathbf{D}}_i - \overline{\mathbf{L}\mathbf{a}}_i^T \cdot \overline{\mathbf{L}\mathbf{a}}_i \end{aligned} \quad i=1..3 \quad (6.30)$$

The solution of the inverse problem of model 24 is given by substituting the factors defined in eq. 6.30 into eq. 6.24. This leads to two solutions per joint-link train at most, that is eight distinguishable configurations.

6.4 Solving model 54 of the Delta robot

6.4.1 Introduction

Model 54 is given in eq. 3.12 together with table 3.10. The solution of the direct problem provides for a given set of joint angles no, one or several sets of world coordinates $\theta_1, \theta_2, \theta_3 \rightarrow x, y, z, \alpha, \beta, \gamma$, whereas the solution of the inverse problem for a given end-effector position has no, one or several sets of joint and orientation angles $x, y, z \rightarrow \theta_1, \theta_2, \theta_3, \alpha, \beta, \gamma$. Examining the system of eqs. 3.12 shows that the direct problem is represented by a system of six non-linear equations coupled with respect to all six unknowns, which is characteristic for parallel robots. New is, however, that the inverse problem gets coupled, too. Each unknown joint angle appears in both of the equations representing a main joint-link train, whereas the unknown fixed angles describing the orientation of the end-effector appear in all of the six equations.

For the direct problem, which represents a "general pose problem" (paragraph 6.2.4), the reduction to a univariate polynomial leads to a 40th order polynomial. For the inverse problem the order of the polynomial, which indicates the non-linearity of the underlying system of equations, is still an open question. However, an indication might be obtained by linearizing the rotation matrix for small angles. This corresponds to the assumption that changes of the end-effector orientation are small with respect to the base when sufficiently far away of singularities. In this case the order of the direct problem drops to ten whereas the order of the inverse problem is below 256. This is an indication for the *inverse problem* being *more non-linear* and thus even *more difficult* to solve *than the direct problem*, one of the most complicated, polynomial solution known today.

This approach will not be further discussed. Two different approaches are introduced instead. Both approaches will be introduced since none of them satisfies both of the given constraints, namely fast solvability and good fitting of the full model 54. For the implementation of the calibrated model 54 one of the two approaches can be chosen depending on the hardware of the controller.

The first approach is based on a linearization of the entire set of the six closure equations (G) by building the Jacobian (JG), whereas the second approach uses the splitting of the closure equations into set G1 and G2:

simplifications	algorithm	DP	IP	Flow Charts of the different algorithms (representation of the solution of DP only)
Linearization around the entry point	linearized iterative	2	2^3	
Substitution of the coupling terms G1(P, R*) G2(P*, R)	cascaded iterative	2×8	$2^3 \times 8$	

Table 6.4: Algorithms proposed for the solution of the direct (DP) and the inverse problem (IP) of model 54. Given are the number of solutions, which can be produced by the algorithms depending on the choice of the starting values or on the solution forwarded to the next set of equations.

The first algorithm is the widely used Newton-Raphson procedure. As mentioned earlier it works very well for that kind of problem. The solution is generally found within three iterations. A good set of starting values (P^* , Q^*) can be gained from the nominal model or from model 24, whereas R^* can be set to zero. To accelerate the time of calculation Schwarz [86] proposed to drop the time-consuming updating of the Jacobian in each iteration step using instead the Jacobian calculated for a "good" set of starting values. Another possibility is given by skipping iteration, taking into account small errors of the result. This is of interest because of the 6x6 Jacobian getting relatively easy to invert formally for the initial guess ($R^*=0$):

$$\{P, R\} \approx \{P^*, 0\} - JG(P^*, 0)^{-1} G(P^*, 0) \quad (6.31)$$

The second algorithm is based on the fact that the first set of eqs. G1 of the direct problem is reduced to a "general position problem" for a given orientation (paragraph 6.2.2), whereas for a given position the second set of eqs. G2 is reduced to a "general orientation problem" (paragraph 6.2.3). For the inverse problem the first set is reduced to a "general joint angle problem" (paragraph 6.2.5) once the orientation is known, whereas the second set is reduced to a "general orientation problem", too, once the joint angles are known. Due to the complexity of the orientation problem it might be better to

solve set G2 using a Newton-Raphson procedure or to linearize the rotation matrix for small angles:

$$\bar{\mathbf{R}} = \begin{bmatrix} c\beta c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ c\beta s\gamma & s\alpha s\beta s\gamma + c\alpha c\gamma & c\alpha s\beta s\gamma - s\alpha c\gamma \\ -s\beta & s\alpha c\beta & c\alpha c\beta \end{bmatrix} \rightarrow \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix} = \Delta \bar{\mathbf{R}} \quad (6.32)$$

This matrix is well known in robotics. It is the sum of the identity matrix and a skew symmetric matrix [Paul 81]. $\Delta G2$ indicates that the rotation matrix of the set G2 is linearized according to eq. 6.32. $\Delta G2$ can be linearly solved for the unknown orientation angles. In order to reduce the time of calculation, this algorithm can be used without iteration, too. However, the result is not satisfactory since the first step is mainly needed to get to the solution of the nominal problem. A final proposition for the creation of a very fast, but rough cascaded solution is to drop all terms of magnitude Δ in the first set G1 and all of magnitude Δ^2 in the second set $\Delta G2$, which reduces sets G1 and $\Delta G2$ to:

$$\begin{aligned} \text{set1: } \bar{\mathbf{C}}\mathbf{B}_i^T \cdot \bar{\mathbf{C}}\mathbf{B}_i &= Lb_i^2 \\ \text{set2: } \bar{\mathbf{C}}\mathbf{B}_i^T \cdot \Delta \bar{\mathbf{d}}_i &= Lb_i * \Delta Lb_i \\ \text{with} & \\ \bar{\mathbf{C}}\mathbf{B}_i &= \bar{\mathbf{P}} - \bar{\mathbf{T}}_i \cdot (\bar{\mathbf{D}}_i + \bar{\mathbf{Q}}_i \cdot \bar{\mathbf{L}}\mathbf{a}_i) \\ \Delta \bar{\mathbf{d}}_i &= \Delta \bar{\mathbf{R}} \times \bar{\mathbf{T}}_i \cdot \bar{\mathbf{d}}_i - \bar{\mathbf{T}}_i \cdot (\Delta \bar{\mathbf{T}}_i \times \bar{\mathbf{d}}_i + \Delta \bar{\mathbf{T}}_i \cdot \bar{\mathbf{Q}}_i \cdot \Delta \bar{\mathbf{C}}_i) \end{aligned} \quad (6.33)$$

with the matrices, vectors and scalars given according to table 3.10. $\Delta \bar{\mathbf{R}}, \Delta \bar{\mathbf{T}}$ are column vectors containing small rotation angles (see also eq. 6.38)

The number of parameters involved in eq. 6.33 is reduced to 45 due to the vanished $\bar{\mathbf{b}}_i$ -vector. The first set is identical to model 24 and can be solved as described in the section 6.3, whereas the second set gives a rough approximation of the orientation of the end-effector and the solution similar to solution of the set $\Delta G2$ for the orientation, treated in this section.

The first algorithm can be directly applied to eq. 3.9 or 3.12. For the second algorithm solutions have to be provided for the direct and inverse problem of set G1 if the orientation is known, and for the orientation problem of set $\Delta G2$ if the position respectively joint angle are given as described in the following section.

6.4.2 The direct problem of the set of equations G1

Expanding the system of equations 3.12 and collecting the unknown position vector $\bar{\mathbf{P}}$ leads to:

$$\begin{aligned} \bar{\mathbf{P}}^T \cdot \bar{\mathbf{P}} + 2\bar{\mathbf{v}}_i^T \cdot \bar{\mathbf{P}} + \bar{\mathbf{v}}_i^T \cdot \bar{\mathbf{v}}_i + \Delta\bar{\mathbf{d}}_i^T \cdot \Delta\bar{\mathbf{d}}_i &= Lb^2 + \Delta Lb^2 \\ \text{with} \\ \bar{\mathbf{v}}_i &= \bar{\mathbf{R}}^* \cdot \bar{\mathbf{T}}_i \cdot \bar{\mathbf{b}}_i - \bar{\mathbf{T}}_i \cdot (\bar{\mathbf{b}}_i + \bar{\mathbf{D}}_i + \Delta\bar{\mathbf{T}}_i \cdot \bar{\mathbf{Q}}_i \cdot \bar{\mathbf{L}}\mathbf{a}_i) \\ \Delta\bar{\mathbf{d}}_i &= \bar{\mathbf{R}}^* \cdot \bar{\mathbf{T}}_i \cdot \bar{\mathbf{d}}_i - \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{T}}_i \cdot (\bar{\mathbf{d}}_i + \bar{\mathbf{Q}}_i \cdot \Delta\bar{\mathbf{C}}_i) \end{aligned} \quad i=1..3 \quad (6.34)$$

where $\bar{\mathbf{R}}^*$ indicates that the orientation of the end-effector is given. The factors of the input equations of the "general position problem" (eq. 6.1) are given by:

$$\begin{aligned} \{E_i, F_i, G_i\} &= 2\bar{\mathbf{v}}_i^T, \\ H_i &= \bar{\mathbf{v}}_i^T \cdot \bar{\mathbf{v}}_i + \Delta\bar{\mathbf{d}}_i^T \cdot \Delta\bar{\mathbf{d}}_i - (Lb^2 + \Delta Lb^2) \end{aligned} \quad i=1..3 \quad (6.35)$$

The solution of the direct problem of the G1 set is given by substitution of the factors given in eq. 6.35 into eq. 6.6. This leads to two solutions at most.

6.4.3 The inverse problem of the set of equations G1

Expanding the system of equations 3.12 and collection for the unknown joint angles contained in the rotation matrices $\bar{\mathbf{Q}}_i$ leads to:

$$\begin{aligned} -2(\bar{\mathbf{v}}_i \cdot \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{T}}_i \cdot \bar{\mathbf{Q}}_i \cdot \bar{\mathbf{L}}\mathbf{a}_i + \bar{\mathbf{w}}_i \cdot \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{T}}_i \cdot \bar{\mathbf{Q}}_i \cdot \Delta\bar{\mathbf{C}}_i) + \\ \bar{\mathbf{v}}_i^T \cdot \bar{\mathbf{v}}_i + \bar{\mathbf{w}}_i^T \cdot \bar{\mathbf{w}}_i + \bar{\mathbf{L}}\mathbf{a}_i^T \cdot \bar{\mathbf{L}}\mathbf{a}_i + \Delta\bar{\mathbf{C}}_i^T \cdot \Delta\bar{\mathbf{C}}_i &= Lb_i^2 + \Delta Lb_i^2 \\ \text{with} \end{aligned} \quad i=1..3 \quad (6.36)$$

$$\bar{\mathbf{v}}_i = \bar{\mathbf{P}} + \bar{\mathbf{R}}^* \cdot \bar{\mathbf{T}}_i \cdot \bar{\mathbf{b}}_i - \bar{\mathbf{T}}_i \cdot (\bar{\mathbf{b}}_i + \bar{\mathbf{D}}_i)$$

$$\bar{\mathbf{w}}_i = \bar{\mathbf{R}}^* \cdot \bar{\mathbf{T}}_i \cdot \bar{\mathbf{d}}_i - \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{T}}_i \cdot \bar{\mathbf{d}}_i$$

where $\bar{\mathbf{R}}^*$ indicates again that the orientation of the end-effector is given. Collecting the factors given in the input equations of the "general joint angle problem" (eq. 6.23) leads to:

$$\begin{aligned} \{a_i, b_i, e_i\} &= -2(\bar{\mathbf{v}}_i \cdot \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{T}}_i \cdot \text{Sw}(z, \bar{\mathbf{L}}\mathbf{a}_i) + \bar{\mathbf{w}}_i \cdot \bar{\mathbf{T}}_i \cdot \Delta\bar{\mathbf{T}}_i \cdot \text{Sw}(z, \Delta\bar{\mathbf{C}}_i)) \\ c_i &= Lb_i^2 + \Delta Lb_i^2 - (e_i + \bar{\mathbf{v}}_i^T \cdot \bar{\mathbf{v}}_i + \bar{\mathbf{w}}_i^T \cdot \bar{\mathbf{w}}_i + \bar{\mathbf{L}}\mathbf{a}_i^T \cdot \bar{\mathbf{L}}\mathbf{a}_i + \Delta\bar{\mathbf{C}}_i^T \cdot \Delta\bar{\mathbf{C}}_i) \end{aligned} \quad i=1..3 \quad (6.37)$$

The Sw-function is defined in annex B.2, provides a place swapping between a vector and a rotation matrix. The inverse problem of the G1 set is now solved by substitution

of the factors given in eq. 6.37 into eq. 6.24 having no or two solutions per joint-link train.

6.4.4 The orientation problem of the set of equations $\Delta G2$

By postmultiplying the linearized rotation matrix given in eq. 6.32 with an arbitrary vector the following relation holds:

$$\Delta \bar{\bar{\mathbf{R}}} \cdot \bar{\mathbf{v}} = \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & -\gamma & \beta \\ \gamma & 0 & -\alpha \\ -\beta & \alpha & 0 \end{bmatrix} \right) \cdot \begin{Bmatrix} v_x \\ v_y \\ v_z \end{Bmatrix} = \begin{Bmatrix} v_x \\ v_y \\ v_z \end{Bmatrix} + \begin{Bmatrix} \alpha \\ \beta \\ \gamma \end{Bmatrix} \times \begin{Bmatrix} v_x \\ v_y \\ v_z \end{Bmatrix} = \bar{\mathbf{v}} + \Delta \bar{\bar{\mathbf{R}}} \times \bar{\mathbf{v}} \quad (6.38)$$

where \times represents a cross product.

Substituting eq. 6.38 into the second set of equations of model 54 given in 3.12 and collecting the vector $\Delta \bar{\bar{\mathbf{R}}}$, which contains the three unknown orientation angles, yields:

$$\Delta G2: \left(\bar{\bar{\mathbf{T}}}_i \cdot \bar{\mathbf{b}}_i \times \bar{\mathbf{w}}_i + \bar{\bar{\mathbf{T}}}_i \cdot \bar{\mathbf{d}}_i \times \bar{\mathbf{v}}_i \right)^T \cdot \Delta \bar{\bar{\mathbf{R}}} + \bar{\mathbf{v}}_i^T \cdot \bar{\mathbf{w}}_i = Lb_i * \Delta Lb_i$$

with

$$\bar{\mathbf{v}}_i = \bar{\mathbf{P}}^* - \bar{\bar{\mathbf{T}}}_i \cdot (\bar{\mathbf{D}}_i + \Delta \bar{\bar{\mathbf{T}}}_i \cdot \bar{\bar{\mathbf{Q}}}_i^* \cdot \bar{\mathbf{L}}\mathbf{a}_i) \quad i=1..3 \quad (6.39)$$

$$\bar{\mathbf{w}}_i = \bar{\bar{\mathbf{T}}}_i \cdot (\bar{\mathbf{d}}_i - \Delta \bar{\bar{\mathbf{T}}}_i \cdot (\bar{\mathbf{d}}_i + \bar{\bar{\mathbf{Q}}}_i^* \cdot \Delta \bar{\mathbf{C}}_i))$$

where $\bar{\mathbf{P}}^*$ and $\bar{\bar{\mathbf{Q}}}^*$ indicate that either the position or the joint angles are imposed.

The system of equations given in 6.39 can be linearly solved for the unknown joint angles, which leads to a unique solution for the orientation problem of set $\Delta G2$.

6.4.5 Conclusion

The second algorithm reaches its fixed point typically after three iterations, too, but an error remains due to the linearization of the second set $G2$ to $\Delta G2$. Compared with the first algorithm the second one needs fewer iteration steps to get very close to the solution. One and a half iteration steps (two times evaluation of $G1$, once evaluation of $\Delta G2$) are in general enough to get sufficiently close to the fixed point.

