



National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

THE ERROR PROPAGATION IN ROBOTS

by

CHUN LU

**A thesis
presented to the University of Ottawa
on December, 1989
in partial fulfillment of the
requirements for the degree of
MASTER of APPLIED SCIENCE
in
MECHANICAL ENGINEERING**

**Ottawa-Carleton Institute for
Mechanical and Aerospace Engineering**



Chun Lu, Ottawa, Canada, 1990



National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

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

ISBN 0-315-60103-5

Canada



UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

Abstract

The accuracy of a robot manipulator has been receiving scrutiny since the widespread acceptance of robot manipulators. The relationship between two consecutive joint coordinate frames of a robot manipulator can be completely defined by five link parameters; one is the joint variable and the other four are the geometric parameters. The basis for the open-loop manipulator control is often the relationship between the Cartesian coordinates of the end-effector and the joint coordinates; therefore, the accuracy of the Cartesian position and orientation of the end-effector with regard to the real world depends on the errors of the five link parameters for each link.

For design optimization and robot calibration, it is very important to develop a model for quantitative characterization and evaluation of the positioning and orientational errors of the end-effector. A static error propagation model is developed in order to describe the relationships between the six Cartesian errors and the five independent kinematic errors for each link.

In this thesis, a general method for evaluating the end-effector errors produced by a mix of arbitrarily distributed errors is presented. Based on this method, any different combinations of biased and mixed error distributions can be dealt with directly to give a quantitative error propagation analysis. Numerical results are presented for one, two and three degrees-of-freedom robot manipulators. Comparison of the results of the proposed model with other published model are presented and analyzed.

ACKNOWLEDGEMENT

First, I wish to express my thanks to my advisor, professor D. Neculescu, for his many valuable suggestions and advise during the course of this investigation. I also want to express my appreciations to professor A. Fahim and professor D. Redekop for their frequent helps in this study.

Assistance and encouragement from my brother in-law, Dr. Y. Mao, and my family are sincerely appreciated. At last, I want to express my thanks to Mrs. Y. F. Fu for her continuous encouragement and understanding.

Contents

Contents	i
List of Tables	ii
List of Figures	iii
Nomenclature	iv
1 Introduction	1
2 The General Error Propagation Model of a Robot Manipulator	5
2.1 Error Propagation Model Between Two Consecutive Links	5
2.2 Error Propagation Model Between End-Effector and Base Frame . .	9
3 Stochastic Error Propagation Model of a N Degrees-of-Freedom Robot Manipulator	17
3.1 The Joint Probability of Functions of Random Variables	17
3.2 Distributions of The Link Parameter Errors	20
3.2.1 Properties of Beta Distributions	20
3.2.2 Relationship between The Normalized Variable and The Actual Variable	22
3.3 The Stochastic Error Propagation Model of a N Degrees-of-Freedom Robot Manipulator	22
4 Stochastic Error Propagation Analysis of a One Degree-of-Freedom Robot Manipulator	27
4.1 Error Propagation Model of One Degree-of-Freedom Robot Manipulator	27
4.2 Stochastic Error Propagation Model of a One Degree-of-Freedom Robot Manipulator	30
4.3 Numerical Stochastic Simulations	33

5	Stochastic Error Propagation Analysis of a Two Degrees-of-Freedom Robot Manipulator	49
5.1	Error Propagation Model of a Two Degrees-of-Freedom Robot Manipulator	50
5.2	Stochastic Error Propagation Model	54
5.3	Numerical Stochastic Simulations	57
6	Stochastic Error Propagation Analysis of a Three Degrees-of-Freedom Robot Manipulator	74
6.1	Error Propagation Model	75
6.2	Stochastic Error Propagation Model	82
6.3	Numerical Stochastic Simulations	86
7	Discussion and Conclusion	110
	References	114
	Appendix	118
A	Modified A Transformation Matrix	118
B	Program for Predicting The Positioning and Orientational Error of a End-Effector	121

List of Tables

4.1	link parameter table	28
5.1	link parameter table	50
6.1	link parameter table	76

List of Figures

2.1	general error propagation graph	13
3.1	linear diagram of X and Y	23
4.1	schematic drawing of one degree-of-freedom robot manipulator	28
4.2	$d\theta_1$ distributions in different sub-domains of θ_1 ; $\theta_1 \in [-90^\circ, 90^\circ]$	35
4.3	dX distribution for which $\theta_1 \in [0^\circ, 15^\circ]$	36
4.4	dX distribution for which $\theta_1 \in (15^\circ, 75^\circ)$	37
4.5	dX distribution for which $\theta_1 \in [75^\circ, 90^\circ]$	38
4.6	dX distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$ (the whole domain)	39
4.7	dY distribution for which $\theta_1 \in [0^\circ, 15^\circ]$	40
4.8	dY distribution for which $\theta_1 \in (15^\circ, 75^\circ)$	41
4.9	dY distribution for which $\theta_1 \in [75^\circ, 90^\circ]$	42
4.10	dY distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$ (the whole domain)	43
4.11	δ_Z distribution for which $\theta_1 \in [0^\circ, 15^\circ]$	44
4.12	δ_Z distribution for which $\theta_1 \in (15^\circ, 75^\circ)$	45
4.13	δ_Z distribution for which $\theta_1 \in [75^\circ, 90^\circ]$	46
4.14	δ_Z distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$ (the whole domain of θ_1) .	47
5.1	schematic drawing of a planar two degrees-of-freedom robot manipulator	50
5.2	$d\theta_1$ distributions in different sub-domains of θ_1 ; $\theta_1 \in [-90^\circ, 90^\circ]$	59
5.3	$d\theta_2$ distributions in different sub-domains of $(\theta_1 + \theta_2)$; $(\theta_1 + \theta_2) \in [-180^\circ, 180^\circ]$	60

5.4	dX distribution for which $\theta_1 \in [0^0, 15^0]$ and $(\theta_1 + \theta_2) \in [0^0, 15^0]$	61
5.5	dX distribution for which $\theta_1 \in (15^0, 75^0)$ and $(\theta_1 + \theta_2) \in (15^0, 75^0)$. . .	62
5.6	dX distribution for which $\theta_1 \in [75^0, 90^0]$ and $(\theta_1 + \theta_2) \in [75^0, 90^0]$. . .	63
5.7	dX distribution for which $\theta_1 \in [-90^0, 90^0]$, (the whole domain of θ_1) and $(\theta_1 + \theta_2) \in [-180^0, 180^0]$, (the whole domain of $(\theta_1 + \theta_2)$). . . .	64
5.8	dY distribution for which $\theta_1 \in [0^0, 15^0]$ and $(\theta_1 + \theta_2) \in [0^0, 15^0]$	65
5.9	dY distribution for which $\theta_1 \in (15^0, 75^0)$ and $(\theta_1 + \theta_2) \in (15^0, 75^0)$. . .	66
5.10	dY distribution for which $\theta_1 \in [75^0, 90^0]$ and $(\theta_1 + \theta_2) \in [75^0, 90^0]$. . .	67
5.11	dY distribution for which $\theta_1 \in [-90^0, 90^0]$, (the whole domain of θ_1) and $(\theta_1 + \theta_2) \in [-180^0, 180^0]$, (the whole domain of $(\theta_1 + \theta_2)$). . . .	68
5.12	δ_Z distribution for which $\theta_1 \in [0^0, 15^0]$ and $(\theta_1 + \theta_2) \in [0^0, 15^0]$	69
5.13	δ_Z distribution for which $\theta_1 \in (15^0, 75^0)$ and $(\theta_1 + \theta_2) \in (15^0, 75^0)$. . .	70
5.14	δ_Z distribution for which $\theta_1 \in [75^0, 90^0]$ and $(\theta_1 + \theta_2) \in [75^0, 90^0]$. . .	71
5.15	δ_Z distribution for which $\theta_1 \in [-90^0, 90^0]$, (the whole domain of θ_1) and $(\theta_1 + \theta_2) \in [-180^0, 180^0]$, (the whole domain of $(\theta_1 + \theta_2)$). . . .	72
6.1	schematic drawing of the 3 DOF manipulator for simulation	75
6.2	$d\theta_1$ distributions in different sub-domains of θ_1 ; $\theta_1 \in [-90^0, 90^0]$. . .	89
6.3	$d\theta_2$ distributions in different sub-domains of θ_2 ; $\theta_2 \in [-90^0, 90^0]$. . .	90
6.4	$d\theta_3$ distributions in different sub-domains of $(\theta_2 + \theta_3)$; $(\theta_2 + \theta_3) \in$ $[-180^0, 180^0]$	91
6.5	dX distribution for which $\theta_2 \in [0^0, 15^0]$ and $(\theta_2 + \theta_3) \in [0^0, 15^0]$. . .	92
6.6	dX distribution for which $\theta_2 \in [-90^0, 90^0]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^0, 180^0]$, (the whole domain of $(\theta_2 + \theta_3)$). . . .	93
6.7	dY distribution for which $\theta_2 \in [0^0, 15^0]$ and $(\theta_2 + \theta_3) \in [0^0, 15^0]$	94
6.8	dY distribution for which $\theta_2 \in [-90^0, 90^0]$, (the whole domain of θ_2) and $\theta_2 + \theta_3 \in [-180^0, 180^0]$, (the whole domain of $\theta_2 + \theta_3$).	95
6.9	dZ distribution for which $\theta_2 \in [0^0, 15^0]$ and $(\theta_2 + \theta_3) \in [0^0, 15^0]$	96
6.10	dZ distribution for which $\theta_2 \in (15^0, 75^0)$ and $(\theta_2 + \theta_3) \in (15^0, 75^0)$. . .	97
6.11	dZ distribution for which $\theta_2 \in [75^0, 90^0]$ and $(\theta_2 + \theta_3) \in [75^0, 90^0]$. . .	98

6.12	dZ distribution for which $\theta_2 \in [-90^0, 90^0]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^0, 180^0]$, (the whole domain of $\theta_2 + \theta_3$).	99
6.13	δ_X distribution for which $\theta_2 \in [0^0, 15^0]$ and $(\theta_2 + \theta_3) \in [0^0, 15^0]$	100
6.14	δ_X distribution for which $\theta_2 \in (15^0, 75^0)$ and $(\theta_2 + \theta_3) \in (15^0, 75^0)$. . .	101
6.15	δ_X distribution for which $\theta_2 \in [75^0, 90^0]$ and $(\theta_2 + \theta_3) \in [75^0, 90^0]$. . .	102
6.16	δ_X distribution for which $\theta_2 \in [-90^0, 90^0]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^0, 180^0]$, (the whole domain of $\theta_2 + \theta_3$).	103
6.17	δ_Y distribution for which $\theta_2 \in [0^0, 15^0]$ and $(\theta_2 + \theta_3) \in [0^0, 15^0]$	104
6.18	δ_Y distribution for which $\theta_2 \in (15^0, 75^0)$ and $(\theta_2 + \theta_3) \in (15^0, 75^0)$. . .	105
6.19	δ_Y distribution for which $\theta_2 \in [75^0, 90^0]$ and $(\theta_2 + \theta_3) \in [75^0, 90^0]$. . .	106
6.20	δ_Y distribution for which $\theta_2 \in [-90^0, 90^0]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^0, 180^0]$, (the whole domain of $(\theta_2 + \theta_3)$	107
6.21	δ_Z distribution for which $\theta_2 \in [-90^0, 90^0]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^0, 180^0]$, (the whole domain of $(\theta_2 + \theta_3)$	108
A.1	drawing of two consecutive axes with near parallel joint axes	120

Nomenclature

A_n	modified A transformation matrix
A_n^0	the A_n matrix assuming nominal link parameters
A_n^c	actual A_n transformation matrix
a_n	the length translated along the rotated $X_{n-1} = X_n$ axis
$a_{n-1,X}^0$	X component of a_{n-1}^0 vector defined in (2.35)
$a_{n-1,Y}^0$	Y component of a_{n-1}^0 vector defined in (2.35)
$a_{n-1,Z}^0$	Z component of a_{n-1}^0 vector defined in (2.35)
$B(r,s)$	beta function defined in (3.7)
C_i	$COS(\theta_i)$
C_{y_i}	$COS(y_i)$
$C_{y_i y_j}$	$COS(y_i + y_j)$
d	differential translation vector of Δ
dA_n^0	differential change of A_n^0 transformation matrix
d_n	distance translated along Z_{n-1} axis
${}^{n-1}d$	differential translation vector of ${}^{n-1}\Delta$
dP	Cartesian positioning error of the end-effector
dq_i	differential change of link parameter q_i
dT_N^0	differential change of T_N^0
dX	X component of Cartesian positioning error
dY	Y component of Cartesian positioning error
dZ	Z component of Cartesian positioning error
d_x	X component of the differential translation vector d defined in (2.39)
d_y	Y component of the differential translation vector d defined in (2.39)
d_z	Z component of the differential translation vector d defined in (2.39)

E^1	first order differential term of dT_N^0
$E(X)$	expected value of random parameter X
$f_{D_{dP}, D_\delta}(dP, \delta)$	probability density function of the end-effector being located at the specified position and rotation
$f_{dX}(dx)$	probability density function of Cartesian positioning error dX
$f_{dY}(dy)$	probability density function of Cartesian positioning error dY
$f_{dZ}(dz)$	probability density function of Cartesian positioning error dZ
$f_{X_1, \dots, X_n}(x_1, \dots, x_n)$	joint probability density function defined in (3.4)
$f_{Y_1, \dots, Y_n}(y_1, \dots, y_n)$	joint probability density function defined in (3.4)
$f_{\delta_X}(\delta_x)$	probability density function of Cartesian orientational error δ_X
$f_{\delta_Y}(\delta_Y)$	probability density function of Cartesian orientational error δ_Y
$f_{\delta_Z}(\delta_Z)$	probability density function of Cartesian orientational error δ_Z
$J(X_1, \dots, X_n)$	Jacobian defined in (3.3)
$K1_n$	a 3×1 vector defined by (2.20)
$k2_n$	a 3×1 vector defined by (2.21)
$k3_n$	a 3×1 vector defined by (2.22)
$k4_n$	a 3×1 vector defined by (2.23)
$k5_n$	a 3×1 vector defined by (2.24)
L_a	coefficient matrix of the linear Cartesian positioning error propagation model defined in (2.61)
L_d	coefficient matrix of the linear Cartesian positioning error propagation model defined in (2.61)
L_θ	coefficient matrix of the linear Cartesian positioning error propagation model defined in (2.61)
L_α	coefficient matrix of the linear Cartesian positioning error propagation model defined in (2.61)