6.5 Solving the nominal model of the Argos structure

6.5.1 The direct problem

The nominal model is given by eq. 3.26 and table 3.18. For a set of given joint angles the solution of the direct problem supplies no, one or several sets of Euler angles $\alpha_1, \alpha_2, \alpha_3 \rightarrow \vartheta_1, \vartheta_2, \vartheta_3$. Expanding the first of the equations given in 3.26 leads to:

$$s(\vartheta_2) * (s(\vartheta_1) * c(\alpha_1 + \alpha_{01}) - c(\vartheta_1) * s(\alpha_1 + \alpha_{01})) = 0 \quad (6.40)$$

where c and s are abbreviations for the cosine and the sinus functions. For simplicity the given joint angle and the encoder offset will be collected in a single variable:

$$\alpha_{0i} = \alpha_i + \alpha_{0i} \quad i=1..3 \quad (6.41)$$

Rewriting eq. 6.40 yields:

$$s(\vartheta_2) * s(\vartheta_1 - \alpha_{01}) = 0 \quad (6.42)$$

To satisfy this equation the first or the second factor can be zero. The first factor being zero leads to the solution of $\vartheta_{2,1,2} = 0, \pi$, which are independent of the motor angles. This corresponds to a direct singularity if the S-joint is located on the first motor axis. In this case the first pantograph can be freely twisted without influencing the orientation of the end-effector. This kind of singularity can also be found for the other two pantographs [Gosselin 95]. The second factor being zero leads to the solutions of $\vartheta_{1,2} = \alpha_{01}, \alpha_{01} + \pi$. Due to the chosen parameterization, the first two Euler angles ϑ_1, ϑ_2 correspond to the angles of the first two joints (motor and passive R-joint) of the first joint-link train. They describe the location of the S-joint fixed to the first pantograph. If both by angle can vary in a full-circle interval, this description becomes redundant. Thus, set $\alpha_{01} + \pi, -\vartheta_2$ would describe the same position of the S-joint as α_{01}, ϑ_2 . The second solution can therefore be dropped:

$$\vartheta_1 = \alpha_{01} \quad (6.43)$$

According to this equation the first Euler angle needed to describe the orientation of the end-effector is equal to the angle of the first motor plus its offset. The *entire* parameterization is chosen in such a way that eq. 6.43 gets valid yielding this simple equation.

Expansion of the remaining two equations of the nominal model given by eq. 3.26 and table 3.18 leads to:

$$\begin{aligned}
\begin{Bmatrix} c(\vartheta_2) \\ s(\vartheta_2) \\ 1 \end{Bmatrix}^T \begin{bmatrix} 0 & c(\alpha_{01})s(\alpha_{02}) & 0 \\ 0 & -c(\alpha_{02}) & 0 \\ s(\alpha_{01})s(\alpha_{02}) & 0 & 0 \end{bmatrix} \begin{Bmatrix} c(\vartheta_3) \\ s(\vartheta_3) \\ 1 \end{Bmatrix} &= 0 \\
\begin{Bmatrix} c(\vartheta_2) \\ s(\vartheta_2) \\ 1 \end{Bmatrix}^T \begin{bmatrix} -s(\alpha_{01})s(\alpha_{03}) & 0 & 0 \\ c(\alpha_{03}) & 0 & 0 \\ 0 & -c(\alpha_{01})s(\alpha_{03}) & 0 \end{bmatrix} \begin{Bmatrix} c(\vartheta_3) \\ s(\vartheta_3) \\ 1 \end{Bmatrix} &= 0
\end{aligned}
\quad \text{or} \quad \begin{aligned} \bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{m}} \cdot \bar{\mathbf{q}}_3 &= 0 \\ \bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{k}} \cdot \bar{\mathbf{q}}_3 &= 0 \end{aligned} \quad (6.44)$$

This corresponds exactly to the input form of the "general orientation problem" in eq. 6.7. Due to the orthogonal chosen motor axes ($\bar{\mathbf{w}}_i$) and the vectors pointing to the S-joints ($\bar{\mathbf{v}}_i$), the third columns in the coefficient matrices are zero vectors, which leads to a great simplification of the results compared to the "general orientation problem" (paragraph 6.2.3) and will therefore be solved separately by rewriting the two equations given in eq. 6.44. The derivation of the solution uses the results of Gosselin [95].

$$\begin{aligned}
(\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{m}}_{i1})c(\vartheta_3) + (\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{m}}_{i2})s(\vartheta_3) &= 0 \\
(\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{k}}_{i1})c(\vartheta_3) + (\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{k}}_{i2})s(\vartheta_3) &= 0
\end{aligned} \quad (6.45)$$

where $\bar{\mathbf{m}}_{ij}$ and $\bar{\mathbf{k}}_{ij}$ are the column vectors of the two coefficient matrices.

The condition for the system of equations 6.45 to be fulfilled identically is that the determinant of their factors vanishes:

$$(\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{m}}_{i1})(\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{k}}_{i2}) - (\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{m}}_{i2})(\bar{\mathbf{q}}_2^T \cdot \bar{\mathbf{k}}_{i1}) = 0 \quad (6.46)$$

Expanding and rearranging the result leads to:

$$\sin(\vartheta_2)(C_1 \cos(\vartheta_2) + C_2 \sin(\vartheta_2)) = 0 \quad (6.47)$$

with the factors C given by:

$$\begin{aligned}
C_1 &= -\cos(\alpha_{01})\sin(\alpha_{02})\cos(\alpha_{03}) - \sin(\alpha_{01})\cos(\alpha_{02})\sin(\alpha_{03}) \\
C_2 &= \cos(\alpha_{02})\cos(\alpha_{03}) - \cos(\alpha_{01})\sin(\alpha_{01})\sin(\alpha_{02})\sin(\alpha_{03})
\end{aligned} \quad (6.48)$$

The first factor of eq. 6.47 leads again to the singularities on the motor axes, whereas the second factor can be solved for the second Euler angle. Back-substitution into the first equation of 6.45 leads to the final solution of the direct problem of the nominal model of the Argos structure, which has *always* four solutions:

$$\begin{aligned}
\vartheta_1 &= \alpha_1; \\
\vartheta_{2,1,2} &= \arctan 2(\pm C_1, \mp C_2); \\
\vartheta_{3,1,4} &= \arctan 2(\pm \bar{\mathbf{q}}_{2,1,2}^T \cdot \bar{\mathbf{m}}_{i1}, \mp \bar{\mathbf{q}}_{2,1,2}^T \cdot \bar{\mathbf{m}}_{i2});
\end{aligned}
\quad \begin{aligned} \bar{\mathbf{m}}_{i,1,2} &\rightarrow 6.44 \\ C_{1,2} &\rightarrow 6.48 \end{aligned} \quad (6.49)$$

6.5.2 The inverse problem

The nominal model is given by eq. 3.26 and table 3.18. For a set of given Euler angles the solution of the inverse problem supplies no, one or several sets of joint angles $\vartheta_1, \vartheta_2, \vartheta_3 \rightarrow \alpha_1, \alpha_2, \alpha_3$ and is based on a derivation given by Gosselin [95]. The system of eqs. 3.26 is decoupled in the unknown joint angles and can therefore be rewritten as:

$$a_i \cos(\alpha_i + \alpha_{0i}) + b_i \sin(\alpha_i + \alpha_{0i}) = 0 \quad i=1..3 \quad (6.50)$$

where the factors are given by:

$$\begin{aligned} a_1 &= s(\vartheta_1)s(\vartheta_2), \quad b_1 = -c(\vartheta_1)s(\vartheta_2) \\ a_2 &= -s(\vartheta_2)s(\vartheta_3), \quad b_2 = s(\vartheta_1)c(\vartheta_3) + c(\vartheta_1)c(\vartheta_2)s(\vartheta_3) \\ a_3 &= s(\vartheta_2)c(\vartheta_3), \quad b_3 = -c(\vartheta_1)s(\vartheta_3) - s(\vartheta_1)c(\vartheta_2)c(\vartheta_3) \end{aligned} \quad (6.51)$$

Eq. 6.50 can now be solved for the unknown joint angle yielding *always* two solutions per joint-link train:

$$\alpha_{i,2} = \arctan 2(\pm a_i, \mp b_i) - \alpha_{0i}; \quad i=1..3 \quad (6.52)$$

Figure 6.5 shows the two different solutions for one pantograph. For the mock-up of the Argos structure the passive R-joints can only move within a range of zero to π . Thus, only the first solution is reachable within the limitations of the joints.

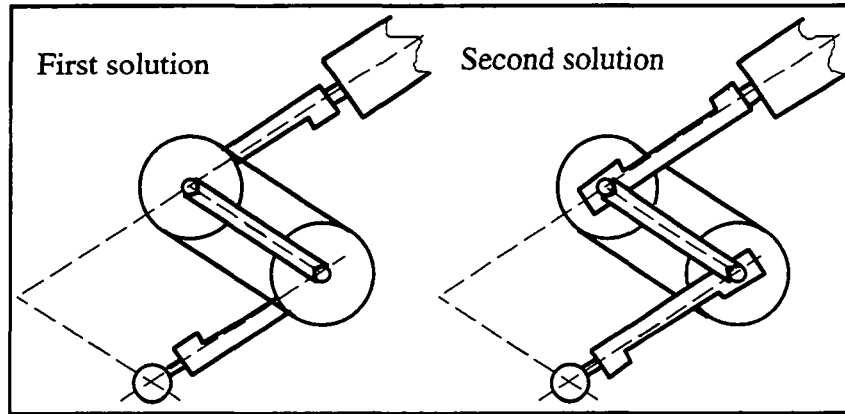


Fig. 6.5: The two solutions of the inverse problem of the Argos structure

Distinction of the first and the second solution is based on $\bar{\mathbf{w}}_i, \bar{\mathbf{u}}_i, \bar{\mathbf{v}}_i$ - vector having to build for the first and second joint-link train a left-handed system and for the third joint-link train a right-handed system. The constraint for a right-handed system is, that the triple product of the three vectors is positive:

$$(\bar{\mathbf{w}}_i \times \bar{\mathbf{u}}_i) \cdot \bar{\mathbf{v}}_i > 0 \xrightarrow{\text{Expansion}} b_i \cos(\alpha_i + \alpha_{0i}) - a_i \sin(\alpha_i + \alpha_{0i}) > 0 \quad i=1..3 \quad (6.53)$$

which leads together with eq. 6.50 to a unique solution for the inverse problem of the nominal model of the Argos structure:

$$\begin{aligned} \rightarrow \begin{cases} \alpha_1 = \arctan 2(a_1, -b_1) - \alpha_{01}; \\ \alpha_2 = \arctan 2(a_2, -b_2) - \alpha_{02}; \\ \alpha_3 = \arctan 2(-a_3, b_3) - \alpha_{03}; \end{cases} & \quad \begin{cases} a_i, b_i \rightarrow 6.51 \end{cases} \\ & \quad i=1..3 \quad (6.54) \end{aligned}$$

This simple criteria can also be used to sort the solutions of the direct problem.

6.6 Solving model 9 of the Argos structure

6.6.1 The direct problem

Model 9 is given by eq. 3.26 and table 3.16. The solution of the direct problem supplies for a set of given joint angles no, one or several sets of Euler angles $\alpha_1, \alpha_2, \alpha_3 \rightarrow \vartheta_1, \vartheta_2, \vartheta_3$. Expanding the first of the equations given in 3.26 together with table 3.16 leads again to the same equation as for the nominal model (eq. 6.40). Its solution is therefore given by eq. 6.43 with:

$$\vartheta_1 = \alpha_{01} \quad (6.55)$$

where joint angle and encoder offset are again collected in a single variable:

$$\alpha_{0i} = \alpha_i + \alpha_{0i} \quad i=1..3 \quad (6.56)$$

The remaining two equation of 3.26 together with table 3.16 can thus be rewritten as:

$$\bar{\mathbf{u}}_i^T \cdot \bar{\mathbf{R}} \cdot \bar{\mathbf{v}}_i = \bar{\mathbf{u}}_i^T \cdot \text{Rot3}(x, \alpha_{01}) \cdot \text{Rot3}(y, \vartheta_2) \cdot \text{Rot3}(x, \vartheta_3) \cdot \bar{\mathbf{v}}_i = 0 \quad i=2..3 \quad (6.57)$$

where the entire first rotation matrix is known. Therefore, the following parameter substitution is introduced:

$$\bar{\mathbf{t}}_i^T = \bar{\mathbf{u}}_i^T \cdot \text{Rot3}(x, \alpha_{01}) = (\text{Rot3}(-x, \alpha_{01}) \cdot \bar{\mathbf{u}}_i)^T \quad i=2..3 \quad (6.58)$$

Substitution into eq. 6.57 yields:

$$\{t_{ix}, t_{iy}, t_{iz}\} \cdot \begin{bmatrix} c(\vartheta_2) & 0 & s(\vartheta_2) \\ 0 & 1 & 0 \\ -s(\vartheta_2) & 0 & c(\vartheta_2) \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\vartheta_3) & -s(\vartheta_3) \\ 0 & s(\vartheta_3) & c(\vartheta_3) \end{bmatrix} \cdot \begin{Bmatrix} v_{ix} \\ v_{iy} \\ v_{iz} \end{Bmatrix} = 0 \quad i=2..3 \quad (6.59)$$

Using the same procedure as in eq. 6.28 the vectors containing the unknown Euler angles are extracted:

$$\left\{ c(\vartheta_2), s(\vartheta_2), 1 \right\} \cdot \begin{bmatrix} t_{ix} & 0 & t_{iz} \\ -t_{iz} & 0 & t_{ix} \\ 0 & t_{iy} & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & v_{ix} \\ v_{iy} & -v_{iz} & 0 \\ v_{iz} & v_{iy} & 0 \end{bmatrix} \cdot \begin{Bmatrix} c(\vartheta_3) \\ s(\vartheta_3) \\ 1 \end{Bmatrix} = 0 \quad i=2..3$$

which can be rewritten using the Sw-function defined in annex B.2:

$$\bar{\mathbf{q}}_2^T \cdot \text{Sw}(-y, \bar{\mathbf{t}}_i)^T \cdot \text{Sw}(x, \bar{\mathbf{v}}_i) \cdot \bar{\mathbf{q}}_3 = 0 \quad i=2..3 \quad (6.60)$$

Comparing these two equations with the two input equations defining the "general orientation problem" (eq. 6.7) shows that they are identical with the coefficient matrices given by:

$$\begin{aligned} \bar{\mathbf{m}} &= \text{Sw}(-y, \text{Rot } 3(-x, \alpha_{01}) \cdot \bar{\mathbf{u}}_2) \cdot \text{Sw}(x, \bar{\mathbf{v}}_2) \\ \bar{\mathbf{k}} &= \text{Sw}(-y, \text{Rot } 3(-x, \alpha_{01}) \cdot \bar{\mathbf{u}}_3) \cdot \text{Sw}(x, \bar{\mathbf{v}}_3) \end{aligned} \quad (6.61)$$

The direct problem of model 9 can now be solved by substituting eq. 6.61 into eq. 6.7. Thereby the solution is given by eq. 6.18, having eight solutions at most.

6.6.2 The inverse problem

For a set of given Euler angles the solution of the inverse problem supplies no, one or several sets of joint angles $\vartheta_1, \vartheta_2, \vartheta_3 \rightarrow \alpha_1, \alpha_2, \alpha_3$. As for the nominal model the system of eqs. 3.26 together with table 3.16 is decoupled in the unknown joint angles and can therefore be rewritten as:

$$a_i \cos(\alpha_i + \alpha_{0i}) + b_i \sin(\alpha_i + \alpha_{0i}) = 0 \quad i=1..3 \quad (6.62)$$

where the factors are together with table 3.16 given by:

$$\{a_i, b_i\} = \bar{\mathbf{v}}_i^T \cdot \bar{\mathbf{R}}^T \cdot \bar{\mathbf{R}}_i \cdot \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (6.63)$$

The solution of the inverse problem of model 9 is given by eq. 6.54 and the factors given in eq. 6.63.

6.7 Solving model 27 of the Argos structure

6.7.1 Introduction

Model 27 of the Argos structure is given in eq. 3.22. For a given set of joint angles the solution of the direct problem provides no, one or several sets of world coordinates $\alpha_1, \alpha_2, \alpha_3 \rightarrow x, y, z, \vartheta_1, \vartheta_2, \vartheta_3$, whereas the solution of the inverse problem supplies no, one or several sets of joint angles and position coordinates $\vartheta_1, \vartheta_2, \vartheta_3 \rightarrow \alpha_1, \alpha_2, \alpha_3, x, y, z$ for a given end-effector orientation. Comparing these definitions with the definitions for model 54 of the Delta robot (paragraph 6.4.1) shows that model 27 of the Argos structure is the opposite of model 54 of the Delta robot. Therefore, all propositions made in section 6.4 can be adapted to model 27 of the Argos and the reader is referred to that section. In particular, this concerns the propositions made for the two algorithms. Therefore, only some additional indications on the adaptation of the algorithms given in section 6.4 are presented in this section.

6.7.2 The direct problem

The direct problem of model 27 lies between the "general pose problem" (paragraph 6.2.4) and the "general orientation problem" (paragraph 6.2.3). Geometrically, it corresponds to fit a triangle to three circles in space. Nanua [90] has shown how such a problem can be reduced to a univariate polynomial of 16th order. Due to the high degree of the polynomial the solution is not suitable for a real-time application.

The Newton-Raphson procedure, the first algorithm proposed in section 6.4, can be adapted to the present problem without difficulties by taking the solution of the nominal model of the Argos structure given in paragraph 6.5.1 as an initial guess for the orientation.