L_β	coefficient matrix of the linear Cartesian positioning error propagation model defined in (2.61)
$n_{n-1,X}^0$	X component of n_{n-1}^0 vector defined in (2.35)
$n_{n-1,Y}^0$	Y component of n_{n-1}^0 vector defined in (2.35)
$n_{n-1,Z}^0$	Z component of n_{n-1}^0 vector defined in (2.35)
$O_{n-1,X}^0$	X component of O_{n-1}^0 vector defined in (2.35)
$O_{n-1,Y}^0$	Y component of O_{n-1}^0 vector defined in (2.35)
$O_{n-1,Z}^0$	Z component of O_{n-1}^0 vector defined in (2.35)
P_{n-1}^0	position vector part of T_{n-1}^0
$p_{n-1,X}^0$	X component of p_{n-1}^0 defined in (2.35)
$p_{n-1,Y}^0$	Y component of p_{n-1}^0 defined in (2.35)
$p_{n-1,Z}^0$	Z component of p_{n-1}^0 defined in (2.35)
r	parameter of beta function
R_{n-1}^0	rotation matrix part of T_{n-1}^0
R_θ	coefficient matrix of the linear Cartesian orientational error propagation model defined in (2.60)
R_α	coefficient matrix of the linear Cartesian orientational error propagation model defined in (2.60)
R_β	coefficient matrix of the linear Cartesian orientational error propagation model defined in (2.60)
s	parameter of beta function
S_i	$SIN(\theta_i)$
T_N^0	the nominal transformation matrix between end-effector and base frame of a N degrees-of-freedom robot manipulator
T_N^a	the actual transformation matrix between end-effector and base frame of a N degrees-of-freedom robot manipulator
T_{n-1}^0	differential coordinate transformation defined in (2.34)
μ_X	mean value of the random parameter X
δ	differential rotation vector of Δ
${}^{n-1}\delta$	differential rotation vector of ${}^{n-1}\Delta$

- δ_X X component of Cartesian orientational error defined in (2.40)
- δ_Y Y component of Cartesian orientational error defined in (2.40)
- δ_Z Z component of Cartesian orientational error defined in (2.40)
- θ^t the column vector defined by equation (3.22)
- θ_n the angle rotated about Z_{n-1} axis
- α_n the twisted angle rotated about X_n axis
- β_n the angle rotated about Y_n axis
- Δ error matrix transformation between end-effector and base frame which is with respect to base frame
- Δa error matrix of link parameter a defined by (2.64)
- Δa^t the inverse of the error matrix Δa defined by (2.64)
- Δd error matrix of link parameter d defined by (2.63)
- Δd^t the inverse of the error matrix Δd defined by (2.63)
- ΔR the first order approximation of the differential rotation defined in (2.57)
- $\Delta \theta$ error matrix of link parameter θ defined by (2.62)
- $\Delta \theta^t$ the inverse of the error matrix $\Delta \theta$ defined by (2.62)
- $\Delta \alpha$ error matrix of link parameter α defined by (2.65)
- $\Delta \alpha^t$ the inverse of the error matrix $\Delta \alpha$ defined by (2.65)
- $\Delta \beta$ error matrix of link parameter β defined by (2.66)
- $\Delta \beta^t$ the inverse of the error matrix $\Delta \beta$ defined by (2.66)

Δa_n	error of the link parameter a_n
Δd_n	error of the link parameter d_n
$\Delta \theta_n$	error of the link parameter θ_n
$\Delta \alpha_n$	error of the link parameter α_n
$\Delta \beta_n$	error of the link parameter β_n
${}^{n-1}\Delta$	error matrix transformation between the $i - 1^{th}$ coordinate frame and i^{th} coordinate frame
${}^{n-1}\Delta_i$	unit differential change transformation caused by i^{th} link parameter and applied to $n - 1$ coordinate frame
Γ	gamma function defined in equation (3.9)
σ_X	standard deviation of the random parameter X
$\frac{\partial A_n^0}{\partial q_i}$	partial derivative of A_n^0 about link parameter i

Chapter 1

Introduction

A frequent function of industrial robots is to move objects to various locations in the work space. In general, a robot structure may be subject to three different types of error sources: kinematic errors in the individual members, dynamic errors due to gravity effect, mechanical clearance, etc. and errors in positioning the joints accurately. Tasks executed by industrial robots require a certain positioning accuracy.

Robot manufacturers usually specify robot's accuracy in two ways : repeatability and absolute accuracy. The former refers to the robot's error in returning to the previously taught position in a repetitive motion. The latter refers to the error of the end-effector with respect to a coordinate given to controller. Usually, only robot manipulator's repeatability is specified by the manufacturers. The repeatability is a measure of the variation of the end-effector's actual position when the robot is required to move in repeated maneuvers to the same specified values for the joint values in repeated maneuvers. Most commercial robots can have their repeatability around 0.1 mm. But with this repeatability, it is possible for the absolute accuracy to be only able to reach 10 mm [5]. So, programming a robot without "teaching and learning" process will not make the robot move accurately to the desired position as we expected. Therefore, the absolute positioning accuracy of a robot manipulator is a limiting factor in the open loop control mode. But, because of the highly expense of tactile and vision system, automatic manipulator under open loop control is still the main trend in industrial application.

Nowadays, more advanced and complex jobs are executed by robot manipulators. The absolute positioning accuracy of the end-effector of a robot manipulator is becoming more and more important. Since it would be very costly to build every robot to a very low tolerance, it would be more cost effective to analyze the error propagation analysis of a robot manipulator.

In the current literature, most researchers conduct their error propagation analysis based only on normal distribution assumption. Up till now, no published works give a direct mathematical approach to tackle the mixed error distributions existing in a robot manipulator.

In the previously published works, the idea of mixed error distributions existing in the link parameters was presented in [7], [3] and [15]. But none of these authors develop a mathematical model of the propagation of the mix of the error distributions. In [30] and [16], the authors developed in 1978 the general error propagation model based on 4 link parameters. But the authors included only the errors of the controlled variables in his presented model and did not fully expand the model. Also the presented model is not general enough to tackle the case when the robot manipulator has two consecutive parallel or near-parallel joint axes. Later Kumar [15] realized that the joint variable errors resulting from encoder equipped joints should be modeled as a uniformly distributed random variables. But the author made a serious mistake in assuming that the positioning errors of the end-effector are also uniformly distributed around the desired position. This mistake can be verified easily by the fact that the summation of two uniform distribution variables tends to constitute a triangular distribution.

In 1984, Wu presented a rigorously developed error propagation model in a linearized form based on four link parameter $(a_n, d_n, \theta_n, \alpha_n)$ A transformation matrix [34]. Wu assumed that all four link parameter errors for any degrees-of-freedom robot manipulator are independent, zero mean normally distributed random variables. The resulting Cartesian positioning errors of the end-effector are correlated and normally distributed random variables. The eigen vectors of the covariance matrix are used to find a new coordinate system for the end-effector positioning errors in which they are composed out of independent random variables. Consequently, the positioning error distribution functions for a required positioning error envelope can be found. This approach is however restricted to the eigen vector directions and is not general enough to include the case where the robot manipulator has two

consecutive parallel or near-parallel joint axes.

In [12] and [13], a new technique is introduced to derive the four link parameter values when the robot manipulator has two consecutive parallel or near-parallel joint axes. The authors present a linear error propagation model, but consider that a bias exists in the measurement instead from the gravity effect and configuration dependency. Foulloy and Kelley approached the calibration problem directly, without parameter identification, by devising a single compensation matrix which is valid for a small portion of the robot workvolume [9]. Endpoint position measurements are used in a least mean square error formulation to obtain optimal values for the compensation matrix.

Later, numerous papers presented rigorous applications of parameter identification techniques for static error propagation analysis based on end-effector Cartesian position measurements [4], [5], [27], [28]. In [4], Chen and Chao presented a rigorous parameter identification techniques for static error propagation based on a six parameter transformation matrix. In the presented model, the structure of the equation has physical meaning, but the coefficients of the equations have no physical meaning. In [14], Huey and Anand proposed a linearized error propagation model for normally distributed geometric parameter errors and joint variable errors assumed independent, uncorrelated and with zero mean . The authors realize that the output errors depend on both configuration and loading, but only a simplified worst case analysis is performed.

A least squares regression identification techniques is used for geometric parameter identification in [4] and a recursive least squares techniques is used in a multistage approach in [27]. In [28], Shama and Whitney modeled the geometric and non-geometric parameter errors by a nonlinear multivariable polynomial, but the structure and coefficients of the polynomial do not have any physical meaning.

Azadivar addresses the importance of the bias in the joint variable errors, but the author simplifies his model and does not include the effects of the bias of the joint variable errors in his analysis [2]. In [3], an error analysis approach based on Monte Carlo simulation techniques was proposed, but this approach includes only joint tolerance propagation, not error propagation. Chen, Wang and Yang consider all the link parameter errors are zero and normally distributed with the 3σ equal to 1 and assume that the biased effect exist in the measurement error only.

In [2], Ahmad considers the joint errors due to the static effect of the load in the end-effector, but he does not give a mathematical probability model to describe the random positioning errors of the mechanical manipulator.

In [29], Veitschegger and Wu improve Wu's previously published work [34] to include the case of two consecutive parallel or near-parallel joint axes by introducing a fifth parameter in the A transformation matrix[29]. The main thrust of Wu's paper is to develop comprehensive models for error propagation and apportionment analysis of a robot manipulator. However, their contributions do not deal with the problem of mixed and biased distributions.

Stochastic error propagation modeling for a robot manipulator is still less well developed. Recent publications described a stochastic model for mix of error distributions for a 2 DOF [21] and 6 DOF [22] robots. In this dissertation, a general error propagation model is presented to give a direct quantitative analysis of positioning and orientational errors due to the mix of arbitrarily distributed link parameter errors. The results are also documented in a paper submitted for publication [23].

The propagation of static errors is the main concern of the thesis. Dynamic error evaluation for the end-effector position and orientation is generally considered a separate topic and require models of a very different nature. This dissertation is organized in seven chapters. Chapter 2 describes the general error propagation model of a robot manipulator. Chapter 3 treats the stochastic error propagation model of a robot manipulator. Chapter 4 discusses the stochastic error propagation analysis of a one degree-of-freedom robot manipulator. Chapter 5 analyzes the stochastic error propagation analysis of a two degrees-of-freedom robot manipulator. Chapter 6 is devoted to the stochastic error propagation analysis of a three degrees-of-freedom robot manipulator.

Through this dissertation concepts are illustrated by the use of relatively simple examples. The examples treated are hypothetical, but plausible, chosen to exhibit interesting characteristics without too great a computational burden.

The models presented for error propagation are easily programmable, however they are rather computationally intensive for higher degrees of freedom ($N \geq 3$). Nevertheless, they are also potentially valuable contributions to the theory of error propagation and control of a robot manipulator, and they can be applied as a kinematic CAD tool for the design of a robot manipulator.

Chapter 2

The General Error Propagation Model of a Robot Manipulator

The positioning and orientational errors of the end point of a robot manipulator result from the combination of the errors propagated from the five link parameter errors in every joint. This chapter presents the general modeling approach for the positioning and orientational error of the end-effector based on the modified five link parameter A transformation matrix [appendix A].

2.1 Error Propagation Model Between Two Consecutive Links

R. Paul analyzed this case for the propagation of differential changes in two consecutive frames. Wu developed the equations later in the form presented here [33]. These results are used in the next paragraph for the propagation of errors of each frame to the end-effector.

In the case of a robot manipulator, differential changes of the modified A_n transformation matrix are caused by differential changes in the five link parameters ($a_n, d_n, \theta_n, \alpha_n, \beta_n$). For brevity, in the following text A_n refers to the modified A transformation matrix. If the A_n matrix assuming nominal link parameters is given by A_n^0 and the actual A transformation matrix with kinematic errors is given by

A_n^a , then the actual transformation matrix can be expressed as

$$A_n^a = A_n^0 + dA_n^0 \quad (2.1)$$

A unit differential change transformation caused by the i^{th} link parameter and applied to the $n - 1$ coordinate frame is defined as ${}^{n-1}\Delta_i$. In the transformation ${}^{n-1}\Delta_i$, a leading superscript describes the coordinate frame with respect to which the differential transformation is made, and a following subscript denotes the differential variable causing the change.

The following derivation is based on Wu's contributions [33]. In [29] and [34], a differential change dA_n may be expressed in terms of a differential change in $n - 1$ coordinate frame ${}^{n-1}\Delta_i$ and in terms of the differential change of any link parameters as

$$dA_n^0 = \sum_{i=1}^5 {}^{n-1}\Delta_i * A_n^0 dq_i \quad (2.2)$$

where $i = 1, 2, 3, 4, 5$ stands for the five link parameters, $a_n, d_n, \theta_n, \alpha_n$, and β_n respectively, and "*" represents matrix multiplication.

Thus

$$\frac{\partial A_n^0}{\partial q_i} = {}^{n-1}\Delta_i * A_n^0, \quad i = 1, 2, 3, 4, 5 \quad (2.3)$$

Expanding the above equation into matrix forms for $i = 1, 2, 3, 4, 5$ respectively give

$$\frac{\partial A_n^0}{\partial q_1} = {}^{n-1}\Delta_a * A_n^0 = \begin{bmatrix} 0 & 0 & 0 & C\theta_n \\ 0 & 0 & 0 & S\theta_n \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.4)$$

$$\frac{\partial A_n^0}{\partial q_2} = {}^{n-1}\Delta_\alpha * A_n^0 = \begin{bmatrix} -S\theta_n C\alpha_n S\beta_n & S\theta_n S\alpha_n & S\theta_n C\alpha_n C\beta_n & 0 \\ C\theta_n C\alpha_n S\beta_n & -C\theta_n S\alpha_n & -C\theta_n C\alpha_n C\beta_n & 0 \\ S\alpha_n S\beta_n & C\alpha_n & -S\alpha_n C\beta_n & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.5)$$

$$\begin{aligned} \frac{\partial A_n^0}{\partial q_3} &= {}^{n-1}\Delta_\theta * A_n^0 \\ &= \begin{bmatrix} -S\theta_n C\beta_n - C\theta_n S\alpha_n S\beta_n & -C\theta_n C\alpha_n & -S\theta_n S\beta_n + C\theta_n S\alpha_n C\beta_n & -a_n S\theta_n \\ C\theta_n C\beta_n - S\theta_n S\alpha_n S\beta_n & -S\theta_n C\alpha_n & C\theta_n S\beta_n + S\theta_n S\alpha_n C\beta_n & a_n C\theta_n \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \quad (2.6)$$

$$\frac{\partial A_n^0}{\partial q_4} = {}^{n-1}\Delta_d * A_n^0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.7)$$

$$\frac{\partial A_n^0}{\partial q_5} = {}^{n-1}\Delta_\beta * A_n^0 = \begin{bmatrix} -C\theta_n S\beta_n - S\theta_n S\alpha_n C\beta_n & 0 & C\theta_n C\beta_n - S\theta_n S\alpha_n S\beta_n & 0 \\ -S\theta_n S\beta_n + C\theta_n S\alpha_n C\beta_n & 0 & S\theta_n C\beta_n + C\theta_n S\alpha_n S\beta_n & 0 \\ -C\alpha_n C\beta_n & 0 & -C\alpha_n S\beta_n & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.8)$$

Evaluating these partial derivatives for the nominal link parameter values (*i.e.*, $\beta_n = 0$) and solving for the differential change transformation ${}^{n-1}\Delta_i$ give

$${}^{n-1}\Delta_a = \begin{bmatrix} 0 & 0 & 0 & C\theta_n \\ 0 & 0 & 0 & S\theta_n \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.9)$$

$${}^{n-1}\Delta_\alpha = \begin{bmatrix} 0 & 0 & S\theta_n & -d_n S\theta_n \\ 0 & 0 & -C\theta_n & d_n C\theta_n \\ -S\theta_n & C\theta_n & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.10)$$

$${}^{n-1}\Delta_\beta = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.11)$$

$${}^{n-1}\Delta_d = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.12)$$

$${}^{n-1}\Delta_\beta = \begin{bmatrix} 0 & -S\alpha_n & C\theta_n C\alpha_n & a_n S\theta_n S\alpha_n - d_n C\theta_n C\alpha_n \\ S\alpha_n & 0 & S\theta_n C\alpha_n & -a_n C\theta_n S\alpha_n - d_n S\theta_n C\alpha_n \\ -C\theta_n C\alpha_n & -S\theta_n C\alpha_n & 0 & a_n C\alpha_n \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.13)$$

An error matrix transformation is defined with respect to joint coordinate frame $n - 1$ such that

$$dA_n^0 = {}^{n-1}\Delta * A_0^n \quad (2.14)$$

Equating equations (2.2) and (2.14) give

$${}^{n-1}\Delta = \sum_{i=1}^5 {}^{n-1}\Delta_i dq_i \quad (2.15)$$

The error matrix transformation defined in the above equation can be considered to be composed of two vectors ${}^{n-1}d$ and ${}^{n-1}\delta$ known as the differential translation and rotation vectors with respect to the $n - 1$ coordinate frame respectively

$${}^{n-1}d = {}^{n-1}d_x i + {}^{n-1}d_y j + {}^{n-1}d_z k \quad (2.16)$$

$${}^{n-1}\delta = {}^{n-1}\delta_x i + {}^{n-1}\delta_y j + {}^{n-1}\delta_z k \quad (2.17)$$

Substituting ${}^{n-1}d_x$, ${}^{n-1}d_y$, ${}^{n-1}d_z$, ${}^{n-1}\delta_x$, ${}^{n-1}\delta_y$ and ${}^{n-1}\delta_z$ by the (1,4), (2,4), (3,4), (2,3), (1,3) and (1,2) components of the error matrix transformation defined by (2.15) gives

$${}^{n-1}d = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \Delta d_n + \begin{bmatrix} C\theta_n \\ S\theta_n \\ 0 \end{bmatrix} \Delta a_n + \begin{bmatrix} -d_n S\theta_n \\ d_n C\theta_n \\ 0 \end{bmatrix} \Delta \alpha_n + \begin{bmatrix} a_n S\theta_n S\alpha_n - d_n C\theta_n C\alpha_n \\ -a_n C\theta_n S\alpha_n - d_n S\theta_n C\alpha_n \\ a_n C\alpha_n \end{bmatrix} \quad (2.18)$$

$${}^{n-1}\delta = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \Delta \theta_n + \begin{bmatrix} C\theta_n \\ S\theta_n \\ 0 \end{bmatrix} \Delta \alpha_n + \begin{bmatrix} -S\theta_n C\alpha_n \\ C\theta_n C\alpha_n \\ S\alpha_n \end{bmatrix} \quad (2.19)$$

By defining the following five vectors [29]

$$K1_n = [0 \quad 0 \quad 1]^T \quad (2.20)$$

$$K2_n = [C\theta_n \quad S\theta_n \quad 0]^T \quad (2.21)$$

$$K3_n = [-d_n S\theta_n \quad d_n C\theta_n \quad 0]^T \quad (2.22)$$

$$K4_n = [a_n S\theta_n S\alpha_n - d_n C\theta_n C\alpha_n \quad -a_n C\theta_n S\alpha_n - d_n S\theta_n C\alpha_n \quad a_n C\alpha_n]^T \quad (2.23)$$

$$K5_n = [-S\theta_n C\alpha_n \quad C\theta_n C\alpha_n \quad S\alpha_n]^T \quad (2.24)$$

where the superscript "T" means vector transpose, equations (2.18) and (2.19) can be rewritten in the following linear form

$${}^{n-1}d = K1_n \Delta d_n + K2_n \Delta a_n + K3_n \Delta \alpha_n + K4_n \Delta \beta_n \quad (2.25)$$

$${}^{n-1}\delta = K1_n\Delta\theta_n + K2_n\Delta\alpha_n + K5_n\Delta\beta_n \quad (2.26)$$

The error matrix transformation ${}^{n-1}\Delta$ is completely defined by the differential translation and rotation vectors given in the equations (2.25) and (2.26). The importance of the error matrix transformation is that it gives an error propagation model between two consecutive link coordinate frames. Later it can be applied to determine the complete error propagation model between end-effector and base frame.

2.2 Error Propagation Model Between End-Effector and Base Frame

If the position and orientation of the end-effector of a N degrees-of-freedom robot manipulator assuming nominal link parameters is given by T_N^0 and the actual position and orientation of the end-effector is given by T_N^a , then the actual transformation T_N^a can be expressed as

$$T_N^a = T_N^0 + dT_N^0 \quad (2.27)$$

From the homogeneous transformation A_n defined by (A.3), the location of the end-effector with respect to the base can be defined as

$$T_N^a = A_1^a * A_2^a * \dots * A_N^a \quad (2.28)$$

Substituting equations (2.1) and (2.27) into (2.28) gives

$$T_N^0 + dT_N^0 = \prod_{n=1}^N (A_n^0 + dA_n^0) \quad (2.29)$$

Expanding (2.29) gives the following linear form

$$T_N^0 + dT_N^0 = T_N^0 + E^1 + E^2 + E^3 + \dots + E^N \quad (2.30)$$

where E^i represents i^{th} order differential error term. It is assumed that all the link parameter errors are small; therefore, the higher order terms can be ignored, then equation (2.30) can be simplified to

$$\begin{aligned} dT_N^0 &= E^1 \\ &= \sum_{n=1}^N (A_1^0 * A_2^0 * \dots * A_{n-1}^0 * dA_n^0 * \dots * A_N^0) \end{aligned} \quad (2.31)$$

We define an error matrix transformation Δ with respect to base frame such that

$$dT_N^0 = \Delta * T_N^0 \quad (2.32)$$

where Δ is the error matrix transformation which represents the differential translation and rotation transformation with respect to base frame. Substituting equations (2.32) and (2.14) into (2.31) gives

$$\Delta * T_N^0 = \sum_{n=1}^N (A_1^0 * A_2^0 \cdots * A_{n-1}^0 * {}^{n-1}\Delta * A_n^0 * \cdots * A_N^0) \quad (2.33)$$

Then the error matrix transformation Δ can be solved as

$$\Delta = \sum_{n=1}^N (T_{n-1}^0 * {}^{n-1}\Delta * (T_{n-1}^0)^{-1}) \quad (2.34)$$

The above equation is important as it relates differential changes of the link parameters to the differential change of the end-effector. Before we use this result we will first expand the matrix product on the right hand side of the above equation. In doing this, a great deal of simplification occurs and gives us a direct relationship between elements of the error matrix transformations Δ and ${}^{n-1}\Delta$. The transformation T_{n-1}^0 in equation (2.34) is known as the differential coordinate transformation.

If the elements of the differential coordinate transformation T_{n-1}^0 is represented in terms of the vector R_{n-1} and P_{n-1} as

$$\begin{aligned} T_{n-1}^0 &= \begin{bmatrix} R_{n-1}^0 & P_{n-1}^0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} n_{n-1,x}^0 & o_{n-1,x}^0 & a_{n-1,x}^0 & p_{n-1,x}^0 \\ n_{n-1,y}^0 & o_{n-1,y}^0 & a_{n-1,y}^0 & p_{n-1,y}^0 \\ n_{n-1,z}^0 & o_{n-1,z}^0 & a_{n-1,z}^0 & p_{n-1,z}^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.35)$$

then the inverse of T_{n-1}^0 is

$$(T_{n-1}^0)^{-1} = \begin{bmatrix} n_{n-1,x}^0 & n_{n-1,y}^0 & n_{n-1,z}^0 & -p_{n-1}^0 \cdot n_{n-1}^0 \\ o_{n-1,x}^0 & o_{n-1,y}^0 & o_{n-1,z}^0 & -p_{n-1}^0 \cdot o_{n-1}^0 \\ a_{n-1,x}^0 & a_{n-1,y}^0 & a_{n-1,z}^0 & -p_{n-1}^0 \cdot a_{n-1}^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.36)$$

where $n_{n-1}, o_{n-1}, a_{n-1}$ and p_{n-1} are the four column vectors and " \cdot " represents the vector dot product. If we define the inverse differential coordinate transformation $(T_{n-1}^0)^{-1}$ as

$$(T_{n-1}^0)^{-1} = \begin{bmatrix} n'_{n-1,x} & o'_{n-1,x} & a'_{n-1,x} & p'_{n-1,x} \\ n'_{n-1,y} & o'_{n-1,y} & a'_{n-1,y} & p'_{n-1,y} \\ n'_{n-1,z} & o'_{n-1,z} & a'_{n-1,z} & p'_{n-1,z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.37)$$

then equation (2.34) can be rewritten as

$$\Delta = \sum_{n=1}^N \begin{bmatrix} 0 & -{}^{n-1}\delta \cdot a'_{n-1} & {}^{n-1}\delta \cdot o'_{n-1} & n'_{n-1} \cdot (({}^{n-1}\delta \times p'_{n-1}) + {}^{n-1}d) \\ {}^{n-1}\delta \cdot a'_{n-1} & 0 & -{}^{n-1}\delta \cdot n'_{n-1} & o'_{n-1} \cdot (({}^{n-1}\delta \times p'_{n-1}) + {}^{n-1}d) \\ -{}^{n-1}\delta \cdot o'_{n-1} & {}^{n-1}\delta \cdot n'_{n-1} & 0 & a'_{n-1} \cdot (({}^{n-1}\delta \times p'_{n-1}) + {}^{n-1}d) \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.38)$$

where the components of ${}^{n-1}d$ are defined by equations (2.18) and (2.25) and the components of ${}^{n-1}\delta$ are defined by (2.19) and (2.26). As ${}^{n-1}\Delta$, the error matrix transformation Δ can be described by two vectors d and δ , the differential translation and rotation vectors with respect to base frame, respectively

$$d = d_x i + d_y j + d_z k \quad (2.39)$$

$$\delta = \delta_x i + \delta_y j + \delta_z k \quad (2.40)$$

Equating the components of (2.38) to the corresponding components of (2.39) and (2.40) gives

$$d_x = \sum_{n=1}^N n'_{n-1} \cdot (({}^{n-1}\delta \times p'_{n-1}) + {}^{n-1}d) \quad (2.41)$$

$$d_y = \sum_{n=1}^N o'_{n-1} \cdot (({}^{n-1}\delta \times p'_{n-1}) + {}^{n-1}d) \quad (2.42)$$

$$d_z = \sum_{n=1}^N a'_{n-1} \cdot (({}^{n-1}\delta \times p'_{n-1}) + {}^{n-1}d) \quad (2.43)$$