The second algorithm might not be well suited for this problem for the following reasons: The first set of equations G1 given in eq. 3.22 represents a "general orientation problem" (paragraph 6.2.3) for a given position of the virtual rotation center. In order to reduce the "general orientation problem" to a univariate polynomial, the parameterization was chosen dependent on the position of the virtual rotation center. Therefore, the parameterization must be adapted at each iteration step because of the small position changes of the virtual rotation center. This makes the algorithms slow and unsuitable. It is better to solve the first set G1 with the Newton-Raphson procedure for the orientation

angles using the direct solution of the nominal model of the Argos structure as an initial guess. These Euler angles can be passed on to the second set G2 given in eq. 3.22, which in that case represents a "general position problem" (paragraph 6.2.2). Due to the minor difficulties of the "general position problem" the linearization for small displacement of the virtual rotation center is not necessary.

6.7.3 The inverse problem

The inverse problem is paired coupled in the unknown motor angles. Furthermore, all six equations are coupled in the three unknown coordinates of the virtual rotation center. It is expected that the non-linearity is below the one of the inverse problem of model 54 because of less trigonometric functions being involved.

The Newton-Raphson procedure may work without any difficulties for this problem if the inverse problem of the nominal model given in paragraph 6.5.2 is used as an initial guess.

The second algorithm proposed in section 6.4 can also be adapted to this problem by solving the first set G1 given in 3.22 for the joint angles, if the position of the virtual rotation center is considered to be known. It remains to solve the second set G2 for the resulting position, which can either be done linearly or not.

6.7.4 Conclusion

The solution of model 27 of the Argos structure represents the opposite to the solution of model 54 of the Delta robot.

6.8 Conclusion

In this chapter the direct and inverse problem of the different calibration models for the Delta and Argos structure were solved. Model 24 of the Delta robot, model 9 and the nominal model of Argos structure were solved by reduction to univariate polynomials. The solutions for the nominal model of the Delta robot are not given in here and can be found elsewhere [Clavel 91]. The solutions can also be derived by substituting the parameter given in table 3.12 into the solutions of model 24 given by eq. 6.26 and eq. 6.30. Two numerical procedures were proposed for model 54 and model 27. Examination of the solutions of these two models has shown that for calibration models the inverse problem is not always easier to solve than the direct problem. The difficulty of solving the inverse problem of a calibrated model may arise from two different reasons:

For the nominal model solving the inverse problem ends up in finding the inverse solution of a simple serial joint-link train, which is generally easy to solve. An example is the SPS joint-link train of a Stewart Platform (paragraph 3.3.2). For a calibration model the assumption of perfect passive S-joints might be dropped, which leads for the Stewart platform to the non-trivial problem of finding the inverse solution for a general 2RP3R joint-link train [Wang 92].

The second reason occurs only for parallel robots having less than 6 degrees of freedom. In this case the inverse problem of the calibrated model is coupled in the variables which cannot be controlled at the end-effector. An example is model 54, which is coupled in the variable of the unknown orientation of the end-effector.

There are also calibration models where both of the above mentioned difficulties arise when trying to solve the inverse problem. An example is the calibration model of the Delta robot with 138 parameters taking into account all possible deviations of the passive joints (paragraph 3.4.1). The first difficulty consists of finding a solution for the inverse problem of a general 6R joint-link train, which is a difficult task [LeeH 91]. Second, the closure equations are coupled in the three variables of the unknown orientation of the end-effector.

For real-time application the calculation time of the proposed solutions may be of importance as listed in table 6.6. The evaluations were made on a Motorola 68040 processor with mathematical coprocessor, whereas the programming was done in Mathematica™ [Maeder 91, Wolfram 91]. For evaluations of the models of the Delta robot a set of experimentally identified parameters was taken (chapter 5).

model	algorithm	DP [s]	Δr of DP {[μ m], [arc-seconds]}	IP [s]	§
Delta 54	polynomial	14000	-	-	6.2.4
	Newton-Raphson with updating of the Jacobian: Initial guess of model 24, machine precision	4.6	{0,0}	7.6	6.4.1
	as above, but without updating the Jacobian	3.4	{0,0}	-	"
	as above, but initial guess always in the center of the workspace {0,0,200,0,0,0}	2.9	{0,0}	-	"
	as above, but breaking after five steps	2.7	{0.02,0.3}	-	"
	Cascaded iterative, breaking after one and a half steps	2.0	{0.1,1.3}	1.9	6.4.2 - 6.4.4
	Cascaded according to eq. 6.33	1.1	{483,2.9}	1.1	6.4.1
Delta 24	polynomial	0.65	{483,1070}	0.53	6.3
Argos 9	polynomial	3.3	-	0.17	6.6
Argos nominal	closed form	0.05	-	0.08	6.5

Table 6.6: Calculation time of the different solutions of the direct (DP) and inverse problems (IP). Δr contains the errors in position and orientation with respect to the numerical solution with machine precision.

The calculation times given in table 6.6 are strongly dependent on the hardware as well as on the programming environment. A final judgment of the calculation speed can only be gained by an implementation into a commercially available controller. However, some qualitative tendencies in table 6.6 can still be observed such as the fact that fast solvable models are models, which can be reduced to a univariate small-order polynomial. Such kind of reduced models are up to fourth order analytically solvable. The cascaded iterative algorithm for model 54 given in table 6.6 is based on the evaluation of two times a second order polynomial and once a linear system, which may explain its short evaluation time.

7. Conclusion

7.1 Summary

In this work it was attempted to provide systematic approach as a base for future work on accuracy improvement of parallel robots.

To represent the field of parallel robots, two structures, a purely translative (Delta) and a purely rotative were chosen. The rotative mechanism is a novel design of a spherical parallel wrist called Argos.

Detailed studies of these two examples led to some more generally applicable tools for the calibration of parallel robots:

- A *formula* allowing to calculate the *number of non-redundant parameters* for a complete calibration model.
- A *systematic parameterization* using either Denavit-Hartenberg or Hayati parameters.
- *Measurement devices* as well as special set-ups require no external measurement device.
- Use of *implicit* instead of forward calibration and *semiparametric calibration* if no quality control of the mechanical parts is foreseen.
- Use of *Levenberg-Marquardt* or *Gauss-Newton based on singular value decomposition* supplies enough *robustness*.
- Implementation issue showed a similar pattern for both of the examples, which allowed to propose *fast algorithms* to solve the direct and inverse problem of the calibrated models.

In addition to all these aspects related to calibration, propositions for the design of accurate parallel robots have been given and tested for the two chosen examples.

The *main contribution* of this thesis consists in the *experimental verification* of the tools proposed for accuracy improvement of the two parallel structures Delta and Argos, showing an improvement of up to a factor of 10 to be possible.

7.2 Future work

This work may serve as a base for adapting a parallel robot to a high precision task. Once the task is known as for example to built with a Delta topology a machine tool, the appropriate concepts and models can be chosen, according to the accuracy required.

Suggestions for future works:

- Optimization of *measurement points* taken for calibration, a task which is related to the question of observability and sensitivity of parameters.
- Investigations on the *global validity* of an identified parameter set.
- Standardization of *criteria* to characterize the *quality (goodness) of fit*.
- Design of an *accurate, contactless measurement devices*, in particular for all three degrees of freedom in orientation [VDI 91].
- Development of a *real-time sensor* for end-effector feedback of the Delta and other robots.
- Investigation of *errors* generated by the *transmission* between the motor and first revolving joint as well as error generated by a *varying payload*.
- Extension of this work to *dynamic aspects*. The goal is to determinate the inertial properties of the various links, which is defined as level 3 calibration [Everett 87, Roth 87].
- Calibration of an *industrial version* of the Delta robot using the proposed tools.

I hope that this work will contribute to increase the number of industrial applications of the fascinating parallel structures.

Annex A Additional figures

A.1 Generic derivation of spherical mechanisms

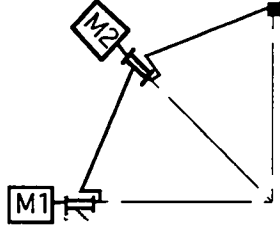
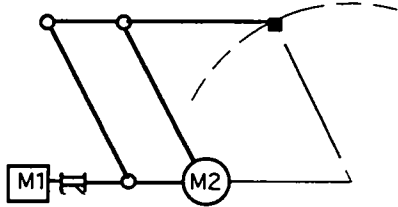
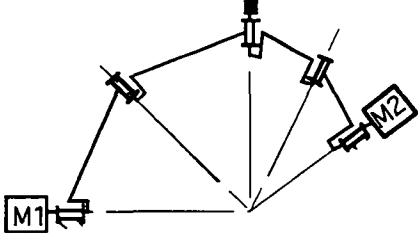
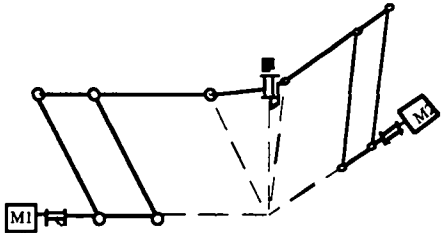
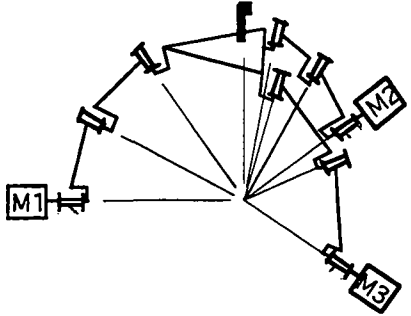
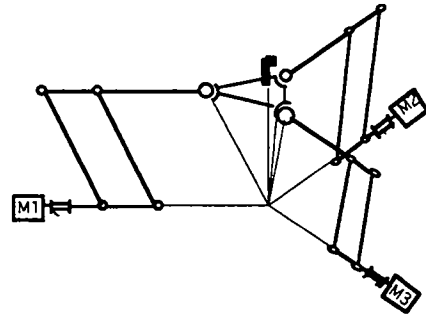
SCARA type structures	Pantograph type structures
Serial with two degrees of freedom (DoF)	
	
[Stackhouse 79] As for a SCARA robot two additional DoF can be added	Addition of three DoF in the parallelogram avoids static overdetermination. With an additional rod parallel to the upper horizontal rod -> " Roticulator " [Rosheim 89]
Parallel with two degrees of freedom (DoF)	
	
spherical 5R-loop named -> " KITE " [Artobolevski 75] Coaxial driving axes [Querfelli 94]	Novel design -> " PantoScope " [Vischer 95] Addition of a torsional DoF in one of two upper bars avoids static overdetermination (plus 2x3 DoF for the parallelograms) Application as force feedback manipulator [Baumann 96] With additional rods parallel to the upper horizontal ones -> " CMS joint " [Hamlin 94]
Parallel with three degrees of freedom (DoF)	
	
-> " Agile eye " [Gosselin 92] Coaxial driving axes [Asada 85] Coplanar driving axes [Gosselin 88]	Novel design -> " ARGOS " [Vischer 95] Aplanar driving axes [Bubendorf 96] With 6 DoF -> " HapticMaster " [Iwata 95]

Table A.1: Synthesis of spherical mechanisms

A.2 Solutions of the general pose problem

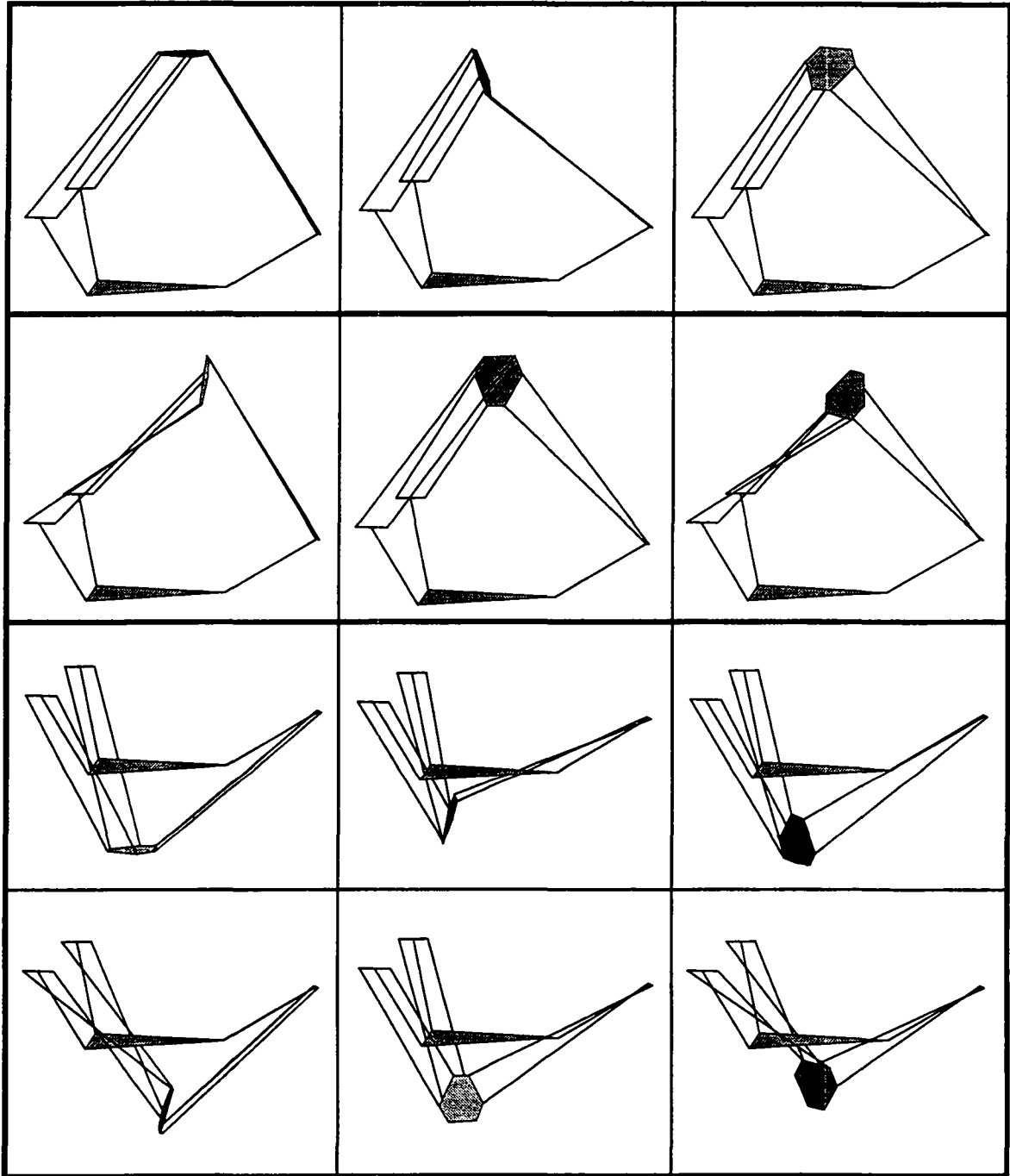


Fig. A.2: Solutions of the general pose problem for a Delta robot for the same set of joint angles. The number of solutions (real roots of the 40th order polynomial) depends on the joint angles and varies between 8 and 16.

Annex B Vector and matrix functions

In this work, the following vector and matrix functions were used: *Rot3*, *Rot4*, *Trans4*, *Sw*, *Sh*. The first three functions are commonly used in robotics and are described in Paul [81] or Craig [89]. The numbers 3 and 4 indicate the rank of the resulting matrix and thus whether it is a homogeneous matrix or not. Here the definition of the *Rot3*, *Sw* and *Sh*-functions are given. The latter two functions *Sw* and *Sh* were introduced in chapter 6. A software package for Mathematica™ containing these functions is available from the author.

B.1 The Rot3 (Rotate)-function

An unit-vector $\bar{\mathbf{k}}$ and an angle θ are the input arguments of the Rot3-function. The output is a *rotation matrix* describing a rotation about the axis $\bar{\mathbf{k}}$ by the angle θ . The formula is sometimes referred to as "*Rodrigues' formula*":

$$Rot3(\bar{\mathbf{k}}, \theta) = \begin{bmatrix} k_x k_x \text{vers}(\theta) + \cos(\theta) & k_y k_x \text{vers}(\theta) - k_z \sin(\theta) & k_z k_x \text{vers}(\theta) + k_y \sin(\theta) \\ k_x k_y \text{vers}(\theta) + k_z \sin(\theta) & k_y k_y \text{vers}(\theta) + \cos(\theta) & k_z k_y \text{vers}(\theta) - k_x \sin(\theta) \\ k_x k_z \text{vers}(\theta) - k_y \sin(\theta) & k_y k_z \text{vers}(\theta) + k_x \sin(\theta) & k_z k_z \text{vers}(\theta) + \cos(\theta) \end{bmatrix} \quad (\text{B.1})$$

with $\text{vers}(\theta) = 1 - \cos(\theta)$

Example (Rotation about the x-axis):

$$Rot3(\bar{\mathbf{x}}, \theta) = Rot3\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \theta\right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix}$$

In the main text, the notation of rotations about the principal axes $\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\mathbf{z}}$ is simplified to $Rot3(x, \theta)$, $Rot3(y, \theta)$ and $Rot3(z, \theta)$ not each time indicating that the first argument is a vector, which is already clear from the definition.