$$\delta_x = \sum_{n=1}^N {}^{n-1}\delta \cdot n'_{n-1} \quad (2.44)$$

$$\delta_y = \sum_{n=1}^N {}^{n-1}\delta \cdot o'_{n-1} \quad (2.45)$$

$$\delta_z = \sum_{n=1}^N {}^{n-1}\delta \cdot a'_{n-1} \quad (2.46)$$

Substituting $n'_{n-1}, o'_{n-1}, a'_{n-1}$ and p'_{n-1} from equation (2.36), then the above equations can be expanded as

$$\begin{aligned}
d_x = & \sum_{n=1}^N [n_{n-1,x}^0 n^{n-1} \delta_y(-p_{n-1}^0 \cdot a_{n-1}^0) - n_{n-1,x}^0 n^{n-1} \delta_z(-p_{n-1}^0 \cdot o_{n-1}^0) \\
& + n_{n-1,x}^0 n^{n-1} d_x + o_{n-1,x}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot n_{n-1}^0) \\
& - o_{n-1,x}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot a_{n-1}^0) + o_{n-1,x}^0 n^{n-1} d_y \\
& + a_{n-1,x}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot o_{n-1}^0) - a_{n-1,x}^0 n^{n-1} \delta_y(-p_{n-1}^0 \cdot n_{n-1}^0) \\
& + a_{n-1,x}^0 n^{n-1} d_z] \tag{2.47}
\end{aligned}$$

$$\begin{aligned}
d_y = & \sum_{n=1}^N [n_{n-1,y}^0 n^{n-1} \delta_y(-p_{n-1}^0 \cdot a_{n-1}^0) - n_{n-1,y}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot o_{n-1}^0) \\
& + n_{n-1,y}^0 n^{n-1} d_x + o_{n-1,y}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot n_{n-1}^0) \\
& - o_{n-1,y}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot a_{n-1}^0) + o_{n-1,y}^0 n^{n-1} d_y \\
& + a_{n-1,y}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot o_{n-1}^0) - a_{n-1,y}^0 n^{n-1} \delta_y(-p_{n-1}^0 \cdot n_{n-1}^0) \\
& + a_{n-1,y}^0 n^{n-1} \delta_z] \tag{2.48}
\end{aligned}$$

$$\begin{aligned}
d_z = & \sum_{n=1}^N [n_{n-1,z}^0 n^{n-1} \delta_y(-p_{n-1}^0 \cdot a_{n-1}^0) - n_{n-1,z}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot o_{n-1}^0) \\
& + n_{n-1,z}^0 n^{n-1} d_x + o_{n-1,z}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot n_{n-1}^0) \\
& - o_{n-1,z}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot a_{n-1}^0) + o_{n-1,z}^0 n^{n-1} d_y \\
& + a_{n-1,z}^0 n^{n-1} \delta_x(-p_{n-1}^0 \cdot o_{n-1}^0) - a_{n-1,z}^0 n^{n-1} \delta_y(-p_{n-1}^0 \cdot n_{n-1}^0) \\
& + a_{n-1,z}^0 n^{n-1} d_z] \tag{2.49}
\end{aligned}$$

$$\delta_x = \sum_{n=1}^N [n_{n-1,x}^0 n^{n-1} \delta_x + o_{n-1,x}^0 n^{n-1} \delta_y + a_{n-1,x}^0 n^{n-1} \delta_z] \tag{2.50}$$

$$\delta_y = \sum_{n=1}^N [n_{n-1,y}^0 n^{n-1} \delta_x + o_{n-1,y}^0 n^{n-1} \delta_y + a_{n-1,y}^0 n^{n-1} \delta_z] \tag{2.51}$$

$$\delta_z = \sum_{n=1}^N [n_{n-1,z}^0 n^{n-1} \delta_x + o_{n-1,z}^0 n^{n-1} \delta_y + a_{n-1,z}^0 n^{n-1} \delta_z] \tag{2.52}$$

According to (2.25) and (2.26), we can further expand (2.47) to (2.52) in terms of the $5N$ link parameter errors. Finally the differential translation and rotation vectors of Δ , d and δ , defining the relationship between the end-effector frame and

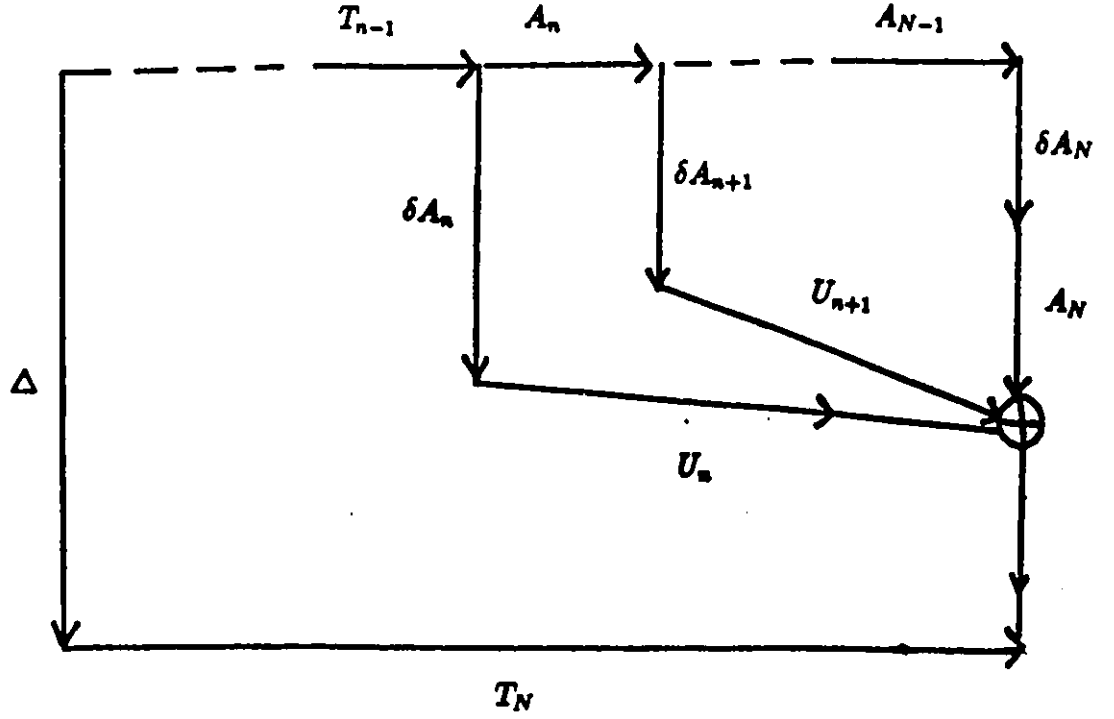


Figure 2.1: general error propagation graph

base frame can be simplified as a linear function of the $5N$ link parameter errors as

$$\begin{aligned}
 d = & \sum_{n=1}^N \{ [p_{n-1}^0 \times (R_{n-1}^0 * K1_n)] \Delta \theta_n + (R_{n-1}^0 * K1_n) \Delta d_n + (R_{n-1}^0 * K2_n) \Delta a_n \\
 & + [(p_{n-1}^0 \times (R_{n-1}^0 * K2_n)) + (R_{n-1}^0 * K3_n)] \Delta \alpha_n \\
 & + [(p_{n-1}^0 \times (R_{n-1}^0 * K5_n)) + (R_{n-1}^0 * K4_n)] \Delta \beta_n \} \quad (2.53)
 \end{aligned}$$

$$\delta = \sum_{n=1}^N \{ (R_{n-1}^0 * K1_n) \Delta \theta_n + (R_{n-1}^0 * K2_n) \Delta \alpha_n + (R_{n-1}^0 * K5_n) \Delta \beta_n \} \quad (2.54)$$

The general positioning and orientational error propagation model for an end-effector of a N degrees-of-freedom robot manipulator is described by equations (2.53) and (2.54) for which the transformation graph is shown in Figure 2.1.

The error propagation relationship from link parameter errors of each link coordinate frame to the end-effector of a robot manipulator may be obtained directly from this general error propagation graph. The greatest advantage of this figure is that the general error propagation model for a robot manipulator can be obtained directly from this figure without the need to go through the tedious deriving procedures described before in this section.

For the first order approximation, from equations (2.38), (2.53) and (2.54) the error matrix transformation can be defined as

$$\begin{aligned} \Delta &= \begin{bmatrix} 0 & -\delta_z & \delta_y & d_x \\ \delta_z & 0 & -\delta_x & d_y \\ -\delta_y & \delta_x & 0 & d_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \Delta R & d \\ 0 & 0 \end{bmatrix} \end{aligned} \quad (2.55)$$

where ΔR is a 3×3 differential rotation matrix defined by differential rotation vector δ and d is a 3×1 differential translation matrix.

Substituting equation (2.55) into (2.32) gives

$$dT_N^0 = \begin{bmatrix} \Delta R & d \\ 0 & 0 \end{bmatrix} * T_N^0 \quad (2.56)$$

which results in

$$dT_N^0 = \begin{bmatrix} \Delta R * R_N^0 & \delta \times P_N^0 + d \\ 0 & 0 \end{bmatrix} \quad (2.57)$$

From the above equation, it can be seen that the first order approximation of the differential rotation is defined by ΔR (i.e., δ); therefore, equation (2.54) gives the first order orientational errors of the end-effector with respect to base frame.

From kinematics, it is known that the positioning error of the end-effector, dP , is influenced by both positioning and orientational errors. From equation (2.57), the positioning error of the end-effector is determined as

$$dP = \delta \times P_N^0 + d \quad (2.58)$$

By substituting equations (2.53) and (2.54) into (2.58) for d and δ , equation (2.58)

can be rewritten as

$$\begin{aligned}
dP &= \sum_{i=1}^N \{ [(R_{n-1}^0 * K1_n) \times P_N^0 + p_{n-1}^0 \times (R_{n-1}^0 \times K1_n)] \Delta\theta_n \\
&+ [R_{n-1}^0 * K1_n] \Delta d_n + [R_{n-1}^0 * K2_n] \Delta a_n \\
&+ [R_{n-1}^0 * K2_n] \times P_N^0 + p_{n-1}^0 \times (R_{n-1}^0 * K2_n) \Delta\alpha_n \\
&+ [(R_{n-1}^0 * K5_n) \times P_N^0 + p_{n-1}^0 \times (R_{n-1}^0 * K5_n) + R_{n-1}^0 * K4_n] \Delta\beta_n \}
\end{aligned} \tag{2.59}$$

The above equation gives the positioning error of the end-effector with respect to base frame as a linear function of the 5N link parameter errors.

By following Wu's definition [29], the general error propagation models for the positioning and orientational errors of the end-effector can be rewritten in the following linear form

$$[\delta] = [R_\theta] \Delta\theta + [R_\alpha] \Delta\alpha + [R_\beta] \Delta\beta \tag{2.60}$$

$$dP = [L_\theta] \Delta\theta + [L_d] \Delta d + [L_a] \Delta a + [L_\alpha] \Delta\alpha + [L_\beta] \Delta\beta \tag{2.61}$$

where

$$\Delta\theta = [\Delta\theta_1, \dots, \Delta\theta_N]^T \tag{2.62}$$

$$\Delta d = [\Delta d_1, \dots, \Delta d_N]^T \tag{2.63}$$

$$\Delta a = [\Delta a_1, \dots, \Delta a_N]^T \tag{2.64}$$

$$\Delta\alpha = [\Delta\alpha_1, \dots, \Delta\alpha_N]^T \tag{2.65}$$

$$\Delta\beta = [\Delta\beta_1, \dots, \Delta\beta_N]^T \tag{2.66}$$

in which $\Delta\theta_n, \Delta d_n, \Delta a_n, \Delta\alpha_n$ and $\Delta\beta_n$ are link parameter errors for n^{th} link ($n = 1, \dots, N$).

$R_\theta, R_\alpha, R_\beta, L_\theta, L_\alpha, L_\beta, L_d$ and L_a are all $3 \times N$ matrices which are functions of the 5N link parameter errors. The n^{th} column of R_θ, \dots, L_a can be expressed as

$$R_{\theta,n} = R_{n-1}^0 * K1_n \tag{2.67}$$

$$R_{\alpha,n} = R_{n-1}^0 * K2_n \tag{2.68}$$

$$R_{\beta,n} = R_{n-1}^0 * K5_n \tag{2.69}$$

$$L_{\theta,n} = (R_{\theta,n} \times P_N^0) + p_{n-1}^0 \times (R_{n-1}^0 * K1_n) \tag{2.70}$$

$$L_{\alpha,n} = (R_{\alpha,n} \times P_N^0) + p_{n-1}^0 \times (R_{n-1}^0 * K2_n) + (R_{n-1}^0 * K3_n) \tag{2.71}$$

$$L_{\beta,n} = (R_{\beta,n} \times P_N^0) + p_{n-1}^0 \times (R_{n-1}^0 * K5_n) + (R_{n-1}^0 * K4_n) \quad (2.72)$$

$$L_{d,n} = R_{\theta,n} \quad (2.73)$$

$$L_{a,n} = R_{\alpha,n} \quad (2.74)$$

The general error propagation model defined by equations (2.60) and (2.61) will be used to develop the error propagation models of a one degree-of-freedom, a two degrees-of-freedom and a three degrees-of-freedom robot manipulators considered in chapter 4, chapter 5 and chapter 6 respectively.

The final comments for this chapter are :

- This chapter developed an error propagation model, based mainly on the results published by W. K. Veitschegger and Chi-Haur Wu in September 1986 [29].
- The presentation uses a different notation and sequence of mathematical derivation as compared to the publication by W. K. Veitschegger and Chi-Haur Wu [29]. This chapter is thus included in the thesis for creating notation and analytic derivation homogeneity.

Chapter 3

Stochastic Error Propagation Model of a N Degrees-of-Freedom Robot Manipulator

The preceding chapter dealt only with the deterministic error propagation model of a robot manipulator. For a N degrees-of-freedom robot manipulator, the desired error propagation model is a function of the $5N$ link parameter errors and the robot's configuration.

In general, all the $5N$ link parameter errors and the configuration variables (θ_n or d_n) are random and dependent. In this chapter, a stochastic model is developed to analyze the positioning and orientational errors of the end-effector due to the mixed and arbitrarily distributed $5N$ link parameter errors and the robot's configuration variable.

3.1 The Joint Probability of Functions of Random Variables

The developments in this section are based on the change of variables using multiple integral transformation of functions of multiple variables applied to joint probability law of random variables [19],[24],[25]. To quantify the positioning and

orientational errors of the end-effector we have to deal with the joint probability function of the link parameter errors and configuration variables. Then the marginal distribution function of the positioning or orientational errors of the end-effector can be calculated from the corresponding joint probability density function of the link parameter errors and the robot configuration variables.

Let's consider first the following general case of n random variables Y_1, Y_2, \dots, Y_n which are functions of another set of continuous random variables X_1, X_2, \dots, X_n as

$$Y_i = g_i(X_1, X_2, \dots, X_n) \quad \text{for} \quad i = 1, 2, \dots, n \quad (3.1)$$

If the functions $g_i(X_1, X_2, \dots, X_n), i = 1, 2, \dots, n$ have continuous first partial derivatives at all points (x_1, x_2, \dots, x_n) , and the set of equations in (3.1) has exactly one set of solutions, which we denote by

$$X_i = h_i(Y_1, Y_2, \dots, Y_n) \quad \text{for} \quad i = 1, 2, \dots, n \quad (3.2)$$

and the determinant of the Jacobian is

$$\begin{aligned} J(X_1, X_2, \dots, X_n) &= \left| \frac{\partial(g_1, g_2, \dots, g_n)}{\partial(X_1, X_2, \dots, X_n)} \right| \\ &= \begin{vmatrix} \frac{\partial g_1}{\partial X_1} & \frac{\partial g_1}{\partial X_2} & \dots & \frac{\partial g_1}{\partial X_n} \\ \frac{\partial g_2}{\partial X_1} & \frac{\partial g_2}{\partial X_2} & \dots & \frac{\partial g_2}{\partial X_n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \frac{\partial g_n}{\partial X_1} & \frac{\partial g_n}{\partial X_2} & \dots & \frac{\partial g_n}{\partial X_n} \end{vmatrix} \\ &\neq 0 \end{aligned} \quad (3.3)$$

at all points (x_1, x_2, \dots, x_n) , and the random variables are jointly continuous and have continuous joint probability density function except at a finite number of points in the (X_1, X_2, \dots, X_n) domain, then the random variables (Y_1, Y_2, \dots, Y_n) defined by (3.1) are jointly continuous with a joint probability function given by

$$\begin{aligned} &f_{Y_1, Y_2, \dots, Y_n}(y_1, y_2, \dots, y_n) \\ &= f_{X_1, X_2, \dots, X_n}(x_1, x_2, \dots, x_n) / |J(x_1, x_2, \dots, x_n)| \\ &= f_{X_1, X_2, \dots, X_n}[h_1(y_1, \dots, y_n), \dots, h_n(y_1, \dots, y_n)] / |J[h_1(y_1, \dots, y_n), \dots, h_n(y_1, \dots, y_n)]| \end{aligned} \quad (3.4)$$

For the case that (3.1) has multiple solutions, equation (3.4) can be extended by summing up the separate terms given by (3.4) for each individual solution, as

$$f_{Y_1, Y_2, \dots, Y_n}(y_1, y_2, \dots, y_n) = \begin{cases} \sum_{j_1=1}^{m_1} \sum_{j_2=1}^{m_2} \dots \sum_{j_n=1}^{m_n} f_{X_1, X_2, \dots, X_n}[h_1(y_1, \dots, y_n), \dots, h_n(y_1, \dots, y_n)] / |J[h_1(y_1, \dots, y_n), \dots, h_n(y_1, \dots, y_n)]| & \text{if } m_j > 0; j = 1, 2, \dots, n \\ 0 & \text{if } m_j = 0; j = 1, 2, \dots, n \end{cases} \quad (3.5)$$

where m_j is the order of multiplicity of the X_j variable.

The computation steps in equation (3.4) are :

- calculation of the solutions of (3.2)
- calculation of the Jacobian defined in equation (3.3)
- replacement of variables $X_i, i = 1, 2, \dots, n$, in (3.2) and (3.3) by the solutions of (3.2) which are functions of Y_1, Y_2, \dots, Y_n .
- division of the joint probability density of X_1, X_2, \dots, X_n by the absolute value of the calculated determinant of the Jacobian.

The above approach will be used to establish the stochastic error propagation model of a robot manipulator. The determination of the joint probability density function of Y_1, Y_2, \dots, Y_n using (3.5) can be quite complicated, both because of the necessity of finding the function of $h_i(y_1, y_2, \dots, y_n), i = 1, 2, \dots, n$, and because of the often complicated form of the matrix whose determinant must be found in order to calculate the Jacobian $J[h_1(y_1, y_2, \dots, y_n), \dots, h_n(y_1, y_2, \dots, y_n)]$.

Therefore, it is essential to reduce the complexity of the mathematical expressions by choosing a suitable set of supplemental variables (Y_2, Y_3, \dots, Y_n), as well as to resort to computers for most of the tedious computations necessary in the analysis.

3.2 Distributions of The Link Parameter Errors

From equations (2.60) and (2.61), it is known that for a N degrees-of-freedom robot manipulator, the positioning and orientational errors of the end-effector are dependent on 6N random variables which are N configuration random variable θ_i ; $i=1,2,\dots,N$ and the 5N link parameter errors.

Due to the gravity effect, there are different degrees of bias of the distribution functions of joint variable errors existing in different sub-workvolume of each joint variable. Consequently, we have to find a suitable distribution to describe the configuration dependent distributions of the N joint variable errors. In searching for the appropriate distributions to represent the 5N link parameter errors, it was found that only the N joint variable errors are configuration dependent and affected generally by gravity because of the compliance and static backlash effects existing in the gear driving train and thus not properly modelled by a normal distribution [1]. Furthermore, the other 4N link parameter errors are so called geometric parameter errors which can be accurately modeled as normal (Gaussian) distributions [16],[29],[30],[33],[34].

For the appropriate distributions to represent the joint variable errors, we have searched many other distributions such as exponential, Poisson, bi-normal, Weibull and gamma distributions. Our final decision on choosing the beta distribution is based on its flexibility. The class of beta distributions is a class of distributions that includes the uniform distribution and is rich enough to provide models for most random variables having a restricted range of possible values. Therefore, it is chosen to model the differently biased distributions of the N joint variable errors existing in different sub-workvolume of each joint variable.

3.2.1 Properties of Beta Distributions

If a random variable X has a beta distribution, then the probability density function $f_X(x)$ of X has the form

$$f_X(x) = \begin{cases} \frac{x^{r-1}(1-x)^{s-1}}{B(r,s)} & \text{if } 0 \leq x \leq 1 \\ 0 & \text{if } x < 0 \text{ or } x > 1 \end{cases} \quad (3.6)$$

where "r" and "s" are positive numbers and are called the parameters of the beta distribution. The constant $B(r,s)$, the beta function [17], is defined as

$$B(r,s) = \frac{\Gamma(r)\Gamma(s)}{\Gamma(r+s)} \quad (3.7)$$

It is known that when $r \geq 0, s \geq 0$

$$B(r,s) = \int_0^1 u^{r-1}(1-u)^{s-1} du \quad (3.8)$$

Thus, when the random variable X has a beta distribution with parameters r and s , then the expected value of X^k can be calculated as

$$\begin{aligned} E(X^k) &= \frac{1}{B(r,s)} \int_0^1 X^{r+k-1}(1-x)^{s-1} dX \\ &= \frac{\Gamma(r+k)\Gamma(r+s)}{\Gamma(r)\Gamma(r+s+k)} \end{aligned} \quad (3.9)$$

where Γ is called gamma function [17].

Letting $k = 1$ and $k = 2$ in the above equation yields

$$E(X) = \frac{\Gamma(r+1)\Gamma(r+s)}{\Gamma(r)\Gamma(r+s+1)} = \frac{r}{r+s} \quad (3.10)$$

$$E(X^2) = \frac{\Gamma(r+2)\Gamma(r+s)}{\Gamma(r)\Gamma(r+s+2)} = \frac{(r+1)r}{(r+s+1)(r+s)} \quad (3.11)$$

and thus

$$\mu_X = E(X) = \frac{r}{r+s} \quad (3.12)$$

$$(\sigma_X)^2 = E(X^2) - (\mu_X)^2 = \frac{rs}{(r+s)^2(r+s+1)} \quad (3.13)$$

from which r and s can be solved as

$$r = \mu_X \left[\frac{\mu_X(1-\mu_X)}{(\sigma_X)^2} - 1 \right] \quad (3.14)$$

$$s = (1-\mu_X) \left[\frac{\mu_X(1-\mu_X)}{(\sigma_X)^2} - 1 \right] \quad (3.15)$$

The importance of equations (3.14) and (3.15) is that if we have many statistically independent observations Y_1, Y_2, \dots, Y_n on random variable Y and we want to fit a beta distribution to these observations, we can estimate the mean value μ_Y by the sample average and the variance σ_Y^2 by the sample variance and then equations (3.14) and (3.15) can be applied to find the beta parameters, r and s , to represent this specific distribution.

3.2.2 Relationship between The Normalized Variable and The Actual Variable

Because the class of beta distribution is defined only for random variables X in $[0,1]$, for a random variable $Y \in [y_m, y_M]$, we use the normalized parameter

$$X = \frac{Y - y_m}{y_M - y_m} \quad (3.16)$$

which has possible values between 0 and 1. The beta distribution that describes the variability of X can be used to obtain the distribution of Y based on the solution of (3.16)

$$Y = (y_M - y_m)X + y_m \quad (3.17)$$

From this linear equation and the corresponding Figure 3.1, we can calculate the cumulative distribution function $F_Y(y)$ of Y by the transformed variable X as follows

$$F_Y(y) = F_X\left(\frac{y - y_m}{y_M - y_m}\right) \quad (3.18)$$

Deriving it with respect to Y gives

$$f_Y(y) = f_X\left(\frac{y - y_m}{y_M - y_m}\right) \cdot \frac{1}{y_M - y_m} \quad (3.19)$$

where $f_X\left(\frac{y - y_m}{y_M - y_m}\right)$ is determined from equation (3.6).

This formula gives the relationship between the transformed variable X and the original variable Y , and will be used frequently in later analysis to transform the probability density function of the transformed variable X to the probability of the original variable Y .

3.3 The Stochastic Error Propagation Model of a N Degrees-of-Freedom Robot Manipulator

From the considerations of section 3.1 we can find that the physical meaning of the error propagation model of a robot manipulator is a transformation from $6N$ random variables $(\Delta\theta_i, \Delta d_i, \Delta\alpha_i, \Delta a_i, \Delta\beta_i, \theta_i, i = 1, 2, \dots, N)$ into 6 random variables $(d_X, d_Y, d_Z, \delta_X, \delta_Y, \delta_Z)$.

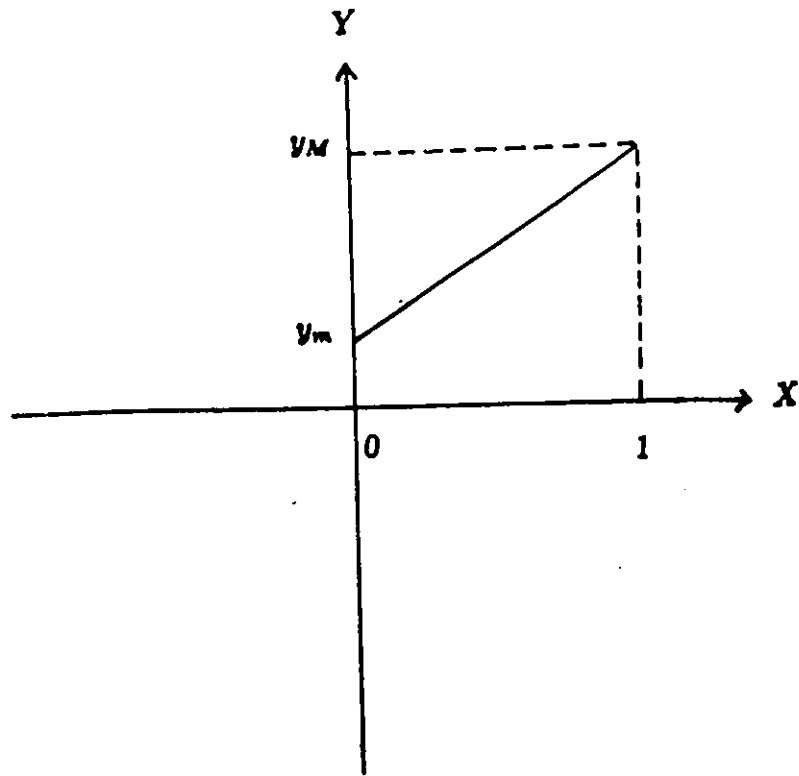


Figure 3.1: linear diagram of X and Y

In this case, $6N - 6$ supplementary output variables have to be defined and later a marginal joint probability density function is calculated by integrating over the whole domain of all the $6N - 6$ supplementary variables in order to get the probability of the end-effector to be located within a defined positioning and orientational error envelope. The choice of the supplementary random variables is arbitrary. A computationally convenient choice is to choose the $6N - 6$ supplementary random output variables identical to a set of $6N - 6$ input random variables.