B.2 The Sw (Swap)-function

The Sw(Swap)-function is closely related to the Rot3-function. It is basically a swapping procedure between the angle of a rotation matrix and a postmultiplied vector.

$$\text{Rot}3(\bar{\mathbf{k}}, \theta) \cdot \bar{\mathbf{v}} \equiv \text{Sw}(\bar{\mathbf{k}}, \bar{\mathbf{v}}) \cdot \begin{Bmatrix} \cos(\theta) \\ \sin(\theta) \\ 1 \end{Bmatrix} = \text{Sw}(\bar{\mathbf{k}}, \bar{\mathbf{v}}) \cdot \bar{\boldsymbol{\theta}} \quad (\text{B.2})$$

The input argument is a unit rotation axis $\bar{\mathbf{k}}$ and a postmultiplied vector $\bar{\mathbf{v}}$. Output is a 3x3 matrix, which satisfies eq. B.2.

$$\text{Sw}(\bar{\mathbf{k}}, \bar{\mathbf{v}}) \equiv \begin{bmatrix} v_x - k_x(\bar{\mathbf{k}} \cdot \bar{\mathbf{v}}) & k_y v_z - k_z v_y & k_x(\bar{\mathbf{k}} \cdot \bar{\mathbf{v}}) \\ v_y - k_y(\bar{\mathbf{k}} \cdot \bar{\mathbf{v}}) & k_z v_x - k_x v_z & k_y(\bar{\mathbf{k}} \cdot \bar{\mathbf{v}}) \\ v_z - k_z(\bar{\mathbf{k}} \cdot \bar{\mathbf{v}}) & k_x v_y - k_y v_x & k_z(\bar{\mathbf{k}} \cdot \bar{\mathbf{v}}) \end{bmatrix} \quad (\text{B.3})$$

Rotations about the principal axes $\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\mathbf{z}}$:

$$\begin{aligned} \text{Rot}3(\pm \bar{\mathbf{x}}, \alpha) \cdot \bar{\mathbf{v}} &= \text{Sw}(\pm \bar{\mathbf{x}}, \bar{\mathbf{v}}) \cdot \begin{Bmatrix} \cos(\alpha) \\ \sin(\alpha) \\ 1 \end{Bmatrix} & \text{with} & \quad \text{Sw}\left(\pm \bar{\mathbf{x}}, \begin{Bmatrix} v_x \\ v_y \\ v_z \end{Bmatrix}\right) = \begin{bmatrix} 0 & 0 & v_x \\ v_y & \mp v_z & 0 \\ v_z & \pm v_y & 0 \end{bmatrix} \\ \text{Rot}3(\pm \bar{\mathbf{y}}, \alpha) \cdot \bar{\mathbf{v}} &= \text{Sw}(\pm \bar{\mathbf{y}}, \bar{\mathbf{v}}) \cdot \begin{Bmatrix} \cos(\alpha) \\ \sin(\alpha) \\ 1 \end{Bmatrix} & \text{with} & \quad \text{Sw}\left(\pm \bar{\mathbf{y}}, \begin{Bmatrix} v_x \\ v_y \\ v_z \end{Bmatrix}\right) = \begin{bmatrix} v_x & \pm v_z & 0 \\ 0 & 0 & v_y \\ v_z & \mp v_x & 0 \end{bmatrix} \\ \text{Rot}3(\pm \bar{\mathbf{z}}, \alpha) \cdot \bar{\mathbf{v}} &= \text{Sw}(\pm \bar{\mathbf{z}}, \bar{\mathbf{v}}) \cdot \begin{Bmatrix} \cos(\alpha) \\ \sin(\alpha) \\ 1 \end{Bmatrix} & \text{with} & \quad \text{Sw}\left(\pm \bar{\mathbf{z}}, \begin{Bmatrix} v_x \\ v_y \\ v_z \end{Bmatrix}\right) = \begin{bmatrix} v_x & \mp v_y & 0 \\ v_y & \pm v_x & 0 \\ 0 & 0 & v_z \end{bmatrix} \end{aligned}$$

1. Example (with the abbreviations: $\cos(\alpha) \rightarrow c\alpha$ and $\sin(\alpha) \rightarrow s\alpha$):

$$\text{Rot}3(\bar{\mathbf{x}}, \alpha) \cdot \bar{\mathbf{v}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \cdot \begin{Bmatrix} v_x \\ v_y \\ v_z \end{Bmatrix} = \begin{bmatrix} 0 & 0 & v_x \\ v_y & -v_z & 0 \\ v_z & v_y & 0 \end{bmatrix} \cdot \begin{Bmatrix} c\alpha \\ s\alpha \\ 1 \end{Bmatrix} = \text{Sw}(\bar{\mathbf{x}}, \bar{\mathbf{v}}) \cdot \bar{\boldsymbol{\alpha}}$$

2. Example:

$$\begin{aligned} \bar{\mathbf{v}}^T \cdot \text{Rot}3(\bar{\mathbf{x}}, \alpha) &= \{v_x, v_y, v_z\} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} = \{c\alpha, s\alpha, 1\} \cdot \begin{bmatrix} 0 & v_y & v_z \\ 0 & v_z & -v_y \\ v_x & 0 & 0 \end{bmatrix} \\ &= (\text{Rot}3(\bar{\mathbf{x}}, \alpha)^T \cdot \bar{\mathbf{v}})^T = (\text{Rot}3(-\bar{\mathbf{x}}, \alpha) \cdot \bar{\mathbf{v}})^T = (\text{Sw}(-\bar{\mathbf{x}}, \bar{\mathbf{v}}) \cdot \bar{\boldsymbol{\alpha}})^T = \bar{\boldsymbol{\alpha}}^T \cdot \text{Sw}(-\bar{\mathbf{x}}, \bar{\mathbf{v}}) \end{aligned}$$

Further examples can be found in chapter 6.

B.3 The Sh (Shrink)-function

The *Sh* (Shrink)-function is used to change the representation of a univariate polynomial from a symmetrical vector / matrix / vector notation to an asymmetrical vector / vector notation:

$$\{x^{n-1}, \dots, x, 1\} \cdot \begin{bmatrix} G_{11} & G_{12} & \dots & G_{1n} \\ G_{21} & \cdot & & \cdot \\ \vdots & & \cdot & \cdot \\ G_{n1} & \cdot & \cdot & G_{nn} \end{bmatrix} \cdot \begin{bmatrix} x^{n-1} \\ \vdots \\ x \\ 1 \end{bmatrix} \equiv \{x^{2n-2}, \dots, x^{n-1}, \dots, x, 1\} \cdot \begin{bmatrix} P_{2n-2} \\ \vdots \\ P_{n-1} \\ \vdots \\ P_1 \\ P_0 \end{bmatrix} \quad (\text{B.4})$$

where the new vector of factors $\bar{\mathbf{P}}$ is the output of the *Sh*-function having the square matrix $\bar{\mathbf{G}}$ as input argument:

$$\text{Sh} \left(\begin{bmatrix} G_{11} & G_{12} & \dots & G_{1n} \\ G_{21} & \cdot & & \cdot \\ \vdots & & \cdot & \cdot \\ G_{n1} & \cdot & \cdot & G_{nn} \end{bmatrix} \right) = \left\{ \begin{array}{c} \hline G_{11} \\ \hline G_{21} \quad + \quad G_{12} \\ \hline \dots \quad + \quad \dots \quad + \quad \dots \\ \hline G_{n1} \quad + \quad \cdot \quad + \quad \cdot \quad + \quad G_{1n} \\ \hline \cdot \quad \quad \cdot \quad \quad \cdot \\ \hline \cdot \quad \quad \cdot \\ \hline G_{nn} \\ \hline \end{array} \right\} = \begin{bmatrix} P_{2n-2} \\ \vdots \\ P_{n-1} \\ \vdots \\ P_1 \\ P_0 \end{bmatrix} \quad (\text{B.5})$$

Applications of the *Sh*-function are given in paragraph 6.2.3.

Annex C Non-linear least-squares estimation

In this annex methods for unconstrained non-linear least-squares estimation are reviewed, based on the books of Schwarz [86], Fletcher [87], Press [89], [Mooring 91] and [Schröer 93]. All methods intend to solve the following problem (cf. section 5.2):

Given are N non-linear error equations forming a residual vector:

$$\bar{\mathbf{f}}(\bar{\mathbf{p}}) \equiv \bar{\mathbf{r}} \quad (\text{C.1})$$

Solve for the unknown n dimensional parameter vector minimizing the quadratic merit function for $N > n$:

$$Q(\bar{\mathbf{p}}) \equiv \bar{\mathbf{r}}^T \cdot \bar{\mathbf{r}} = \bar{\mathbf{f}}(\bar{\mathbf{p}})^T \cdot \bar{\mathbf{f}}(\bar{\mathbf{p}}) \quad (\text{C.2})$$

C.1 Newton's method

A *necessary* condition for a minimum of the merit function Q is that its gradient vector (first order derivatives) with respect to $\bar{\mathbf{p}}$ is zero:

$$\nabla(Q, \bar{\mathbf{p}}) = \nabla(\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}, \bar{\mathbf{p}}) = 2 * \nabla(\bar{\mathbf{f}}^T, \bar{\mathbf{p}}) \cdot \bar{\mathbf{f}} = 2 * \bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}} \equiv \bar{\mathbf{0}} \quad (\text{C.3})$$

where ∇ is the Nabla-Operator allowing to write the gradient of Q as:

$$\nabla \left(Q, \begin{Bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{Bmatrix} \right) = \begin{Bmatrix} \frac{\partial Q}{\partial p_1} \\ \frac{\partial Q}{\partial p_2} \\ \vdots \\ \frac{\partial Q}{\partial p_n} \end{Bmatrix} \quad (\text{C.4})$$

and

$$\nabla \left(\bar{\mathbf{f}}, \begin{Bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{Bmatrix} \right) = \nabla \left(\begin{Bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{Bmatrix}, \begin{Bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{Bmatrix} \right) = \begin{bmatrix} \frac{\partial f_1}{\partial p_1}, \frac{\partial f_1}{\partial p_2}, \dots, \frac{\partial f_1}{\partial p_n} \\ \frac{\partial f_2}{\partial p_1}, \frac{\partial f_2}{\partial p_2}, \dots, \frac{\partial f_2}{\partial p_n} \\ \vdots \\ \frac{\partial f_N}{\partial p_1}, \frac{\partial f_N}{\partial p_2}, \dots, \frac{\partial f_N}{\partial p_n} \end{bmatrix} = \bar{\mathbf{J}} \in \mathbb{R}^{N \times n} \quad (\text{C.5})$$

where the matrix $\bar{\mathbf{J}}$ is called the *identification Jacobian*.

At such a point a *sufficient* condition for a minimum is that the Hessian matrix (second order derivatives) is positive definite [Mooring 91]:

$$\begin{aligned}
 \nabla(\nabla(Q, \bar{\mathbf{p}}), \bar{\mathbf{p}}) &= \nabla(2 * \nabla(\bar{\mathbf{f}}^T, \bar{\mathbf{p}}) \cdot \bar{\mathbf{f}}, \bar{\mathbf{p}}) \\
 &= 2 * (\nabla(\bar{\mathbf{f}}^T, \bar{\mathbf{p}}) \cdot \nabla(\bar{\mathbf{f}}, \bar{\mathbf{p}}) + \nabla(\nabla(\bar{\mathbf{f}}^T, \bar{\mathbf{p}}), \bar{\mathbf{p}}) \cdot \bar{\mathbf{f}}) \\
 &= 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}} + \nabla(\bar{\mathbf{J}}^T, \bar{\mathbf{p}}) \cdot \bar{\mathbf{f}}) > 0
 \end{aligned} \tag{C.6}$$

where the Hessian matrix is defined by applying the Nabla-operator twice:

$$\nabla \left(\begin{Bmatrix} \frac{\partial Q}{\partial p_1} \\ \frac{\partial Q}{\partial p_2} \\ \vdots \\ \frac{\partial Q}{\partial p_n} \end{Bmatrix}, \begin{Bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{Bmatrix} \right) = \begin{bmatrix} \frac{\partial^2 Q}{\partial p_1^2}, \frac{\partial^2 Q}{\partial p_1 \partial p_2}, \dots, \frac{\partial^2 Q}{\partial p_1 \partial p_n} \\ \frac{\partial^2 Q}{\partial p_1 \partial p_2}, \frac{\partial^2 Q}{\partial p_2^2}, \dots, \frac{\partial^2 Q}{\partial p_2 \partial p_n} \\ \vdots \\ \frac{\partial^2 Q}{\partial p_1 \partial p_n}, \frac{\partial^2 Q}{\partial p_2 \partial p_n}, \dots, \frac{\partial^2 Q}{\partial p_n^2} \end{bmatrix} \tag{C.7}$$

and

$$\nabla(\bar{\mathbf{J}}, \bar{\mathbf{p}}) = \begin{array}{c} \begin{array}{c} \frac{\partial^2 f_1}{\partial p_1 \partial p_n} \\ \frac{\partial^2 f_1}{\partial p_1^2} \\ \frac{\partial^2 f_1}{\partial p_1 \partial p_2} \\ \vdots \\ \frac{\partial^2 f_1}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1^2} \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_2} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1^2} \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_2} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_n} \end{array} \\ \vdots \\ \begin{array}{c} \frac{\partial^2 f_1}{\partial p_1 \partial p_2} \\ \frac{\partial^2 f_1}{\partial p_1^2} \\ \vdots \\ \frac{\partial^2 f_1}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_2} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_2} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_n} \end{array} \\ \vdots \\ \begin{array}{c} \frac{\partial^2 f_1}{\partial p_1 \partial p_n} \\ \frac{\partial^2 f_1}{\partial p_1^2} \\ \vdots \\ \frac{\partial^2 f_1}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_n} \end{array} \\ \vdots \\ \begin{array}{c} \frac{\partial^2 f_1}{\partial p_1 \partial p_n} \\ \frac{\partial^2 f_1}{\partial p_1^2} \\ \vdots \\ \frac{\partial^2 f_1}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_2}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_n} \\ \vdots \\ \frac{\partial^2 f_N}{\partial p_1 \partial p_n} \end{array} \end{array} = \bar{\bar{\mathbf{H}}} \in \mathfrak{R}^{N \times n \times n} \tag{C.8}$$

where the tensor $\bar{\bar{\mathbf{H}}}$ will be called the *identification Hessian* (or *curvature matrix*).

Eq. C.3 represents a nonlinear system of n equations coupled in the n unknowns $\bar{\mathbf{p}}$. For numerical resolution a Newton-Raphson algorithm can be used which is based on a local linearization for a set of known kinematic parameters $\bar{\mathbf{p}}_k$:

$$\bar{\mathbf{0}} = \nabla(Q, \bar{\mathbf{p}}) \approx \nabla(Q, \bar{\mathbf{p}})|_{\bar{\mathbf{p}}_k} + \nabla(\nabla(Q, \bar{\mathbf{p}}), \bar{\mathbf{p}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \quad (\text{C.9})$$

Comparison of eq. C.9 with eqs. C.3, C.6 and C.8 shows that this equation can also be written as:

$$-\left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}}\right)|_{\bar{\mathbf{p}}_k} = \left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}} + \bar{\bar{\mathbf{H}}}^T \cdot \bar{\mathbf{f}}\right)|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \quad (\text{C.10})$$

The system of n equations C.10 can be linearly solved for the n unknowns $\bar{\Delta \mathbf{p}}_k$ and added to $\bar{\mathbf{p}}_k$ for the next iteration. The criteria to stop is fulfilled if the merit function Q doesn't change any further. Newton's method to solve *nonlinear least-squares problems* is therefore summarized by:

$-\left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}}\right) _{\bar{\mathbf{p}}_k} = \left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}} + \bar{\bar{\mathbf{H}}}^T \cdot \bar{\mathbf{f}}\right) _{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k$ $\bar{\mathbf{p}}_{k+1} = \bar{\mathbf{p}}_k + \bar{\Delta \mathbf{p}}_k$ $\left(\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}\right) _{\bar{\mathbf{p}}_k} - \left(\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}\right) _{\bar{\mathbf{p}}_{k+1}} < \varepsilon \rightarrow \text{end}$	$\bar{\mathbf{f}} \rightarrow \text{C.1}$ $\bar{\mathbf{J}} \rightarrow \text{C.4}$ $\bar{\bar{\mathbf{H}}} \rightarrow \text{C.8}$
---	---

(C.11)

Newton's method converges quadratically to its fixed point [Schröder 93]. The crucial point of Newton's method is the computationally intensive calculation of the identification Hessian containing second order derivatives of the calibration model. Taking for example model 24 of the Delta robot, $0.5 \times 24 \times 25 = 300$ partial derivatives must be calculated. Taking further 50 measurement points, where the full position is known leads to $300 \times 3 \times 50 = 45'000$ different elements in this tensor. For model 54 with full pose measurement there would even be $445'500$ different elements. According to Press [89] including the identification Hessian can even have a destabilizing effect if the model fits badly or if the measurement is contaminated by outlier points. Dropping the identification Hessian in Newton's method, eq. C.11 merge to *Gauss-Newton method* discussed in the next section.