For a $N = 6$ degrees-of-freedom robot manipulator, the first 6 equations of (3.1) are

$$[Y_1, Y_2, Y_3]^T = dP \quad (3.20)$$

$$[Y_4, Y_5, Y_6]^T = \delta \quad (3.21)$$

where δ and dP are given by equations (2.60) and (2.61) as functions of input variables. The other $6N - 6 = 30$ supplementary equations of (3.1) can be chosen as

$$\begin{aligned} [Y_7, \dots, Y_{12}] &= [\theta_1, \dots, \theta_6] \\ &= \theta^T \end{aligned} \quad (3.22)$$

$$[Y_{13}, \dots, Y_{18}] = [\Delta\theta_1, \dots, \Delta\theta_6]$$

$$= \Delta\theta^T \quad (3.23)$$

$$\begin{aligned} [Y_{19}, \dots, Y_{24}] &= [\Delta d_1, \dots, \Delta d_6] \\ &= \Delta d^T \end{aligned} \quad (3.24)$$

$$\begin{aligned} [Y_{25}, \dots, Y_{30}] &= [\Delta a_1, \dots, \Delta a_6] \\ &= \Delta a^T \end{aligned} \quad (3.25)$$

$$\begin{aligned} [Y_{31}, \dots, Y_{36}] &= [\Delta \alpha_1, \dots, \Delta \alpha_6] \\ &= \Delta \alpha^T \end{aligned} \quad (3.26)$$

In this case, the solution (3.2) of the system (3.1) for the random supplementary output variables Y_7 to Y_{36} are obvious

$$[\theta^T, \Delta\theta^T, \Delta d^T, \Delta a^T, \Delta \alpha^T] = [Y_7, Y_8, \dots, Y_{36}] \quad (3.27)$$

By using equation (3.27), the equations (2.60) and (2.61) can be rewritten as

$$\begin{aligned} [Y_1, Y_2, Y_3]^T &= \{L_\theta(Y_7, \dots, Y_{12})[Y_{13}, \dots, Y_{18}]^T \\ &\quad + L_d(Y_7, \dots, Y_{12})[Y_{19}, \dots, Y_{24}]^T \\ &\quad + L_a(Y_7, \dots, Y_{12})[Y_{25}, \dots, Y_{30}]^T \\ &\quad + L_\alpha(Y_7, \dots, Y_{12})[Y_{31}, \dots, Y_{36}]^T \\ &\quad + L_\beta(Y_7, \dots, Y_{12})\Delta\beta\} \end{aligned} \quad (3.28)$$

$$\begin{aligned} [Y_4, Y_5, Y_6] &= \{R_\theta(Y_7, \dots, Y_{12})[Y_{13}, \dots, Y_{18}]^T \\ &\quad + R_a(Y_7, \dots, Y_{12})[Y_{31}, \dots, Y_{36}]^T \\ &\quad + R_\beta(Y_7, \dots, Y_{12})\Delta\beta\} \end{aligned} \quad (3.29)$$

The solutions for the six unknowns $\Delta\beta$ of the six linear in $\Delta\beta$ equations of (3.28) and (3.29) can be denoted as

$$\Delta\beta = [h_1(Y_1, \dots, Y_{36}), \dots, h_6(Y_1, \dots, Y_{36})]^T \quad (3.30)$$

Then in this case the Jacobian defined in (3.3) can be written as

$$\begin{aligned} J(\theta^T, \Delta\theta^T, \Delta d^T, \Delta a^T, \Delta \alpha^T, \Delta\beta^T) \\ = \frac{\partial(dP^T, \delta^T, Y_7, \dots, Y_{36})}{\partial(\theta^T, \Delta\theta^T, \Delta d^T, \Delta a^T, \Delta \alpha^T, h_1, \dots, h_6)} \end{aligned} \quad (3.31)$$

The Jacobian must satisfy the condition $J \neq 0$ so that the six equations defined by equations (3.28) and (3.29) are linearly independent. Therefore, there is only one unique set of solutions given by (3.30). That is the transformation from $X = (\theta^T, \Delta\theta^T, \Delta d^T, \Delta a^T, \Delta\alpha^T, \Delta\beta^T)$ to $Y = (dP^T, \delta^T, Y_7, \dots, Y_{36})$ is a one-to-one transformation. Equation (3.5) then can be rewritten in the following form

$$f_{Y_1, \dots, Y_{36}}(dP^T, \delta^T, y_7, \dots, y_{36}) = \begin{cases} f_{\theta^T, \Delta\theta^T, \Delta d^T, \Delta a^T, \Delta\alpha^T, \Delta\beta^T}(y_7, \dots, y_{36}, h_1(y_1, \dots, y_{36}), \dots, h_6(y_1, \dots, y_{36})) \\ / | J[y_7, \dots, y_{36}, h_1(y_1, \dots, y_{36}), \dots, h_6(y_1, \dots, y_{36})] | \\ 0 \end{cases} \quad \text{if } m_j = 0; j = 1, 2, \dots, 6 \quad (3.32)$$

where m_j is the order of multiplicity of $\Delta\beta_j$ variable.

The end-effector positioning and orientational error distributions at the given positioning and orientational error, dP and δ , can be calculated from the marginal distribution of equation (3.32) by integrating from Y_7 to Y_{36} over the entire domain of these supplementary random variables $\theta, \Delta\theta, \Delta d, \Delta a$ and $\Delta\alpha$ as

$$\begin{aligned} & f_{dP, \delta}(dP^T, \delta^T) \\ &= \int_{Y_7} \dots \int_{Y_{36}} f_{Y_1, \dots, Y_{36}}(dP^T, \delta^T, y_7, \dots, y_{36}) dY_7 \dots dY_{36} \\ &= \int_{\theta} \int_{\Delta\theta} \int_{\Delta d} \int_{\Delta a} \int_{\Delta\alpha} f_{\theta^T, \Delta\theta^T, \Delta d^T, \Delta a^T, \Delta\alpha^T, \Delta\beta^T} \\ & (y_7, \dots, y_{36}, h_1(dP^T, \delta^T, \theta^T, \Delta\theta^T, \Delta d^T, \Delta a^T, \Delta\alpha^T), \dots, h_6(dP^T, \delta^T, \theta^T, \Delta d^T, \Delta a^T, \Delta\alpha^T)) \\ & / | J(y_7, \dots, y_{36}, h_1(dP^T, \delta^T, y_7, \dots, y_{36}), \dots, h_6(dP^T, \delta^T, y_7, \dots, y_{36})) | \\ & d(\theta^T) d(\Delta\theta^T) d(\Delta d^T) d(\Delta a^T) d(\Delta\alpha^T) \end{aligned} \quad (3.33)$$

Equation (3.33) represents the general stochastic error propagation model for any mix of arbitrary distributions existing in the link parameter errors and any random configuration of a robot manipulator. As only the N joint variable errors are configuration dependent, we can further expand (3.33) in the following way.

For a N degrees-of-freedom robot manipulator, if the joint variable q_i is divided into L_i different sub-workvolumes ($i = 1, 2, \dots, N$) such that in each sub-workvolume the gravity effect is approximately the same, then in each individual sub-workvolume all the $6N$ input random variables become independent. Therefore, for a revolute N degrees-of-freedom robot manipulator equation (3.33) can be fur-

ther expanded as

$$\begin{aligned}
& f_{dP,\delta}(dP^T, \delta^T) \\
&= \sum_{i_1=1}^{L_1} \cdots \sum_{i_6=1}^{L_6} \int_{\theta_{i_1}} \cdots \int_{\theta_{i_6}} \int_{\Delta d} \int_{\Delta a} \int_{\Delta \alpha} \\
& f_{\Delta \beta_1} h_1(dP^T, \delta^T, \theta^T, \Delta \theta^T, \Delta d^T, \Delta a^T, \Delta \alpha^T) \cdots f_{\Delta \beta_6} h_6(dP^T, \delta^T, \theta^T, \Delta \theta^T, \Delta d^T, \Delta a^T, \Delta \alpha^T) f_{\theta_1}(y^T) \\
& \cdots f_{\Delta \alpha_6}(y_{36}) d(\theta^T) d(\Delta \theta^T) d(\Delta d^T) d(\Delta a^T) d(\Delta \alpha^T)
\end{aligned} \tag{3.34}$$

For any other type of robot manipulator, we can derive its general stochastic error propagation model using the presented approach. In summary, this chapter made the following important progress in the error propagation analysis of a robot manipulator :

- In this chapter, earlier results presented in [21],[22] and [23] are further developed to the level of an explicit analytical model for the error propagation analysis of a robot manipulator.
- The model documented in this chapter was used for the development of the computer program given in Appendix B. Compared to the previously published robot error propagation models, this is the first general model which can propagate any kind of analytical or empirical probability density function, symmetrical or non-symmetrical, in any mixture. The model can thus be used to propagate any experimental results regarding the error distributions.

Chapter 4

Stochastic Error Propagation Analysis of a One Degree-of-Freedom Robot Manipulator

A manipulator structure of the type considered in this dissertation will have errors resulting from the joint compliance and gear backlash effect existing in the gear driving train. Thus, the manipulator's joint error distribution will be affected by the gravity forces of the link members and payload in the end-effector. Thus, the accurate distribution for each link parameter error will be a prerequisite to a successful error propagation analysis. This chapter includes error propagation analysis and stochastic modeling approach for one degree-of-freedom robot manipulator.

4.1 Error Propagation Model of One Degree-of-Freedom Robot Manipulator

A one degree-of-freedom robot manipulator can be mainly considered to be composed of a joint actuator and a link. Its schematic drawing is shown in Figure 3.1 as follows.

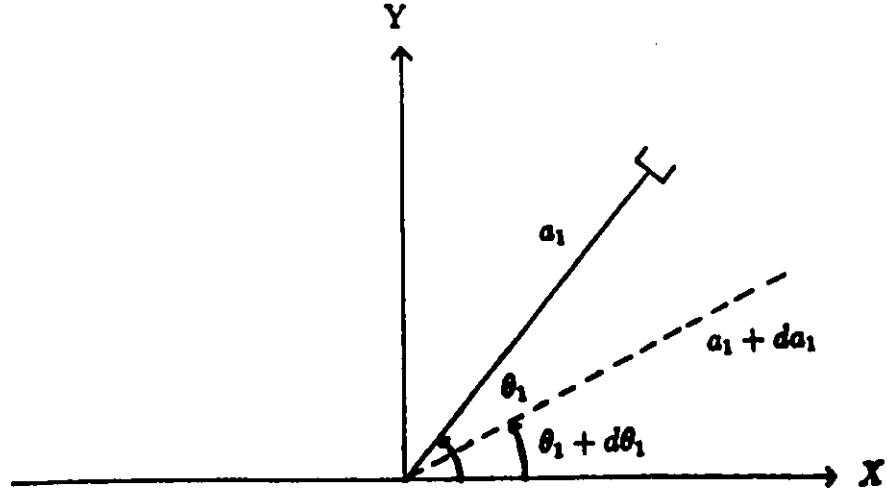


Figure 4.1: schematic drawing of one degree-of-freedom robot manipulator

If the link coordinate frame is assigned according to the D-H convention as shown in Figure 4.1, then the link parameter table can be established as table 4.1.

Table 4.1: link parameter table

link n	variable	α_n	a_n	d_n	β_n
1	θ_1	0	a_1	0	0

The A transformation matrix is defined by equation (A.3) as

$$A_n = \begin{bmatrix} C\theta_n C\beta_n - S\theta_n S\alpha_n S\beta_n & -S\theta_n C\alpha_n & C\theta_n S\beta_n + S\theta_n S\alpha_n C\beta_n & a_n C\theta_n \\ S\theta_n C\beta_n + C\theta_n S\alpha_n S\beta_n & C\theta_n C\alpha_n & S\theta_n S\beta_n - C\theta_n S\alpha_n C\beta_n & a_n S\theta_n \\ -C\alpha_n S\beta_n & S\alpha_n & C\alpha_n C\beta_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

Substituting the link parameter values from Table 4.1 into equation (4.1) gives

$$A_1 = \begin{bmatrix} C_1 & -S_1 & 0 & a_1 C_1 \\ S_1 & C_1 & 0 & a_1 S_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

By applying equations (2.67) to (2.74), we can immediately calculate the matrices R_θ, \dots, L_β as follows :

$$\begin{aligned} R_{\theta,1} &= R_0^0 * K1_1 \\ &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned} \quad (4.3)$$

$$\begin{aligned} R_{\alpha,1} &= R_0^0 * K2_1 \\ &= \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix} \end{aligned} \quad (4.4)$$

$$\begin{aligned} R_{\beta,1} &= R_0^0 * K5_1 \\ &= \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix} \end{aligned} \quad (4.5)$$

$$\begin{aligned} L_{\theta,1} &= (R_{\theta,1} \times P_1^0) + P_0^0 \times (R_0^0 * K1_1) \\ &= \begin{bmatrix} -a_1 S_1 \\ a_1 C_1 \\ 0 \end{bmatrix} \end{aligned} \quad (4.6)$$

$$\begin{aligned} L_{\alpha,1} &= (R_{\alpha,1} \times P_1^0) + P_0^0 \times (R_0^0 * K2_1) + (R_0^0 * K3_1) \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \end{aligned} \quad (4.7)$$

$$\begin{aligned} L_{\beta,1} &= (R_{\beta,1} \times P_1^0) + P_0^0 \times (R_0^0 * K5_1) + (R_0^0 * K4_1) \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \end{aligned} \quad (4.8)$$

$$\begin{aligned} L_{\alpha,1} &= R_{\alpha,1} \\ &= \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix} \end{aligned} \quad (4.9)$$

$$\begin{aligned}
L_{d,1} &= R_{\theta,1} \\
&= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\end{aligned} \tag{4.10}$$

Thus the orientational and positioning errors are given by equations (2.60) and (2.61) as

$$[\delta] = [R_\theta]\Delta\theta + R_\alpha\Delta\alpha + R_\beta\Delta\beta \tag{4.11}$$

$$dP = [L_\theta]\Delta\theta + [L_d]\Delta d + [L_a]\Delta a + [L_\alpha]\Delta\alpha + [L_\beta]\Delta\beta \tag{4.12}$$

Substituting the calculated R_θ, \dots, L_β into the above orientational and positioning error propagation equations gives

$$\delta = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \Delta\theta_1 + \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix} \Delta\alpha_1 + \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix} \Delta\beta_1 \tag{4.13}$$

$$dP = \begin{bmatrix} -a_1 S_1 \\ a_1 C_1 \\ 0 \end{bmatrix} \Delta\theta_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \Delta d_1 + \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix} \Delta a_1 \tag{4.14}$$

where δ and dP are all 3×1 matrices. $\Delta\theta_1, \Delta\alpha_1, \Delta\beta_1, \Delta d_1$ and Δa_1 are link parameter errors. The above two equations fully define the error propagation relationship between the end-effector and five link parameter errors and will be used in the stochastic error propagation analysis in the next section.