¹ To transpose a tensor of the dimension $N \times 3 \times 3$ means to rotate the dimensions once clockwise

$$\rightarrow \bar{\bar{\mathbf{H}}}^T \in \mathbb{R}^{3 \times 3 \times N}$$

C.2 Gauss-Newton method

Gauss-Newton method is based on a *local linearization* of the non-linear error equations given in eq. C.1 for a known set of kinematic parameters $\bar{\mathbf{p}}_k$ [Schwarz 86]:

$$f_j(\bar{\mathbf{p}}) \approx f_j(\bar{\mathbf{p}})|_{\bar{\mathbf{p}}_k} + \nabla(f_j(\bar{\mathbf{p}}), \bar{\mathbf{p}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \equiv \rho_i \quad j=1..N \quad (\text{C.12})$$

or in vector notation:

$$\bar{\mathbf{f}}(\bar{\mathbf{p}}) \approx \bar{\mathbf{f}}(\bar{\mathbf{p}})|_{\bar{\mathbf{p}}_k} + \nabla(\bar{\mathbf{f}}(\bar{\mathbf{p}}), \bar{\mathbf{p}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k = \bar{\mathbf{f}}|_{\bar{\mathbf{p}}_k} + \bar{\mathbf{J}}|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \equiv \bar{\rho} \quad (\text{C.13})$$

Eq. C.13 represents an overdetermined, linear system of equations ($N > n$). The residuals $\bar{\rho}$ differs from the one given in C.1 ($\bar{\mathbf{r}}$). Its solution is a problem of optimization. Again, a merit function, which is the square of the residual vector will be introduced:

$$Q \equiv \bar{\rho}^T \cdot \bar{\rho} = (\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}})|_{\bar{\mathbf{p}}_k} + 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k + \bar{\Delta \mathbf{p}}_k^T \cdot (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \quad (\text{C.14})$$

The *necessary* condition for a minimum of the quadratic merit function Q is that its gradient with respect to $\bar{\Delta \mathbf{p}}_k$ is zero:

$$\begin{aligned} \nabla(Q, \bar{\Delta \mathbf{p}}_k) &= \nabla \left((\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}})|_{\bar{\mathbf{p}}_k} + 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k + \bar{\Delta \mathbf{p}}_k^T \cdot (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \right) \\ &= 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}})|_{\bar{\mathbf{p}}_k} + \left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}} + (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})^T \right)|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \\ &= 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}})|_{\bar{\mathbf{p}}_k} + 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k \equiv \bar{\mathbf{0}} \end{aligned} \quad (\text{C.15})$$

At such a point the sufficient condition for a minimum is that the Hessian matrix is positive semidefinite [Mooring 91]:

$$\begin{aligned} \nabla(\nabla(Q, \bar{\Delta \mathbf{p}}_k), \bar{\Delta \mathbf{p}}_k) &= \nabla \left(2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}})|_{\bar{\mathbf{p}}_k} + 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k, \bar{\Delta \mathbf{p}}_k \right) \\ &= 2 * (\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})|_{\bar{\mathbf{p}}_k} \geq 0 \end{aligned} \quad (\text{C.16})$$

This claim is always satisfied if the identification Jacobian $\bar{\mathbf{J}}$ has full rank [Schwarz 86, Mooring 91] since in this case the inverse of the matrix $(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})$ is the estimated positive semidefinite covariance matrix of the fitted parameters $\bar{\mathbf{p}}_k$ [Press 89]. Eq. C.15 represents a linear, determined system with respect to the unknown vector $\bar{\Delta \mathbf{p}}_k$. The n equations are called *normal equations*. The Gauss-Newton method can now be written as:

$$\begin{array}{|l}
-\left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}}\right)_{\bar{\mathbf{p}}_k} = \left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}}\right)_{\bar{\mathbf{p}}_k} \cdot \Delta \bar{\mathbf{p}}_k \\
\bar{\mathbf{p}}_{k+1} = \bar{\mathbf{p}}_k + \Delta \bar{\mathbf{p}}_k \\
\left(\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}\right)_{\bar{\mathbf{p}}_k} - \left(\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}\right)_{\bar{\mathbf{p}}_{k+1}} < \varepsilon \rightarrow end
\end{array}
\quad
\begin{array}{|l}
\bar{\mathbf{f}} \rightarrow \text{C.1} \\
\bar{\mathbf{J}} \rightarrow \text{C.4}
\end{array}
\tag{C.17}$$

Gauss-Newton method converges linearly to its fixed point [Schröder 93]. Comparison with Newton's method given in eq. C.11 shows that the identification Hessian as defined in eq. C.8 doesn't appear in the Gauss-Newton method: Gauss-Newton method is based on a linear system of equation (eq. C.14) whereas Newton's method is based on a quadratic system of equations¹. The identification Hessian reflects the nonlinearity of the calibration model for the least-squares estimation. Thus, the less nonlinear the calibration model is, the more identical the two methods become. Driels has shown by simulations that the identification of kinematic parameters of a serial robot is weakly nonlinear [Schröder 93, Schröder 93a]. Thus, the Gauss-Newton method may be a good choice for the identification of kinematic parameters of parallel robots.

The system of eqs. C.17 for $\Delta \bar{\mathbf{p}}_k$ could for examples be solved by premultiplying from the left with the inverse of the square matrix $\left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}}\right)$:

$$-\left(\left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}}\right)^{-1} \cdot \bar{\mathbf{J}}^T\right)_{\bar{\mathbf{p}}_k} \cdot \bar{\mathbf{f}}_{\bar{\mathbf{p}}_k} = \Delta \bar{\mathbf{p}}_k \tag{C.18}$$

The matrix $\left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}}\right)^{-1} \cdot \bar{\mathbf{J}}^T$ is called the *pseudoinverse*. It solves overdetermined linear systems of equations by minimizing of the square of the residuals.

¹ Instead of deriving Newton's method as shown in section C.1, the error equations (eq. C.1) can directly be approximated by a quadratic function given by a Taylor series expansion:

$$\bar{\mathbf{f}} \approx \bar{\mathbf{f}}_{\bar{\mathbf{p}}_k} + \bar{\mathbf{J}}_{\bar{\mathbf{p}}_k} \cdot \Delta \bar{\mathbf{p}}_k + \frac{1}{2} \left(\Delta \bar{\mathbf{p}}_k^T \cdot \bar{\mathbf{H}}_{\bar{\mathbf{p}}_k} \cdot \Delta \bar{\mathbf{p}}_k \right)$$

Substitution into the merit function (eq. C.2) and dropping all terms of magnitude $\Delta^{3..4}$ will lead to Newton's method (C.11) after calculation of the gradient, too. This shows a further common feature of Newton's and Gauss-Newton method: The merit functions of both methods are *quadratic* in the unknown vector $\Delta \bar{\mathbf{p}}_k$ [Fletcher 87]. See eq. C.14

Eq. C.18 shows a major drawback of Gauss-Newton method: If the square matrix $(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})$ suffers from rank deficiency, it is no more invertible [Golub 83]. This may happen if several measurements are performed for little varying robot configurations (row dependency in the identification Jacobian) or if the calibration model contains dependent variables (column dependency in the identification Jacobian). Not only exactly dependent rows or columns causes the Gauss-Newton method to fail, also very closely related rows or columns disturb with an *ill-conditioned* identification Jacobian the Gauss-Newton method reasonable. An example is given by the nearly parallel $\bar{\mathbf{b}}_i$ and $\bar{\mathbf{D}}_i$ vectors in model 54, which leads to an almost column dependency in the identification Jacobian.

To overcome the problem of an identification Jacobian, which is singular or numerically close to be singular, many propositions can be found in literature. Two different approaches will be reviewed in the next two sections. One is called *singular value decomposition* and allows to "invert" singular matrices in a least square sense whereas the second one, the *Levenberg-Marquardt method* is a modification of Gauss-Newton method itself.

C.3 Singular Value Decomposition (SVD)

According to [Golub 83] or Press [89], any $N \times n$ matrix $\bar{\mathbf{J}}$ can always be written as:

$$\bar{\mathbf{J}}|_{\bar{\mathbf{p}}_k} = \bar{\mathbf{U}} \cdot \text{diag}(w_i) \cdot \bar{\mathbf{V}}^T \quad (\text{C.19})$$

where $\bar{\mathbf{U}}$ is an $N \times n$ column-orthogonal matrix, $\text{diag}(w_i)$ an $n \times n$ diagonal matrix with positive or zero elements w_j , called the *singular values* of the matrix $\bar{\mathbf{J}}$, and $\bar{\mathbf{V}}$ an $n \times n$ orthogonal matrix.

The singular values are the square roots of the eigenvalues of the $n \times n$ matrix $(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}})$. The ratio from the largest to the smallest singular value is the *condition number*, which shows how close $\bar{\mathbf{J}}$ is to be singular. A matrix $\bar{\mathbf{J}}$ with an infinite condition number is singular whereas a matrix $\bar{\mathbf{J}}$ with a large condition number is called *ill-conditioned*.

$$\kappa(\bar{\mathbf{J}}) = \frac{w_{i\max}}{w_{i\min}} \quad (\text{C.20})$$

Thus, SVD provides information about the quality of a matrix $\bar{\mathbf{J}}$. With the help of the condition number it can be checked, if the identification Jacobian contains dependent parameters or if a bad located set of measurement configurations was taken.

The main advantage of SVD lies in its ability to handle any kind of linear system in a least-squares senses regardless of being ill-conditioned or not. SVD is directly applied to the linear, overdetermined system of error equations (eq. C.13) given as starting point for the Gauss-Newton method:

$$\bar{\rho} = \bar{\mathbf{f}}|_{\bar{\mathbf{p}}_k} + \bar{\mathbf{J}}|_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta\mathbf{p}}_k$$

where the solution $\bar{\Delta\mathbf{p}}_k$ which minimizes the Euclidean norm of the residual vector $\bar{\rho}$ is given by the inversion of eq. C.19¹:

$$\bar{\Delta\mathbf{p}}_k = -\bar{\mathbf{V}} \cdot \text{diag}\left(\frac{1}{w_i}\right) \cdot \bar{\mathbf{U}}^T \cdot \bar{\mathbf{f}}|_{\bar{\mathbf{p}}_k} \quad (\text{C.21})$$

If the identification Jacobian is ill-conditioned small, singular values will perturb eq. C.21. According to Press [89] the inverse of these small singular values have to be replaced by zero in order to obtain a suitable solution for a linear least square problem by eq. C.21.

Using Gauss-Newton method based on singular value decomposition can now be summarized as:

$\begin{aligned} \bar{\mathbf{J}} _{\bar{\mathbf{p}}_k} &= \bar{\mathbf{U}} \cdot \text{diag}(w_i) \cdot \bar{\mathbf{V}}^T \\ \bar{\Delta\mathbf{p}}_k &= -\bar{\mathbf{V}} \cdot \text{diag}\left(\frac{1}{w_i}\right) \cdot \bar{\mathbf{U}}^T \cdot \bar{\mathbf{f}} _{\bar{\mathbf{p}}_k} \quad , \text{if } w_i < \varepsilon_1 \text{ then } \frac{1}{w_i} \rightarrow 0 \\ \bar{\mathbf{p}}_{k+1} &= \bar{\mathbf{p}}_k + \bar{\Delta\mathbf{p}}_k \\ (\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}) _{\bar{\mathbf{p}}_k} - (\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}) _{\bar{\mathbf{p}}_{k+1}} &< \varepsilon_2 \rightarrow \text{end} \end{aligned}$	$\begin{aligned} \bar{\mathbf{f}} &\rightarrow \text{C.1} \\ \bar{\mathbf{J}} &\rightarrow \text{C.4} \end{aligned}$
--	--

(C.22)

Algorithms for SVD of the identification Jacobian can be found in MatLabTM (based on LINPACK) or MathematicaTM [Dongarra 79, Golub 83, Moler 92, Wolfram 91]. Iterative algorithms working only on numerical matrices are used. However, using SVD for linear least square problems corresponds to suppressing of symptoms of a calibration model containing dependent parameters or a badly chosen set of measurement configurations. If possible, it's better to fight the source of these symptoms. This was the motivation to derive a formula allowing the calculation of the number of independent kinematic parameters for a complete calibration model (eq. 3.3).

¹ $\bar{\Delta\mathbf{p}}_k = -\bar{\mathbf{J}}^{-1} \cdot \bar{\mathbf{f}}|_{\bar{\mathbf{p}}_k} = -\left(\bar{\mathbf{U}} \cdot \text{diag}(w_i) \cdot \bar{\mathbf{V}}^T\right)^{-1} \cdot \bar{\mathbf{f}}|_{\bar{\mathbf{p}}_k} = -\left(\bar{\mathbf{V}}^T\right)^{-1} \cdot \left(\text{diag}(w_i)\right)^{-1} \cdot \left(\bar{\mathbf{U}}\right)^{-1} \cdot \bar{\mathbf{f}}|_{\bar{\mathbf{p}}_k} = -\bar{\mathbf{V}} \cdot \text{diag}\left(\frac{1}{w_i}\right) \cdot \bar{\mathbf{U}}^T \cdot \bar{\mathbf{f}}|_{\bar{\mathbf{p}}_k}$

C.4 Levenberg-Marquardt method

The Levenberg-Marquardt (LM) method modifies the normal equations given in eq. C.17 to overcome problems related to singularities of the identification Jacobian [Mooring 91]:

$$\begin{array}{|l}
 -\left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}}\right)_{\bar{\mathbf{p}}_k} = \left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}} + \lambda_k \cdot \bar{\mathbf{I}}\right)_{\bar{\mathbf{p}}_k} \cdot \bar{\Delta \mathbf{p}}_k, \text{ with } \lambda_k > 0 \\
 \bar{\mathbf{p}}_{k+1} = \bar{\mathbf{p}}_k + \bar{\Delta \mathbf{p}}_k \\
 \left(\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}\right)_{\bar{\mathbf{p}}_k} - \left(\bar{\mathbf{f}}^T \cdot \bar{\mathbf{f}}\right)_{\bar{\mathbf{p}}_{k+1}} < \varepsilon \rightarrow \text{end}
 \end{array}
 \quad
 \begin{array}{|l}
 \bar{\mathbf{f}} \rightarrow \text{C.1} \\
 \bar{\mathbf{J}} \rightarrow \text{C.4}
 \end{array}
 \tag{C.23}$$

where $\bar{\mathbf{I}}$ is an $n \times n$ identity matrix and λ_k an adjustable positive scalar factor.

The right-hand side of eq. C.23 is always invertable, independent of the rank of $\bar{\mathbf{J}}^T \cdot \bar{\mathbf{J}}$ [Schwarz 86]. A further advantage of the LM-method is, that it can handle also strong nonlinear systems. If the initial guess of parameters for Newton's or Gauss-Newton method is far away from the optimal set for the minimum of the nonlinear merit function, quadratic approximations of this function may be of poor quality. In other words, the minimum of the quadratic merit function found by either of the two methods lies far away from the minimum of the nonlinear merit function. In such a situation it would be preferable to do a small step in the *steepest descent direction* of the nonlinear merit function in the point $\bar{\mathbf{p}}_k$ in order to get closer to the searched minimum [Press 89]. The steepest descent direction lays in the opposite direction of the gradient vector of the nonlinear merit function Q (eq. C.3). The *steepest descent method* is given by:

$$\bar{\mathbf{p}}_{k+1} = \bar{\mathbf{p}}_k - \frac{1}{\lambda_k} \left(\bar{\mathbf{J}}^T \cdot \bar{\mathbf{f}} \right)_{\bar{\mathbf{p}}_k}, \text{ with } \lambda_k > 0 \tag{C.24}$$

where λ_k has to be adjusted in such a way that the minimum won't be passed by a too large step width.

Eq. C.24 allows to further interpret the LM-method. When λ_k is very large, the right hand side of eq. C.23 is forced into a diagonally dominated matrix and merges into eq. C.24. On the other hand, as λ_k approaches zero, eq. C.23 merges into Gauss-Newton method given by eq. C.17. It remains to adjust λ_k in such a way that far from the searched minimum the steepest decent method in the LM-method becomes dominant and the closer the updated parameter set $\bar{\mathbf{p}}_k$ gets to the searched minimum the more the

LM-method is forced into Gauss-Newton method in order to have a quadratic convergence rate¹.

According to Marquardt λ_k is adjusted in such a way that an iteration step is only executed while decreasing at the same time λ_k , if the merit function gets smaller (-> Gauss-Newton). Otherwise, λ_k is increased (-> Steepest decent) without updating the parameter set. For further information the reader is referred to Press [89] and paragraph 5.3.3.

¹ Actually, the Gauss-Newton method converges linearly, but the closer it gets to the searched minimum, the more it is the same as Newton's method, which has a quadratic convergence rate.

Annex D Solving non-linear systems of equations

In this annex the *method of resultants* reducing non-linear systems of equations to a *univariate polynomial* is reviewed. This method doesn't reflect the state of the art in mathematics, but it is a *transparent method*, which *avoids non-reducible roots* solving *polynomial systems of equations*. The method is also known as *Sylvester's or Bezout's method*. For more detailed information the reader is referred to [Dolster 83]. Resultants are an extension of determinants, which allow to check whether two polynomials have a common root.

D.1 Polynomial systems

The resultant of two polynomials $(P1, P2)$ is defined by the multiplication of all possible combinations of differences of their roots α_i, β_j :

$$P1(x) = \sum_{i=0}^n a_i x^{n-i}, \quad P2(x) = \sum_{j=0}^s b_j x^{s-j} \quad (D.1)$$

with their resultant:

$$\text{Res}(P1, P2, x) \equiv a_0^s b_0^n \prod_{i=1}^n \prod_{j=1}^s (\alpha_i - \beta_j) \quad (D.2)$$

D.1.1 Applying resultants to check for common roots

According to the definition given in eq. D.2 the resultant will vanish if two polynomial have a common root.

Given are two polynomials:

$$\begin{aligned} P1: x^3 - x^2 + 2 &= 0 \\ P2: x^2 + 2x + 1 &= 0 \end{aligned} \quad (D.3)$$

To calculate the resultant of the two polynomials the method of Sylvester will be used, who showed that the resultant can be calculated as the determinant of the following 5×5 matrix:

$$\text{Res}(P1, P2, x) \equiv \begin{vmatrix} 1 & -1 & 0 & 2 & 0 \\ 0 & 1 & -1 & 0 & 2 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \end{vmatrix} \quad (\text{D.4})$$

For the chosen example explicit calculation of D.4 yields in fact to zero because the polynomials have a common root at -1.

D.1.2 Application of determinants for solving polynomial systems

Given is a polynomial system of equations coupled in three variables x, y, z :

$$\begin{aligned} -xz^2 - 5x - y - z &= 0 \\ z^2 - 2xz + 2yz - 16x - 4y - 4z - 11 &= 0 \\ xz^2 + 2xz + 6 &= 0 \end{aligned} \quad (\text{D.5})$$

D.5 is linear in the variables x and y and can therefore be rearranged as:

$$\begin{bmatrix} (-z^2 - 5) & -1 & -z \\ (-2z - 16) & (2z - 4) & (z^2 - 4z - 11) \\ (z^2 + 2z) & 0 & 6 \end{bmatrix} \cdot \begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix} = \bar{\bar{\mathbf{A}}} \cdot \bar{\mathbf{x}} = \bar{\mathbf{0}} \quad (\text{D.6})$$

In order to solve D.6 for x and y overdetermined system of equations, it has to be guarantied that $\bar{\bar{\mathbf{A}}}$ becomes linearly dependent for a valid z . The condition to satisfy is $|\bar{\bar{\mathbf{A}}}| = 0$:

$$|\bar{\bar{\mathbf{A}}}| = z^4 - 10z^3 + 35z^2 - 50z + 24 \equiv 0 \quad (\text{D.7})$$

Solving this univariate fourth order polynomial for z and back-substituting z into D.6 leads to the following sets of solutions:

$$\{x, y, z\}_{1..4} = \{-2, 11, 1\}, \{-3/4, 19/4, 2\}, \{-2/5, 13/5, 3\}, \{-1/4, 5/4, 4\} \quad (\text{D.8})$$

D.1.3 Application of resultants for solving polynomial systems

Ignoring that D.5 is linear in x and y , D.5 could also be written as polynomials in z :

$$\begin{aligned} G_1: & (-x) z^2 - z - (5x + y) = 0 \\ G_2: & z^2 + (-2x + 2y - 4) z - (16x + 4y + 11) = 0 \\ G_3: & (x) z^2 + (2x) z + (6) = 0 \end{aligned} \quad (\text{D.9})$$

The necessary condition for the two polynomials G_1 and G_2 to have a common root in z is that their resultant (eq. D.2) vanishes. Applying D.4 leads to:

$$\text{Res}(G_1, G_2, z) = \begin{vmatrix} -x & -1 & -(5x+y) & 0 \\ 0 & -x & -1 & -(5x+y) \\ 1 & (-2x+2y-4) & -16x-4y-11 & 0 \\ 0 & 1 & (-2x+2y-4) & -16x-4y-11 \end{vmatrix} \equiv 0 \quad (\text{D.10})$$

Expanding D.10 leads to a bivariate polynomial H_1 in x and y :

$$H_1: (4x)y^3 + (28x^2 - 1)y^2 + (46x + 120x^2 + 92x^3)y + (276x^4 + 560x^3 + 260x^2 - 40x - 11) = 0 \quad (\text{D.11})$$

The resultants of the remaining two possible combinations of the three polynomials in D.9 can be calculated accordingly:

$$H_2: \text{Res}(G_2, G_3, z) \equiv 0 \quad \text{and} \quad H_3: \text{Res}(G_1, G_3, z) \equiv 0 \quad (\text{D.12})$$

The H_i -polynomials build a set of three different bivariate polynomials in x and y . To eliminate a variable, for example y , the method of the resultants is applied again resulting in a new set of three univariate polynomials K_i in x :

$$\begin{aligned} K_1: \text{Res}(H_1, H_2, y) &\equiv 0 \\ K_2: \text{Res}(H_2, H_3, y) &\equiv 0 \\ K_3: \text{Res}(H_1, H_3, y) &\equiv 0 \end{aligned} \quad (\text{D.13})$$

The K_i -polynomials being *cumbersome* are not shown explicitly. They are of the order of 14, 14 and 12. The *greatest common divisor* of the K_i -polynomials is the searched univariate polynomial P :

$$P: 3x*(2+x)*(3+4x)*(2+5x)*(1+4x)=0 \quad (\text{D.14})$$

The solution $x=0$ is incompatible with polynomial G_3 and has therefore to be dropped. The remaining four solutions are identical to the ones given in eq. D.8.

The first solution of the example chosen (eq. D.5) based on determinants is much more efficient than the second approach using resultants. The first solution was only possible because x and y are linear in D.5. The example was chosen this way to demonstrate the relationship between resultants and determinants. If none of the variables appears linearly in the polynomials, only the second method can be used. Starting the resultant

method with the variable of the lowest order is recommended. Since computational complexity is reduced and will prevent to oversee linear variables.

The system of equations D.5 can also be solved without the use of determinants or resultants by solving the first eq. for x , substituting the solution into the third equation and solving that one for y . Substitution of x and y into the second equation of D.5 yields the same univariate polynomial as in D.7. Again this is only possible because D.5 is linear in x and y . Equal treatment of a general polynomial system of equations will *introduce roots* which make the reduction to a univariate polynomial *impossible*. This can be verified by trying to solve D.5 in a reverse order: Extract z of equation one and substitute this result into equation three in order to solve this eq. for x and so on ... and you get trapped.

Application of resultants to solve polynomial systems of equations is not the most powerful tool in mathematics. Having more than two polynomials and applying the resultant method will generally lead to a univariate polynomial of a higher degree than is the number of solutions of the original system of polynomials. A method which supplies directly a univariate polynomial of minimal order is based on the use of *Gröbner bases*, for which the reader is referred to Adams [94]. For a comparison and overview of different methods see Raghavan [95].

D.2 Transcendental systems

This annex gives an example for the reduction of a system of equations including transcendental functions to a univariate equation. Two different methods are discussed and illustrated by an example. Given are two equations coupled in the unknowns α, β :

$$\begin{bmatrix} a(\beta) & b(\beta) \\ c(\beta) & d(\beta) \end{bmatrix} \begin{Bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{Bmatrix} = \begin{Bmatrix} k_1(\beta) \\ k_2(\beta) \end{Bmatrix} \quad (\text{D.15})$$

where the factors a, b, c, d, k are any kind of function of β such as for example:

$$\begin{aligned} a(\beta) &= \beta^2 + 2\beta + 1, & b(\beta) &= 3, & c(\beta) &= -\beta, \\ d(\beta) &= 3\beta - 5, & k_1(\beta) &= 4\beta, & k_2(\beta) &= \beta^2 + 2\beta \end{aligned} \quad (\text{D.16})$$

D.2.1 Use of the trigonometric identity

Firstly, D.15 is solved linearly for $\cos(\alpha)$ and $\sin(\alpha)$:

$$\cos(\alpha) = \frac{\begin{vmatrix} k_1 & b \\ k_2 & d \\ a & b \\ c & d \end{vmatrix}}{ad - bc} = \frac{k_1 d - k_2 b}{ad - bc}, \quad \sin(\alpha) = \frac{\begin{vmatrix} a & k_1 \\ c & k_2 \\ a & b \\ c & d \end{vmatrix}}{ad - bc} = \frac{k_2 a - k_1 c}{ad - bc} \quad (\text{D.17})$$

Substitution of this result into the trigonometric identity $\cos^2(\alpha) + \sin^2(\alpha) = 1$ leads to:

$$(k_1 d - k_2 b)^2 + (k_2 a - k_1 c)^2 = (ad - bc)^2 \quad (\text{D.18})$$

which contains only the unknown β .

As a numerical example the factors given in D.16 are substituted into D.18. This yields:

$$4\beta^8 + 16\beta^7 + 168\beta^5 - 50\beta^4 - 268\beta^3 + 544\beta^2 + 60\beta - 25 = 0 \quad (\text{D.19})$$

which is a univariate polynomial in β of the 8th order.

D.2.2 Use of resultants

A half tangent substitution is performed at first in order to convert the transcendental equations given by D.15 into polynomials:

$$t = \tan\left(\frac{\alpha}{2}\right), \quad \cos(\alpha) = \frac{1 - t^2}{1 + t^2}, \quad \sin(\alpha) = \frac{2t}{1 + t^2} \quad (\text{D.20})$$

and thus:

$$\begin{aligned} P_1: (k_1 + a) t^2 - 2b t + (k_1 - a) &= 0 \\ P_2: (k_2 + c) t^2 - 2d t + (k_2 - c) &= 0 \end{aligned} \quad (\text{D.21})$$

The condition for the two polynomials P_i to have a common root in t is their vanishing resultant. Applying D.3 leads to:

$$\text{Res}(P_1, P_2, t) = \begin{vmatrix} (k_1 + a) & -2b & (k_1 - a) & 0 \\ 0 & (k_1 + a) & -2b & (k_1 - a) \\ (k_2 + c) & -2d & (k_2 - c) & 0 \\ 0 & (k_2 + c) & -2d & (k_2 - c) \end{vmatrix} \equiv 0 \quad (\text{D.22})$$

Expansion leads to the same result as already found in D.18. Both methods are quite easy to handle, however the second method may be more generally applied.

References

[Adams 94]

Adams W. W. and Loustaunau Ph., "An Introduction to Gröbner Bases", American Mathematical Society, 1994

[Ananthanarayanan 92]

Ananthanarayanan S.P., Szymczyk C. and Goldenberg A.A. "Identification of Kinematic Parameters of Multiple Closed Chain Robotic Manipulators Working in Coordination", IEEE Int. Conf. on Robotics and Automation, Nice, France, 1992, pp 358 - 363, Vol. 1

[Artobolevski 75]

Artobolevski I., "Les mécanismes dans la technique moderne", Vol. 1, editions MIR Moscou, 1975

[Asada 85]

Asada H. and Cor Granito J.A., "Kinematic and Static Characterization of Wrist Joints and Their Optimal Design", IEEE Int. Conf. on Robotics and Automation, ST. Louis, Missouri, USA, 1985 March 25 -28, pp 244 - 250

[Asada 88]

Asada H. and Kakumoto Y., "The Dynamic RCC Hand for High-Speed Assembly", Proc. IEEE Int. Conf. on Robotics and Automation, Philadelphia, PA, April 1988, Vol. 1, pp. 120 - 125

[Badano 93]

Badano F., Betemps M., Burckhardt C.W., Clavel R. and Jutard A. "Assembly of Champferless Parts Using a Fast Robot", 24th ISIR Int. Symp. on Industrial Robots, Tokyo, Japan, 4-6 Nov. 1993, pp.: 89-96

[Baumann 96]

Baumann R., Glauser D. and Tappy D., "Force Feedback for Virtual Reality based Minimally Invasive Surgery Simulator", Proc. Medicine Meets Virtual Reality 4, San Diego, January 1996, pp 564 - 579

[Bennet 91]

Bennett D.J. and Hollerbach J. M., "Autonomous Calibration of Single-Loop Closed Kinematic Chains Formed by Manipulators with Passive Endpoint Constraints", IEEE Transaction on Robotics and Automation, Vol. 7, No. 5, Oct. 1991

[Bernhardt 93]

R. Bernhardt and S. L. Albright, "Robot calibration", Chapman and Hall Verlag (UK), 1993

[Borm 91]

Borm J.H. and Menq C.H., "Determination of Optimal Measurement Configurations for Robot Calibration Based on Observability Measure", Int. J. Robotics Research, Vol. 10, No. 1, Feb. 1991

[Bottema 79]

Bottema O. and Roth B., "Theoretical Kinematics", North Holland, Amsterdam, 1979

[Branets 73]

Branets V.N and Shmyglevskiy I.P. "Application of Quaternions to Rigid Body Rotation Problems", Nauka Press Moscow, U.S.S.R.: Trans. into English, 1974; NASA Technical Translation

[Broderick 88]

Broderick P.L. and Cipra R.J., "A Method for determining and correcting robot position and orientation errors due to manufacturing", ASME Journal of Mechanisms, Transmissions and Automation in Design, vol. 110, 3-10 (1988)

[Bubendorf 96]

Bubendorf D., "Argos: Le Nouveau Robot à Structure Parallèle Sphérique", projet de 7ème semestre, Institute de Microtechnique, Ecole Polytechnique fédérale de Lausanne (EPFL), Hiver 95-96

[Castelain 86]

Castelain J.-M., "Application de la méthode hypercomplexe aux modélisations géométriques et différentielles des robots constitués d'une chaîne cinématique simple", Thèse d'état, Université de Valenciennes et du Hainaut- Cambrésis, 1986

[Chen 86]

Chen J. and Chao L.M., "Positioning Error Analysis for Robot Manipulator with All Rotary Joints", IEEE Int. Conf. on Robotics and Automation, 1986, pp 1011 - 1016, Vol. 2

[Chiang 92]

Chiang C.H. , "Spherical Kinematics in Contrast to Planar Kinematics", Mechanism and Machine Theory, Vol. 27, No. 3, May, 1992, pp. 243 - 250

[Clavel 85]

Clavel R., "Dispositif pour le déplacement et le positionnement d'un élément dans l'espace", brevet suisse n° 672089 A5, priorité décembre 1985

[Clavel 91]

Clavel R., "Conception d'un robot parallèle rapide à 4 degrés de liberté", PhD-thesis No. 925, Ecole Polytechnique fédérale de Lausanne (EPFL), 1991

[Codourey 91]

Codourey A., "Contribution à la commande des robots rapides: application au robot DELTA à entraînement direct", PhD-thesis No. 922, Ecole Polytechnique fédérale de Lausanne (EPFL), 1991

[Craig 89]

Craig J.J., "Introduction to Robotics: Mechanics and Control", 2nd ed., Addison-Wesley, 1989

[Craig92]

Craig J.J., "Robot Calibration Facilitates Off-Line Programming", Robotics World/ March 1992

[de Casteljau 87]

de Casteljau P., "Les Quaternions", Hermès, Paris, 1987

[Demaurex 79]

Demaurex M.-O., "Approche théorique de la conception de la structure mécanique d'un robot industriel", PhD thesis No. 322, Ecole Polytechnique fédérale de Lausanne (EPFL), 1979

[Denavit 55]

Denavit J. and Hartenberg R.S., "A Kinematic Notion for Lower-pair Mechanisms Based on Matrices", Transaction of ASME Journal of Applied Mechanics, 22(2): June 1955, pp 215-221

[Devaquet 92]

Devaquet G. and Brauchli H., "A simple mechanical model for the DELTA Robot", Robotersysteme 8, pp 193-199 (1992)

[Dijksman 76]

Dijksman E.A., "Motion geometry of mechanisms", Cambridge University Press, Cambridge, 1976

[Dimentberg 65]

Dimentberg F.M., "The Screw Calculus and its Applications in Mechanics", Nauka Press Moscow, U.S.S.R.: Trans. into English, 1968, Foreign Technology Division WP-AFB, Ohio