4.2 Stochastic Error Propagation Model of a One Degree-of-Freedom Robot Manipulator

Applying the formulas (4.13) and (4.14) to a model of one degree-of-freedom robot manipulator specified in Fig. 4.1, the equations identifying the Cartesian positioning and orientational errors of the end-effector with respect to the base frame are expressed as

$$dX = (-a_1 S_1)\Delta\theta_1 + (C_1)\Delta a_1 \tag{4.15}$$

$$dY = (a_1 C_1)\Delta\theta_1 + (S_1)\Delta a_1 \tag{4.16}$$

$$dZ = \Delta d_1 \quad (4.17)$$

$$\delta_X = (C_1)\Delta\alpha_1 + (-S_1)\Delta\beta_1 \quad (4.18)$$

$$\delta_Y = (S_1)\Delta\alpha_1 + (C_1)\Delta\beta_1 \quad (4.19)$$

$$\delta_Z = \Delta\theta_1 \quad (4.20)$$

First, the X component of the Cartesian positioning error is taken into consideration. To evaluate the distribution of dX , the deriving procedure is started with choosing the output random parameters as

$$Y_1 = dX = (-a_1 S_1)\Delta\theta_1 + (C_1)\Delta a_1 \quad (4.21)$$

$$Y_2 = \theta_1 \quad (4.22)$$

$$Y_3 = \Delta\theta_1 \quad (4.23)$$

for which we can define the input random parameters as

$$X_1 = \theta_1 \quad (4.24)$$

$$X_2 = \Delta\theta_1 \quad (4.25)$$

$$X_3 = \Delta a_1 = \frac{Y_1 + (a_1 S(Y_2))Y_3}{C(Y_2)} \quad ; c(Y_2) \neq 0 \quad (4.26)$$

The joint probability of $Y_1, Y_2,$ and Y_3 can be found from the joint probability of X_1, X_2 and X_3 using the joint probability formula (3.5). The distribution function of the positioning error dX can be found by the marginal distribution method described in (3.33). It is obvious that the transformation between (Y_1, Y_2, Y_3) and (X_1, X_2, X_3) is a one to one transformation . In this case the determinant of the Jacobian defined by (3.3) becomes

$$\begin{aligned} J(X_1, X_2, X_3) &= \left| \frac{\partial(g_1, g_2, g_3)}{\partial(X_1, X_2, X_3)} \right| \\ &= \begin{vmatrix} -a_1 C_1 \Delta\theta_1 - S_1 \Delta a_1 & -a_1 S - 1 & C_1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{vmatrix} \\ &= C_1 \end{aligned} \quad (4.27)$$

where the Jacobian must satisfy the condition $J = C_1 \neq 0$. Then the joint probability density of Y_1, Y_2 and Y_3 can be calculated from (3.5) as

$$f_{Y_1, Y_2, Y_3}(y_1, y_2, y_3) = f_{X_1, X_2, X_3}(y_2, y_3, \frac{y_1 + (a_1 S y_2) y_3}{C y_2}) / |c y_2| \quad (4.28)$$

Then the distribution of Cartesian positioning error dX can be found from (3.33)

as

$$\begin{aligned}
f_{Y_1}(y_1) &= f_{dX}(y_1) \\
&= \int_{Y_2} \int_{Y_3} f_{X_1, X_2, X_3}(y_2, y_3, \frac{y_1 + (a_1 s y_2) y_3}{C y_2}) / |C y_2| dY_2 dY_3 \\
&= \int_{\theta_1} \int_{\Delta\theta_1} f_{\theta_1, \Delta\theta_1, \Delta\alpha_1}(\theta_1, \Delta\theta_1, \frac{y_1 + (a_1 S_1) \Delta\theta_1}{C_1}) / |C_1| d(\theta_1) d(\Delta\theta_1)
\end{aligned} \tag{4.29}$$

The derivation procedures for the error distribution of dX , dY , dZ , δ_X , δ_Y and δ_Z are the same. By using the same procedures as above, the error distribution functions for dY , dZ , δ_X , δ_Y , and δ_Z can be derived as follows.

$$\begin{aligned}
f_{dY}(dY) &= \int_{\theta_1} \int_{\Delta\theta_1} f_{\theta_1, \Delta\theta_1, \Delta\alpha_1}(\theta_1, \Delta\theta_1, \frac{dy - a_1 C_1 \Delta\theta_1}{S_1}) / |S_1| \cdot d(\theta_1) \cdot d(\Delta\theta_1) \\
&\quad ; S_1 \neq 0
\end{aligned} \tag{4.30}$$

$$f_{dZ}(dz) = f_{\Delta d_1}(\Delta d_1) \tag{4.31}$$

$$\begin{aligned}
f_{\delta_X}(\delta_x) &= \int_{\theta_1} \int_{\Delta\alpha_1} f_{\theta_1, \Delta\alpha_1, \Delta\beta_1}(\theta_1, \Delta\alpha_1, \frac{\delta_x - (C_1) \Delta\alpha_1}{-S_1}) / |-S_1| \cdot d(\theta_1) \cdot d(\Delta\alpha_1) \\
&\quad ; S_1 \neq 0
\end{aligned} \tag{4.32}$$

$$\begin{aligned}
f_{\delta_Y}(\delta_y) &= \int_{\theta_1} \int_{\Delta\alpha_1} f_{\theta_1, \Delta\alpha_1, \Delta\beta_1}(\theta_1, \Delta\alpha_1, \frac{\delta_y - (S_1) \Delta\alpha_1}{C_1}) / |C_1| \cdot d(\theta_1) \cdot d(\Delta\alpha_1) \\
&\quad ; C_1 \neq 0
\end{aligned} \tag{4.33}$$

$$f_{\delta_Z}(\delta_z) = f_{\Delta\theta_1}(\Delta\theta_1) \tag{4.34}$$

Because of the effect of random backlash and joint compliance, the joint variable errors become configuration dependent. Thus, under the gravity effect, the joint variable errors take different degrees of biased distributions in different sub-workvolume. If the domain of joint variable θ_1 can be divided into L_1 smaller sub-domains such that in each sub-domain the gravity effect is approximately the same, then in each sub-domain the joint variable errors become configuration independent and can be approximated as some fixed forms of distributions.

Therefore, if the domain of joint variable θ_1 can be divided into L_1 independent sub-domains, then by using (3.34) equations (4.21) to (4.26) can be further expanded as

$$f_{dx}(dx) = \sum_{i=1}^{L_1} \int_{\theta_{1,i}} \int_{\Delta\theta_1} f_{\theta_1}(\theta_1) f_{\Delta\theta_1}(\Delta\theta_1) f_{\Delta\alpha_1} \left(\frac{dx + (\alpha_1 S_1) \Delta\theta_1}{C_1} \right) / |C_1| \cdot d(\theta_1) \cdot d(\Delta\theta_1) \quad ; C_1 \neq 0 \quad (4.35)$$

$$f_{dy}(dy) = \sum_{i=1}^{L_1} \int_{\theta_{1,i}} \int_{\Delta\theta_1} f_{\theta_1}(\theta_1) f_{\Delta\theta_1}(\Delta\theta_1) f_{\Delta\alpha_1} \left(\frac{dy - \alpha_1 C_1 \Delta\theta_1}{S_1} \right) / |S_1| \cdot d(\theta_1) d(\Delta\theta_1) \quad ; S_1 \neq 0 \quad (4.36)$$

$$f_{dz}(dz) = f_{\Delta d_1}(\Delta d_1) \quad (4.37)$$

$$f_{\delta_x}(\delta_x) = \int_{\theta_1} \int_{\Delta\alpha_1} f_{\theta_1}(\theta_1) f_{\Delta\alpha_1}(\Delta\alpha_1) f_{\Delta\beta_1} \left(\frac{\delta_x - C_1 \Delta\alpha_1}{-S_1} \right) / |S_1| \cdot d(\theta_1) \cdot d(\Delta\alpha_1) \quad ; S_1 \neq 0 \quad (4.38)$$

$$f_{\delta_y}(\delta_y) = \int_{\theta_1} \int_{\Delta\alpha_1} f_{\theta_1}(\theta_1) f_{\Delta\alpha_1}(\Delta\alpha_1) f_{\Delta\beta_1} \left(\frac{\delta_y - S_1 \Delta\alpha_1}{C_1} \right) / |C_1| \cdot d(\theta_1) \cdot d(\Delta\alpha_1) \quad ; C_1 \neq 0 \quad (4.39)$$

$$f_{\delta_z}(\delta_z) = \sum_{i=1}^{L_1} \int_{\theta_{1,i}} f_{\theta_1}(\theta_1) f_{\Delta\theta_1}(\Delta\theta_1) d(\theta_1) \quad (4.40)$$

where $f_{\theta_1}(\theta_1)$, $f_{\Delta\theta_1}(\Delta\theta_1)$, $f_{\Delta d_1}(\Delta d_1)$, $f_{\Delta\alpha_1}(\Delta\alpha_1)$, $f_{\Delta\alpha_1}(\Delta\alpha_1)$ and $f_{\Delta\beta_1}(\Delta\beta_1)$ are probability density functions corresponding to input random variables.

4.3 Numerical Stochastic Simulations

Equations (4.35) to (4.40) completely define the distributions for the Cartesian positioning and orientational errors of the end-effector. To illustrate the results without extra experimental burden, some numerical simulations based on the following assumptions is presented.

1. joint 1 is limited to $\pm 90^\circ$.

2. In near horizontal area $\theta_1 \in [-15^\circ, 15^\circ]$, the joint variable error $d\theta_1$ takes a significant biased beta distribution with beta parameters $r_1 = 2$ and $s_1 = 10$ because of the great effect of gravity.
3. In near vertical area $\theta_1 \in [75^\circ, 90^\circ]$ or $\theta_1 \in [-90^\circ, -75^\circ]$, the joint variable error $d\theta_1$ is assumed uniformly distributed because the encoder's output error takes a uniform distribution and the gravity effect can be neglected when the mechanical arm is in near vertical position.
4. In the other domain of θ_1 , the joint variable error is assumed to take a less biased beta distribution with the beta parameters $r_2 = 2$ and $s_2 = 4$.
5. $\sigma_{\Delta\theta_1} = 0.1^\circ$; $\Delta\theta_1$ is assumed to be located within the 5σ band.
6. all the other link parameter errors are assumed to their nominal values with zero variance, i.e., $\sigma_{\text{triangle}a_1} = \sigma_{\Delta\beta_1} = 0^\circ$ and $\sigma_{\Delta a_1} = \sigma_{\Delta d_1} = 0\text{m}$.

The joint variable error distributions based on the above assumptions are shown in Figure 4.2. The simulation results comparing the differences between this general mixed error propagation approach (represented by the curves with the black dots) and the previous work based only on the assumption that all link parameter errors are normally distributed with zero mean (represented by the curves with triangular markers) are shown in Figures 4.3 to 4.14. For example, in near horizontal area (see Figures 4.3, 4.7 and 4.11), the positioning and orientational error distribution functions of the end-effector take a more biased distribution which is resulted from the significant effect of gravity in this area. Even in near vertical area (see Figures 4.5, 4.9 and 4.13), the presented model results in a symmetric but wider range error distribution. It is obvious that the presented general stochastic error propagation model is more general and realistic, because it considers the arbitrarily mixed error propagation phenomena existing in a one degree-of-freedom robot manipulator and can be applied to obtain the exact positioning or orientational error distribution function of the end-effector. The importance of this general stochastic error propagation model is that it can be applied to find the mean value of the positioning and orientational errors of the end-effector in each sub-workvolume to improve the absolute accuracy of the robot manipulator.