[Dongarra 79]

Dongarra J.J., Moler C.B., Bunch J.R. and Stewart G.W., "LINPACK User's Guide", Siam Philadelphia, 1979

[Doster 83]

Doster G., "Eléments de la théorie des déterminants avec application à l'algèbre, la trigonométrie, la géométrie analytique dans le plan et dans l'espace.", Paris, Gauthier-Villars, Imprimeur-Libraire, 1883

[Driels 91]

Driels M. and Wiegand M., "Passive Calibration of a Teleoperator Mechanism", in: "Applications of Modeling & Identification to Improve Machine Performance " Ed.: Everett L.J. & Driels M Proc. of ASME Annual winter Meeting, Atlanta, GA, Dec. 1-6, 1991, Vol. 29 pp 24 - 32

[Egloff 95]

Egloff O., "Acquisition de Données à Partir de Labview™", projet de 8ème semestre, Institut de Microtechnique, Ecole Polytechnique fédérale de Lausanne (EPFL), Eté 1995

[Engel 91]

Engel G., "Stand der internationalen Normung zu Kenngrößen für Industrieroboter", in: "Industrieroboter messen und prüfen", VDI Bericht Nr. 921, 1991, pp. 1 - 14

[Everett 87]

Everett L.J., Driels M. and Mooring B.W., "Kinematic Modeling for Robot Calibration", Proc. IEEE Int. Conf. on Robotics and Automation , Raleigh, NC, Mar. 1987, Vol. 1, pp. 183-189

[Everett 88a]

Everett L.J. and Lin C.Y., "Kinematic calibration of manipulators with closed loop actuated joints", Proc. of IEEE International Conf. on Robotics and Automation, Philadelphia, PA, 1988, pp. 792-797

[Everett 88b]

Everett L. J. and Hsu T.- W., "The Theory of Kinematic Parameter Identification for Industrial Robots", Trans of the ASME. Vol. 110, March 1988, pp 96 - 101

[Everett 88c]

Everett L.J. and Suryodiprojo A.H., "A Study of Kinematic Models for Forward Calibration of Manipulators", Proc. IEEE Int. Conf. on Robotics and Automation, Philadelphia, PA, April 1988, Vol. 2, pp. 798 - 801

[Everett 89]

Everett L.J., "Forward Calibration of Closed-Loop Jointed Manipulators", Int. J. Robotics Research, Vol. 8, No. 4, August 1989

[Felder 90]

Felder M., "Méthode de mesure pour caractériser un robot", Automation Precision Sonderausgabe 1990, pp. 16 - 24

[Fichter 86]

Fichter E.F., "A Stewart Platform Based Manipulator: General Theory and Practical Construction", International of Journal of Robotic Research, Vol. 5, No. 2, Summer 1986, pp. 157 -182

[Fletcher 87]

Fletcher R., "Practical Method of Optimization", John Wiley & Sons, 1987, 2nd ed.

[Flury 94]

Flury P., "Etude de la précision du système stéréotaxique robotisé Minerva", PhD thesis No. 1236, Ecole Polytechnique fédérale de Lausanne (EPFL), 1994

[Funda 88]

Funda J. and Paul R.P., "A Comparison of Transforms and Quaternions in Robotics", Proc. IEEE Int. Conf on Robotics and Automation 1988, Vol.2 , pp. 886-891

[Geng 94]

Geng Z. and Haynes L.S., "An Effective Kinematic Calibration Method for Stewart Platforms", ISRAM, pp. 87 -92, Hawai, 15-17 august 1994

[Geodetics 94]

N.N., "Geodetics 8 axis GPM 1000 rapid manufacturing center", sells brochure, Geodetic Technology International Holdings NV, Geneva, Switzerland

[Golub 83]

Golub G.H. and van Loan Ch. F., "Matrix Computations", Oxford: North Oxford Academic Publishing, 1983

[Gosselin 88]

Gosselin C. M., "Kinematic analysis, optimization and programming of parallel robotic manipulators", Ph.D. dissertation , Dept. Mech. Eng., McGill Univ., Montréal, Québec, Canada, 1988

[Gosselin 91]

Gosselin C.M. and Sefrioui J., "Polynomial solutions for the direct kinematic problem of planar three-degree-of freedom parallel manipulators", 5th Int. Conf. on Advanced Robotics, 1991, Pisa (Italy) pp. 1124-1129

[Gosselin 92]

Gosselin C.M., Sefrioui J. and Richard M.J., "On the Direct Kinematics of General Spherical Three-Degree-of- Freedom Parallel Manipulators", Robotics, Spatial Mechanisms, and Mechanical Systems (ASME) 22nd Biennial Mechanisms Conference Scottsdale, AZ, Sept. 1992, Vol. 45, pp. 7 - 11

[Gosselin 94]

Gosselin C.M. and Hamel J.F., "The Agile Eye: A High-Performance Three-Degree-of-Freedom Camera-Orienting Device", Proceedings of the IEEE Int. Conference on Robotics and Automation, San Diego, CA, May 1994, Vol. 1, pp. 781 - 786

[Gosselin 95]

Gosselin C.M. and Gagné M., "A closed-form solution for the direct Kinematics of a special class of spherical three-degree of freedom parallel manipulators", *Computational Kinematics*, eds. J.P. Merlet and B. Ravani, Kluwer Academic Publisher, Sophia Antipolis, France, Sept 4. - 6., 1995, pp. 231-240

[Goswami 93]

Goswami A., Quaid A. and Peshkin M., "Complete Parameter Identification of a Robot from Partial Pose Information", *IEEE Int. Conf. on Robotics and Automation*, May 2 - 6, 1993, Atlanta, GA, Vol. 1, pp. 168 - 173

[Grace 92]

Grace A., "Optimization Toolbox", The MathWorks, Inc. 1992

[Guglielmetti 94]

Guglielmetti Ph., "Model - Based Control of Fast Parallel Robots: A Global Approach in Operational Space", PhD Thesis No. 1228; Ecole Polytechnique fédérale de Lausanne (EPFL), 1994

[Hamlin 94]

Hamlin G.J. and Sanderson A.C., "A Novel Concentric Multilink Spherical Joint with Parallel Robotics Applications", *Proceedings of the IEEE International Conference on Robotics and Automation*, San Diego, CA, Vol. 2, pp. 1267 - 1272, May 1994

[Hayati 83]

Hayati S. A., "Robot Arm Geometric Link Parameter Estimation", *Proc. 22nd IEEE Conf. Decision and Control*, Dec. 1983, pp. 1477 - 1483

[Hayati 85]

Hayati S. A. and Mirmirani M., "Improving the Absolute Positioning Accuracy of Robot Manipulators", *J. of Robotic Systems*, 1985, Vol. 2, No. 4, pp. 395-413

[Heeren 92]

Heeren T.A.G. and Veldpaus F.E., "An Optical System to Measure the End Effector Position for On-Line Control Purposes", *Int. J. Robotics Research*, Vol. 11, No. 1, Feb. 1992, pp. 53-63

[Hiller 88]

Hiller M. and Woernle C., "The Characteristic Pair of Joint - An Effective Approach for the Inverse Kinematic Problem of Robots", *Proc. IEEE Int. Conf. on Robotics and Automation*, Philadelphia, PA, April 1988, Vol. 2, pp. 846 - 853

[Hollis 91]

Hollis R.L., Salcudean S.E. and Allan A.P., "A Six Degree of Freedom Magnetically Levitated Variable Compliance Fine Motion Wrist: Design, Modeling and Control", *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 3, June 1991

[Hollerbach 89]

Hollerbach J. M., "A Survey of Kinematic Calibration", in: "The Robotics Review I", pp. 207 - 242, ed.: Khatib et al., MIT Press, Cambridge, MA, 1989

[Hollerbach 93]

Hollerbach J.M. and Lokhorst D.M., "Closed-Loop Kinematic Calibration of the RSI 6 DOF Hand Controller", *IEEE Int. Conf. on Robotics and Automation*, May 2 - 6, 1993, Atlanta, GA, Vol. 2, pp. 142 - 148

[Honegger 96]

"Steuerung und Regelung einer Stewart-Plattform als Werkzeugmaschine", Diplomarbeit am Inst. für Robotik, Swiss Federal Institute of Technology in Zurich (ETHZ), Feb. 1996

[Hornick 91]

Hornick M.L., "Compensation and Calibration for Improvement of Static Absolute Accuracy of IR Part 2: Measurement Techniques for Estimation of Parameter Errors", 22th ISIR Int. Symp. on Industrial Robots, Detroit, USA, 21 - 24 Oct. 1991

[Hunt 78]

Hunt K.H., "Kinematic geometry of mechanisms", Oxford, Great Britain: Oxford Calderon Press 1978

[Hunt 83]

Hunt K.H., "Structural Kinematic of In-parallel-actuated Robot-arms", *ASME Journal of Mechanisms, Transmissions and Automation in Design*, Vol. 105, Dec. 1983, pp. 705-712

[Husty 94]

Husty M.L., "An Algorithm for Solving the Direct Kinematics of Stewart-Gough-Type Platforms", Internal report, cim-94-1, Center of intelligent machines, McGill Univ., Montréal, Canada, June 1994, also in: Mech. Mach. Theory Vol. 31, No. 4. pp. 365 - 380, 1996

[Innocenti 92]

Innocenti C. and Parenti-Castelli V., "Direct Kinematics of the 6-4 Fully Parallel Manipulator with Position and Orientation Uncoupled", In: "Robotic Systems" Edited by: Tzafestas S. G., Kluwer Academic Publishers, Netherland, 1992, pp.: 3 - 10

[Innocenti 93]

Innocenti C. and Parenti-Castelli V., "Echelon form solution of direct kinematics for the general fully-parallel wrist", Mech. Mach. Theory Vol. 28, No. 4, pp. 553-561, 1993

[Innocenti 95]

Innocenti C., "Algorithms for Kinematic Calibration of fully-Parallel Manipulators", Computational Kinematics, eds. J.P. Merlet and B. Ravani, Kluwer Academic Publisher, Sophia Antipolis, France, Sept 4. - 6., 1995, pp. 241 - 250

[Iwata 95]

Iwata H., "HapticMaster", Virtual Reality News, July 1995, Vol. 4(6), p. 27

[Jagadeesh 95]

Jagadeesh B. and Kurien Issac K., "Calibration of a closed loop three degrees of freedom manipulator", 9th World IFToMM, Milano, Italy, august 29 - sept 2., 1995, Vol. 3, pp. 1762 - 1766

[Judd 90]

Judd R.P. and Kanasinski A.L.B., "A Technique to Calibrate Industrial Robots with Experimental Verification", IEEE Transactions of Robotics and Automation, Vol. 6, No 1, February 1990

[Khalil 86]

Khalil W. and Kleinfinger J.F., "A new geometric notation for open and closed-loop robots", IEEE Int. Conf. on Robotics and Automation, San Francisco, CA, Apr. 1986, pp 1174 - 1180, Vol. 2

[Khalil 89]

Khalil W., Ceanen J.L. and Enguhard C., "Identification and Calibration of the Geometric Parameters of Robots", First International Symposium on Experimental Robotics, June 19-22, 1989 Montreal, Canada, Lecture Notes in Control and Information Sciences: Ed. Hayward V. & Khatib O., Springer, pp 528 - 538

[Khalil 95]

Khalil W., Garcia G. and Delagarde J.-F., "Calibration of the Geometric Parameters of Robots without External Sensors", Proceedings of the IEEE International Conference on Robotics and Automation, Nagoya, Japan. Vol.3, pp. 3030 - 3044, May 1995

[Kim 87]

Kim M.-S., "Entwicklung eines Parameteridentifikationsverfahren zur Erhöhung der absoluten Positioniergenauigkeit vom Industrierobotern" Hanser Verlag, 1987, Forschungsberichte aus der Praxis No. 63

[Kozakiewicz 90]

Kozakiewicz C., Ogiso T. and Miyake N.P., "Calibration Analysis of a Direct Drive Robot" IEEE Int. Workshop on Intelligent Robots and Systems, IROS 90, 1990, pp.: 1 - 8

[Kugiumtzis 94]

Kugiumtzis D. and Lillekjendlie B., "Estimation Model for Kinematic Calibration of Manipulators with a Parallel Structure" Journal of Robotic Systems 11(5), 1994, pp. 399-410

[Kung 89]

Kung S.Y. and Hwang J.N., "Neural Network Architecture for Robotic Applications", IEEE Transaction on Robotics and Automation, Vol. 5, No. 5, Oct. 1989

[Kyle 93a]

Kyle S.A., "Dimensional Reference for Robot Calibration", Proc. of the Optical 3-D Measurement Techniques II, ed.: Grün A. and Kahmen H., Oct. 4-7, 1993 Zurich, Switzerland, pp.: 606-613

[Kyle 93b]

Kyle S.A., "Non-contact measurement for robot calibration", in: "Robot Calibration", eds.: Bernhardt R. and Albright S.L., Chapman & Hall, London, 1993, pp. 79 - 100

[LeeK 88]

Lee K.M. and Shah D.K., "Part 1: Kinematic Analysis of a 3 DOF In-Parallel Actuated Manipulator", IEEE Journal of Robotics and Automation, Vol. 4, June 1988, pp. 354-360

[LeeS 91]

Lee S. and Bekey G.A., "Application of Neural Networks to Robotics", Control and Dynamic Systems, Vol. 39, 1991, pp.: 1-65, Academic Press

[LeeH 91]

Lee H.Y., Woernle C. and Hiller M., "A Complete Solution for the Inverse Kinematic Problem of the General 6R Robot Manipulator", Transactions of ASME Journal of Mechanical Design, December 1991, Vol. 113, pp 481- 486

[Lin 89]

Lin C.Y., "Kinematic Calibration for closed loop Robots", PhD Thesis Texas A&M University, 1989

[Maeder 91]

Maeder R., "Programming in Mathematica", 2nd., Addison-Wesley Publishing Company, 1991

[Magnani 95]

Magnani I. and Clavel R., "New designs for micro robots", Intr. Progress in Precision Engineering, Proc. of the 8th Intr. Precision Engineering Seminar, Compiègne, France, May 1995, pp. 537 - 540

[Markendorf 90]

Markendorf A., "Räumliche Bewegungen mit der 3D-Laserinterferometrie erfassen", Laser-Praxis, October 1990, pp. 129 -130

[Masory 93]

Masory O., Wang J. and Zhuang H., "On the Accuracy of a Stewart Platform - Part II Kinematic Calibration and Compensation", IEEE Int. Conf. on Robotics and Automation, May 2 - 6, 1993, Atlanta, GA, Vol. 1, pp 725 - 731

[Masory 94]

Masory O. and Jiahuy Y., "Measurement of Pose Repeatability of Stewart Platform", In "Advances in Robot Kinematics and Computational Geometry", eds. Lenarcic, J. and Ravani, B. Kluwer academic Publishers, Dordrecht, 1994, pp. 199 - 206

[Mathematica 93]

N.N., "Guide to Standard Mathematica Packages", 3rd., Wolfram Research, 1993

[Maurine 96]

Maurine P. and Dombre E., "A Calibration Procedure for the parallel robot Delat 4", Proc. of the 1996 IEEE, Intr. Conf. on Robotics and Automation, Minneapolis, Minnesota, April 1996

[Mayer 94]

Mayer R.R. J. and Parker G.A., "A portable Instrument for 3-D Dynamic Robot Measurement using Triangulation and Laser Tracking", IEEE Trans. on Robotics and Automation, Vol. 10, No. 4, August 1994, pp. 504-516

[Meghed 83]

Megahed S. and Renaud M., "Dynamic Modeling of Robots Manipulators Containing Closed Kinematic Chains", Advanced Software in Robotics: Proceedings of (the first) international meeting; Liege, May 4-6, 1983

[Merlet 90]

Merlet J.-P., "Les robots parallèles", Hermès, Paris 1990

[Merlet 93a]

Merlet J.-P., "Les robots parallèles", 21 Juin 1993, Habilitation à diriger les recherches, version préliminaire

[Merlet 93b]

Merlet J.-P., "Calibration of parallel manipulators", in "Computational Kinematics" ed.: J. Angeles, G. Hommel and Kovacs P., Dagstuhl-Seminar Report 75, p.18

[Milos 95]

Milos R., "Cinématique des robots et les nombres hypercomplexes", projet de 7ème semestre, Institut de Microtechnique, Ecole Polytechnique fédérale de Lausanne (EPFL), Hiver 94-95

[Moler 92]

Moler C. B., Bangert S., Kleinman S. and Little J., "MATLAB: High-Performance Numerical Computation and Visualization Software", The MathWorks, Inc. 1992

[Mooring 91]

Mooring B.W. Roth Z.S. and Driels M., "The Fundamentals of Manipulator Calibration", Verlag: John Wiley & Sons, 1991

[Nahvi 94]

Nahvi A., Hollerbach J.M. and Hayward V., "Calibration of a parallel robot using multiple Kinematic closed loops", Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 407 - 412

[Nahvi 96]

Nahvi A. and Hollerbach J.M., "The Noise Amplification Index for Optimal Pose Selection in Robot Calibration", IEEE Int. Conf. on Robotics and Automation, pp. 647 - 654, 24 - 26 april 1996

[Nanua 90]

Nanua P., Waldron K.J. and Murthy V., "Direct Kinematic Solution of a Stewart Platform", IEEE Transaction on Robotics and Automation, Vol. 6, No. 4, Aug. 1990, pp 438 - 444

[Nguyen 91]

Nguyen C.C., Antrazi S.S., Zhou Z.L. and Campbell C.E., "Experimental Study of Motion Control and Trajectory Planing for a Stewart Platform Robot Manipulator", IEEE Int. Conf. on Robotics and Automation, Sacramento, CA, April 1991, Vol. 2, pp. 1873-1878

[Parikian 95]

Parikian T.F., "Kinematic Analysis and Design of Parallel Robots", Technical Report No. 95-02, Ecole Polytechnique fédérale de Lausanne (EPFL), 1995

[Parker 87]

Parker G.A., "The Present Status of Measurement Techniques for Robot Calibration", Colloquium on robot calibration and performance measurement : London, 7 Jan 1987, pp.: 1/1 - 1/5

[Patarinski 93]

Patarinski S.P. and Uchiyama M., "Position / Orientation Decoupled Parallel Manipulators", ICAR Int. Conf. on Advanced Robotics, November 1 - 2, 1993, Tokyo, Japan, pp. 153 - 158

[Paul 81]

Paul R.P., "Robot Manipulators: Mathematics, Programming and Control", MIT Press, 1981

[Payannet 85]

Payannet D., "Modélisation et correction des erreurs statiques des robots manipulateurs", PhD thesis université des sciences et techniques du Languedoc, Académie de Montpellier, Oct. 1985

[Pernette 96]

Pernette E. and Clavel R., "Parallel robots and microrobotics", Montpellier, ISRAM 1996

[Pierrot 90]

Pierrot F., Reynaud C. and Fournier A., "DELTA: A simple and efficient parallel robot", Robotica (1990) Vol. 8, pp. 105 - 109

[Pierrot 91]

Pierrot F., "Robots Pleinement Parallèles Légers: Conception, Modélisation et Commande", PhD thesis, University of Montpellier II, april 1991

[Pollard 42]

Pollard W.L.W., "Position - Contolling Apparatus", United States Patent Office, June 1942

[Prenninger 93]

Prenninger J.P., Vincze M. and Gander H., "Contactless Position and Orientation Measurement of Robot End-Effectors", IEEE Int. Conf. on Robotics and Automation, May 2 - 6, 1993, Atlanta, GA, Vol. 1, pp. 180 - 185

[Press 89]

Press W.H., Flannery B.P., Teukolsky S.A. and Vetterling W.T., "Numerical Recipes in Pascal - The Art of Scientific Computing", Cambridge University Press, 1989

[Querfelli 94]

Querfelli M. and Kumar V., "Optimization of a Spherical Five-Bar Parallel Drive Linkage", J. of Mechanical Design Vol. 116, March 1994, pp. 166 - 173

[Raghavan 91]

Raghavan M., "The Stewart Platform of General Geometry has 40 Configurations", J. of Mech. Design, Trans. ASME. v. 115, 2, June 1993, pp 277-282

[Raghavan 95]

Raghavan M. and Roth B., "Solving polynomial systems for the the kinematic analysis of mechanisms and robot manipulators", ASME J. of Mechanical Design, 117(2):71-79, Juin1995, Special 50th Anniversary Design Issue

[Ravani 83]

Ravani B. and Roth B., "Motion Synthesis using Kinematic Mappings", J. of Mechanisms, Transmissions and Automation in Design, 1983, Vol 105, pp 460-467

[Ravani 84]

Ravani B. and Roth B., "Mappings of Spatial Kinematics", J. of Mechanisms, Transmissions and Automation in Design, Sept. 1984, Vol. 106 , pp 341- 347

[Reboulet 88]

Reboulet C., "Modélisation des robots parallèles", Technique de la robotique, tome 1, architecture et commands, eds.: J.-D. Boissonnat, B. Faverjon et J.-P. Merlet, HERMES, Paris, 1988

[Renders 91]

Renders J.M., Rossignol E., Becquet M. and Hanus R., "Kinematic Calibration and Geometrical Parameter Identification for Robots", IEEE Transaction on Robotics and Automation, Vol. 7, No. 6, Dec. 1991, pp 721 - 732

[Ropponen 95]

Ropponen T. and Arai T., "Accuracy Analysis of a Modified Stewart Platform Manipulator", Proceedings of the IEEE International Conference on Robotics and Automation, Nagoya, Japan, Vol.1, May 1995, pp. 521 - 525

[Rosheim 89]

Rosheim M.E., "Robot Wrist Actuators", John Wiley & Sons, New York Chichester Brisbane Toronto Singapore, 1989

[Roth 87]

Roth Z.S., Mooring B.W. and Raviani B., "An Overview of Robot Calibration", IEEE Journal of Robotics and Automation, Vol. 3, No. 5, Oct. 1987, pp 377-385

[Sakakibara 89]

Sakakibara S. and Hiraizumi M., "Elements of next Generation robot systems which are achieving practical use - off-line programming system and three dimensional vision sensor", 20th ISIR Int. Symp. on Industrial Robots Tokyo, Japan, 4-6 Oct. 1989, pp. 743 - 750

[Sayeh 94]

Sayeh S. and Newman W.S., "A new Technique for solving robot calibration equations with partially known constraints", Proceedings of the IEEE International Conference on Robotics and Automation, San Diego, CA, Vol. 1, pp. 387 - 392, May 1994

[Schmidt 91]

Schmidt Th., "Entwicklung eines Messroboters zur flexiblen Post-Process-Messung in der varantenreichen Produktion", in: "Industrieroboter messen und pruefen", VDI Bericht Nr. 921, 1991, pp. 243 - 251

[Schröer 93]

Schröer K., "Identifikation von Kalibrationsparametern kinematischer Ketten", Carl Hanser Verlag Ed.1, 1993

[Schröer 93a]

Schröer K., "Theory of kinematic modelling and numerical procedures for robot calibration", in: "Robot Calibration"; eds.: Bernhardt R. and Albright S.L., Chapman & Hall, London, 1993, pp. 157 - 195

[Schröer 94]

Schröer K., "RoboCal - The IPK Robot Calibration Package", Industrial robot, Vol 21, No. 6, 1994, pp. 35 - 39

[Schulz 94a]

Schulz W., "Weltweiter Aufschwung bei Werkzeugmaschinen erwartet", VDI Nachrichten ; Jg. 48 / Nr. 39, 30.09.1994

[Schulz 94b]

Schulz W., "Technologische Offensive in den USA", VDI Nachrichten ; Jg. 48 / Nr. 40, 07.10.1994

[Schwarz 86]

Schwarz H.R., "Numerische Mathematik", B.G. Teuber Stuttgart 1986

[Shamma 87]

Shamma J.S. and Whitney D.E., "A Method for Inverse Robot Calibration", ASME J. Dynamic Systems, Measurement and Control, march 1987, Vol. 109, Iss 1, pp. 36-43

[Sheth 71]

Sheth P.N. and Uicker J.J., "A generalized symbolic notation for mechanisms", Journal of Engineering for Industry , Trans. ASME, Feb. 1971, pp 102 - 112

[Spur 89]

Spur G. and Schröer K., "Kalibrierung von Industrierobotern", In: " Vorschubantriebe in der Fertigungstechnik" Ed.: G. Pritschow, G. Spur and M. Weck, Hanser, 1989

[Stackhouse 79]

Stackhouse T., "A new Concept in Robot wrist Flexibility", 9th ISIR Int. Symp. on Industrial Robots 13 -15 March 1979, Wasington, DC, USA, pp. 589 - 599

[Sternheim 87]

Sternheim F., "Computation of the direct and inverse geometric models of the DELTA 4 parallel robot", Robotersysteme, vol. 1987, pp. 199-203

[Stevens 94]

Stevens B.S. and Clavel R., "The Delta parallel robot, its future in industry", In "Advances in Robot Kinematics and Computational Geometry", eds. Lenarcic, J. and Ravani, B. Kluwer academic Publishers, Dordrecht, 1994, pp. 273 - 278

[Stevens 95]

Stevens B.S., "Control Characterisation and Optimisation of Fast Industrial Robots", PhD thesis No. 1362, Ecole Polytechnique fédérale de Lausanne (EPFL), 1995

[Stewart 65]

Stewart D., "A Platform with Six Degrees of Freedom", Proceedings of the Institution of Mechanical Engineering, Vol. 180, Part 1, No. 15, 1965-66, pp. 371-386

[Stone 86]

Stone H.W., Sanderson A.C. and Neuman C.P., "Arm Signature Identification", IEEE Int. Conf. on Robotics and Automation, San Francisco, CA, Apr. 1986, pp 41 - 48, Vol. 1

[Stone 87]

Stone H.W., "Kinematic Modeling Identification, and Control of Robotic Manipulators", Kluwer Academic Publishers, Bosten, Dordrecht, Lancaster 1987

[Sugimoto 85]

Sugimoto K. and Okata T., "Compensation of Position Errors Caused by Geometric Deviation in Robot System", Robotics Research: the 2th Inter. Sym., MIT Press, Cambridge, MA, 1985, pp.: 231 - 236

[Tanner 90]

Tanner Ch., "Das Masskörperverfahren zur Vermessung von Robotern", PhD Thesis Swiss Federal Insitute of Technology in Zurich (ETHZ) Nr. 9261, 1990

[Thornton 88]

Thornton G.S., "The GEC Tetrabot - A New Serial - Parallel Assembly Robot", Proc. IEEE Int. Conf. on Robotics and Automation, Philadelphia, PA, April 1988, Vol. 1, pp. 437 - 441

[Toyama 89]

Toyama S. and Hatae S., "Error Analysis of Platform Type of Robot by means of Screw Algebra", 20th ISIR Int. Symp. on Industrial Robots Tokyo, Japan, 4-6 Oct. 1989, pp.: 635 - 642

[Tradt 91]

Tradt H.R., "Stand der Messtechnik für die Ermittlung der Kenngrößen von Industrierobotern" in: "Industrieroboter messen und prüfen", VDI Bericht Nr. 921, 1991, pp. 43 - 56

[Van Brussel 90]

Van Brussel H., "Evaluation and Testing of Robots", Annals of CIRP, Vol. 39/2, 1990, pp. 657- 664

[VDI 88]

N.N. , "Kenngrößen für Industrieroboter Einsatzspezifische Kenngrößen", VDI 2861 Blatt 2, 1988

[VDI 91]

N.N. , "Industrieroboter messen und prüfen", Berichte 921, VDI Verlag Düsseldorf, 1991

[Veitschegger 87]

Veitschegger W.K., "Robot Kinematic Error Analysis, Modeling, Calibration, and Compensation", PhD-Thesis, Northwestern University, Evanston, IL, June 1987

[Veitschegger 88]

Veitschegger W.K. and Wu C.H., "Robot Calibration and Compensation", IEEE Journal of Robotics and Automation, Vol. 4, No 6, December 1988

[Vischer 92]

Vischer D., "Cooperating Robot with Visual and Tactile Skills", IEEE Int. Conf. on Robotics and Automation, Nice, France, 1992, pp 2018 - 2025, Vol. 3

[Vischer 95]

Vischer P., "Argos: A novel parallel spherical structure", Technical report, No 95-03, Ecole Polytechnique fédérale de Lausanne (EPFL), march 1995

[Wampler 92]

Wampler C. and Arai T., "Calibration of Robots having kinematic closed-loops using non-linear least squares estimator", IFToMM-jc Conf., Intr. Sym. on Theory of Machines and Mechanisms, Nagoya (1992), Sept. 24-26, pp. 153 - 158

[Wampler 95]

Wampler C.W., Hollerbach J.M. and Arai T., "An implicit loop method for kinematic Calibration and its application to closed-chain mechanisms", IEEE Trans. on robotics and automation, vol. 11, no. 5, October 1995

[Wampler 96]

Wampler C.W., "Forward Displacement Analysis of General six-in-parallel SPS (Stewart) Platform Manipulators using Soma Coordinates", Mech. Mach. Theory Vol. 31, No. 3. pp. 331 - 337, 1996

[Wang 92]

Wang J., "Workspace Evaluation and Kinematic Calibration of Stewart Platform", Ph.D. dissertation, Florida Atlantic University, Boca Raton, FL 1992

[Wang 93]

Wang J. and Masory O., "On the Accuracy of a Stewart Platform - Part I, the Effect of Manufacturing Tolerances", IEEE Int. Conf. on Robotics and Automation, May 2 - 6, 1993, Atlanta, GA Vol. 1, pp. 114 - 120

[Whitney 86]

Whitney D.E., Lozinski C.A. and Rourke J.M., "Industrial Robot forward calibration method and results", ASME Journal of Dynamic Systems; Meas. Contr., Vol. 108, pp. 1-8, Mar. 1986

[Wiegand 96]

Wiegand A., Hebsacker M. and Honegger M., "Parallele Kinematik und Linearmotoren", Technische Rundschau, Nr. 25, 1996

[Wohlhart 94]

Wohlhart K., "Displacement analysis of the general spherical Stewart Platform", Mech. Mach. Theory, Vol. 29. No. 4, 1994, pp. 581-589

[Wolfram 91]

Wolfram S., "Mathematica: A system for doing mathematics by computer", 2nd., Addison-Wesley Publishing Company, 1991

[Zeiss 94]

N.N. , "Positioniereinheit in 6 Freiheitsgraden: HEXAPOD", sells brochure, Zeiss, Jena 1994

[Zhong 95]

Zhong X.-L. and Lewis J. M., "A New Method for Autonomous Robot Calibration", Proc. International Conference on Robotics and Automation, Nagoya, Japan, Vol.2 , pp. 1790 - 1795, May 1995

[Zhuang 90]

Zhuang H. and Roth Z. S., "A Complete and Parametrically Continuous Kinematic Model for Robot Manipulators", Proc. IEEE Int. Conf. on Robotics and Automation 1990, Vol.1, pp. 92 -97

[Zhuang 91]

Zhuang H., Roth Z. and Hamano F., "A Method for Kinematic Calibration of Stewart Platforms", Proc. of ASME Annual winter Meeting, Atlanta, GA, Dec. 1-6, 1991, Vol. 29 pp 43-48

[Zhuang 94]

Zhuang H., Wang K. and Roth Z.S., "Optimal selection of measurement configurations for robot calibration using simulated annealing", IEEE Trans. 1994, pp.: 393 - 398

[Zhuang 95]

Zhuang H., Masory O. and Yan J., "Kinematic calibration of a Stewart platform using pose measurement obtained by a single theodolite" Proc. of Intr. Con. Intelligent Robots and Systems, Pittsburgh 1995, Part 2, pp 329 - 334

[Zhuang 96]

Zhuang H. and Liu L., "Self calibration of a class of parallel manipulators" IEEE Int. Conf. on Robotics and Automation, pp. 994 - 999, 24 - 26 april 1996

[Zobel 93]

Zobel P.B. and Clavel R., "On the static Calibration of the Delta Parallel Robot", IASTED Robotics and Manufacturing, Oxford, England, 1993, pp 88 - 91

[Zobel 94]

Zobel P.B. and Di Stefano P., "Allocation of dimensional tolerances for a parallel robot", 2nd biennial European joint conference on engineering systems design and analysis, July 4-7, 1994 London



Curriculum Vitae

Personal Data

Name	Peter Andreas Vischer
Date of birth	11.09.64 in Wädenswil (ZH)
Citizenship	Basel Stadt, Switzerland

Education

09/93 - 10/96	PhD-thesis, research and teaching assistant Subjects: robotics, automation and micro-engineering
04/93 - 08/93	Research assistant, Institut de microtechnique, Ecole polytechnique fédérale de Lausanne (EPFL) Project: "automated lens assembly" in collaboration with the company Leica
10/86 - 01/92	Diploma in Mechanical Engineering, Eidgenössische Technische Hochschule Zürich (ETHZ) Subjects: general mechanical engineering, biomedical technology, combustion engines
86	Military service
81 - 85	Matura (Typus C), Freies Gymnasium Zürich

Professional Experience

02 - 07/92	University of Sydney, Department of Physiology. Subjects: Building an apparatus to subject cells cultures to pulsatile flow and finding a solution of the Navier Stokes equation of a rigid rectangular duct in the case of unsteady laminar flow.
86 - 92	4 months of practical work in different companies. Subjects: design and construction of a hydrodynamic pump, application of carbon fiber materials to construct of artificial limbs, development of a femurcondule milling machine.