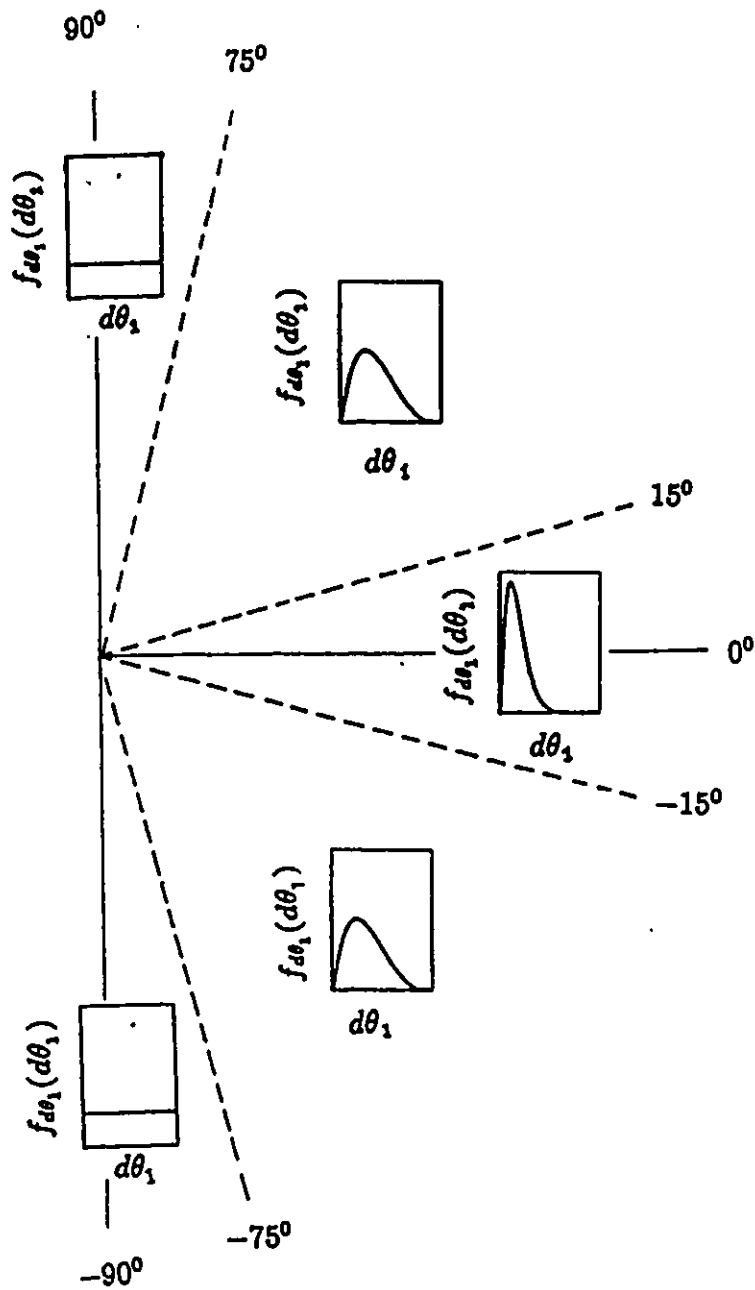


Figure 4.2: $d\theta_1$ distributions in different sub-domains of θ_1 ; $\theta_1 \in [-90^\circ, 90^\circ]$.

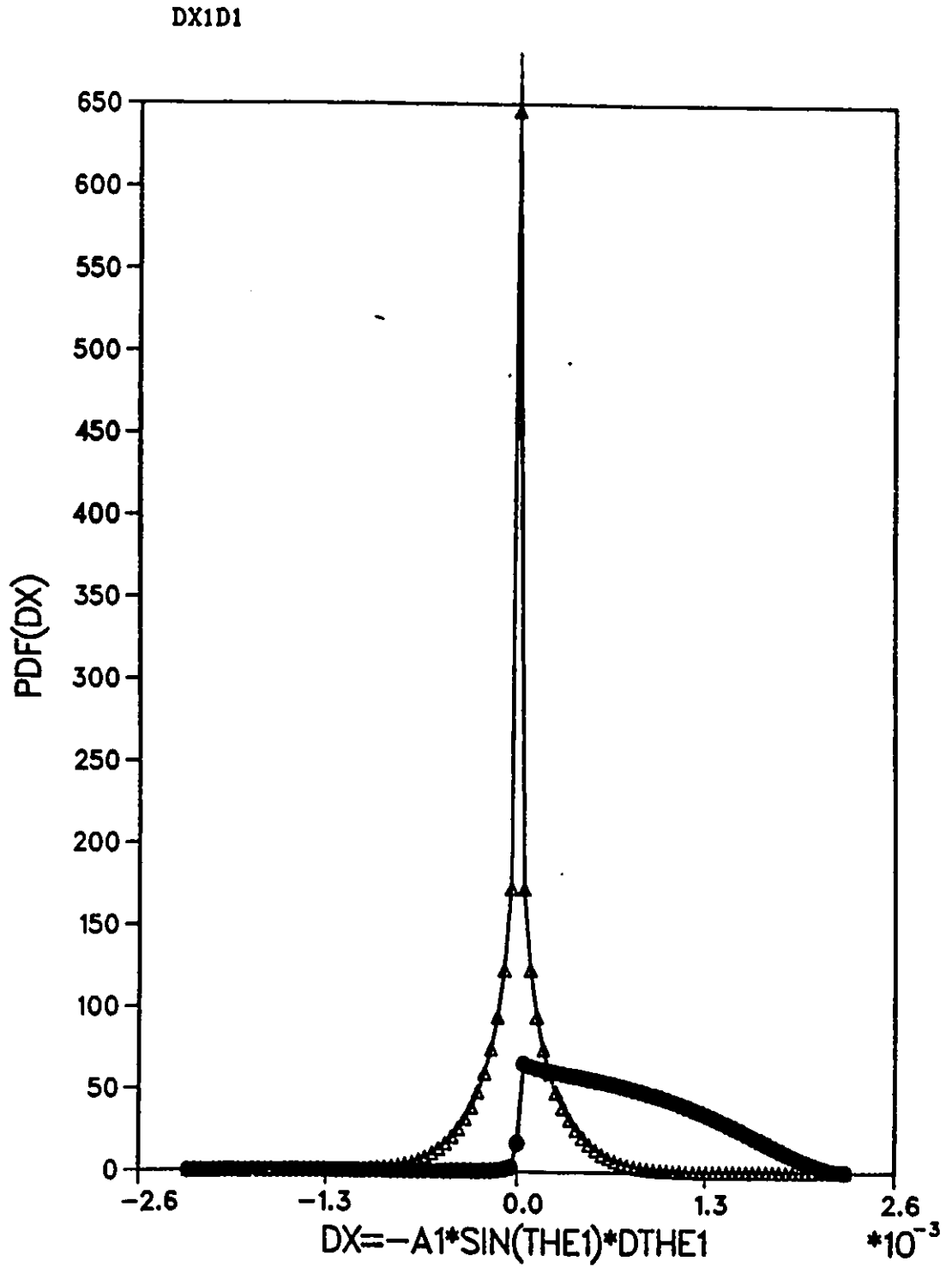


Figure 4.3: dX distribution for which $\theta_1 \in [0^\circ, 15^\circ]$.

DX1D2

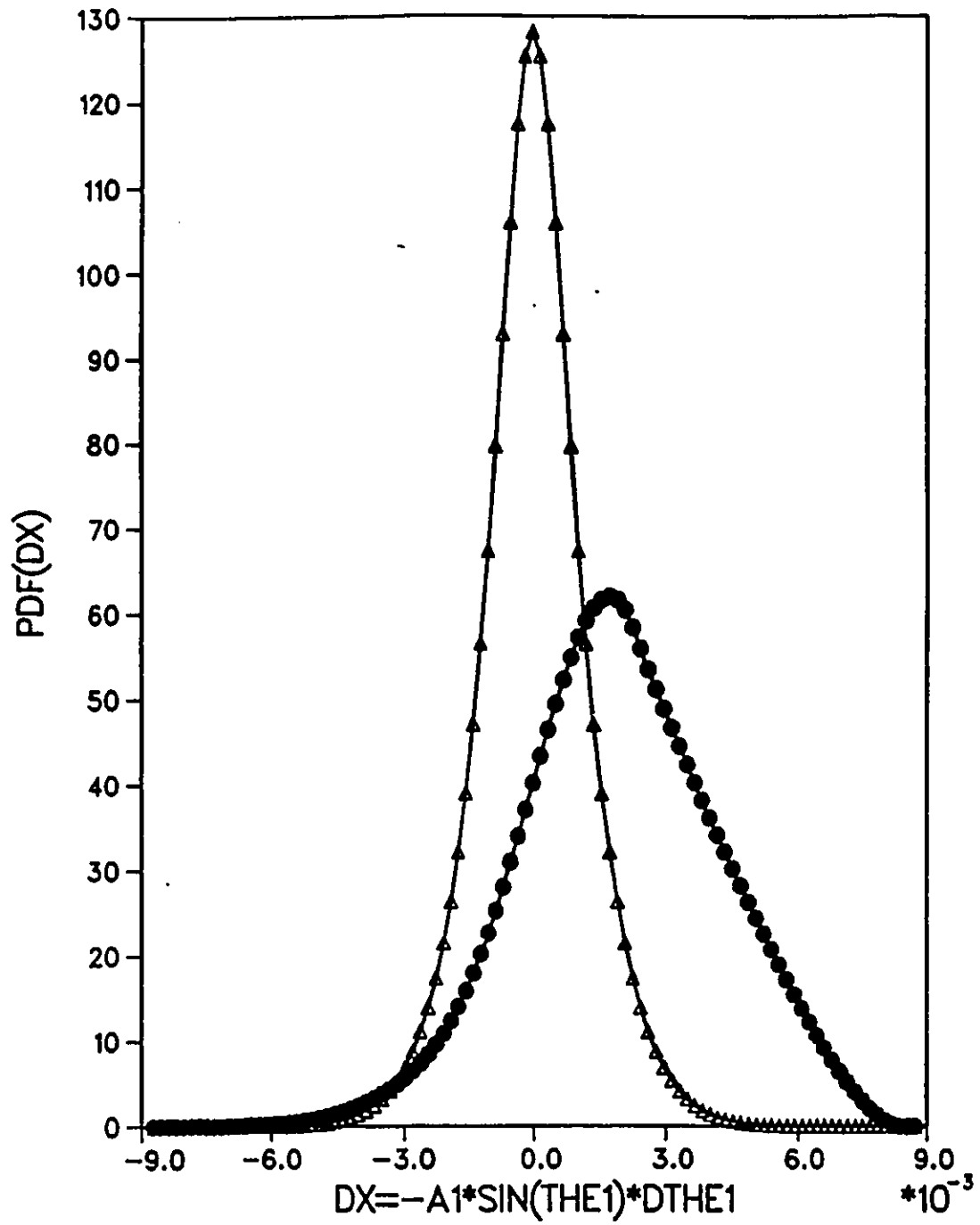


Figure 4.4: dX distribution for which $\theta_1 \in (15^\circ, 75^\circ)$.

DX1D3

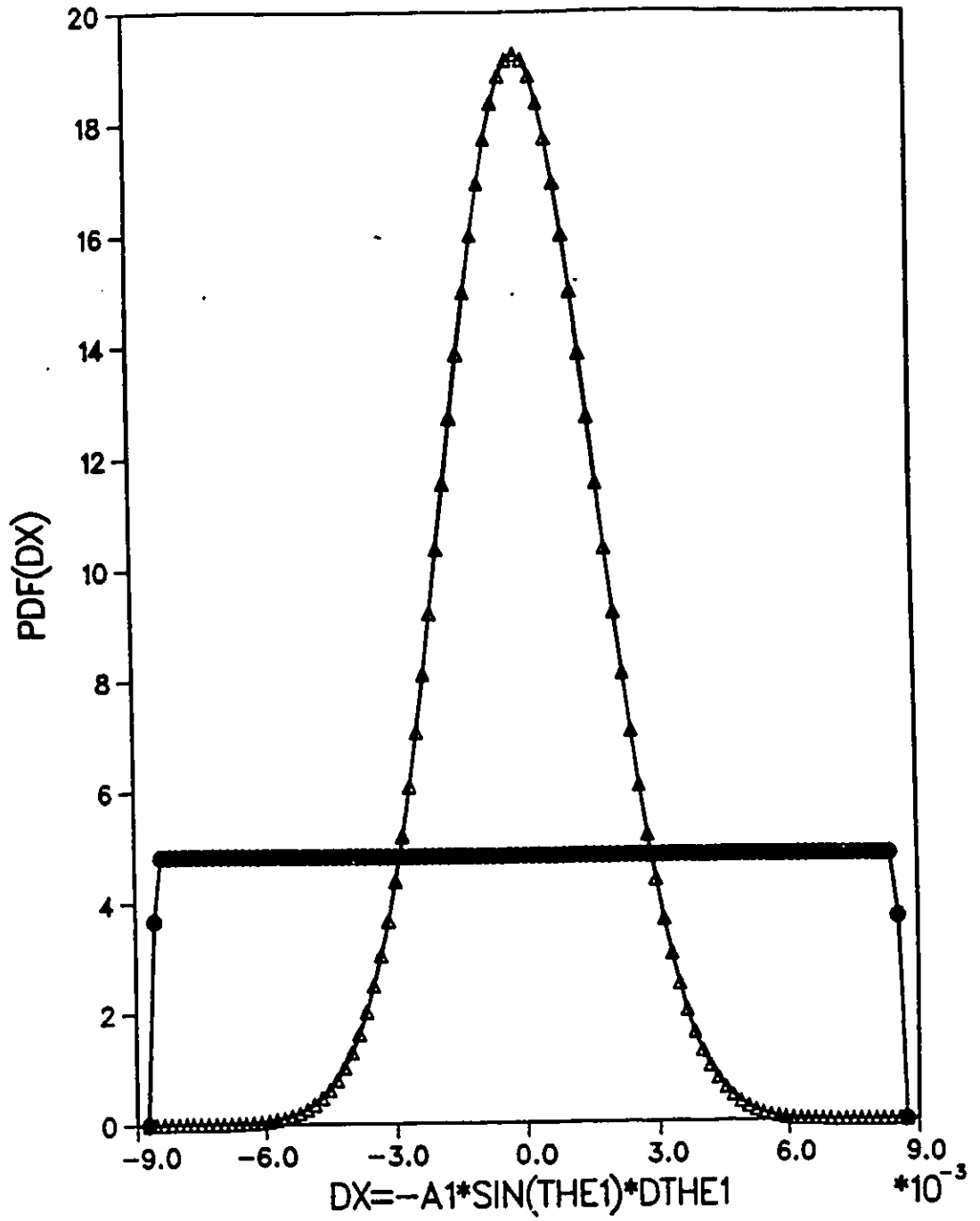


Figure 4.5: dX distribution for which $\theta_1 \in [75^\circ, 90^\circ]$.

DX1D

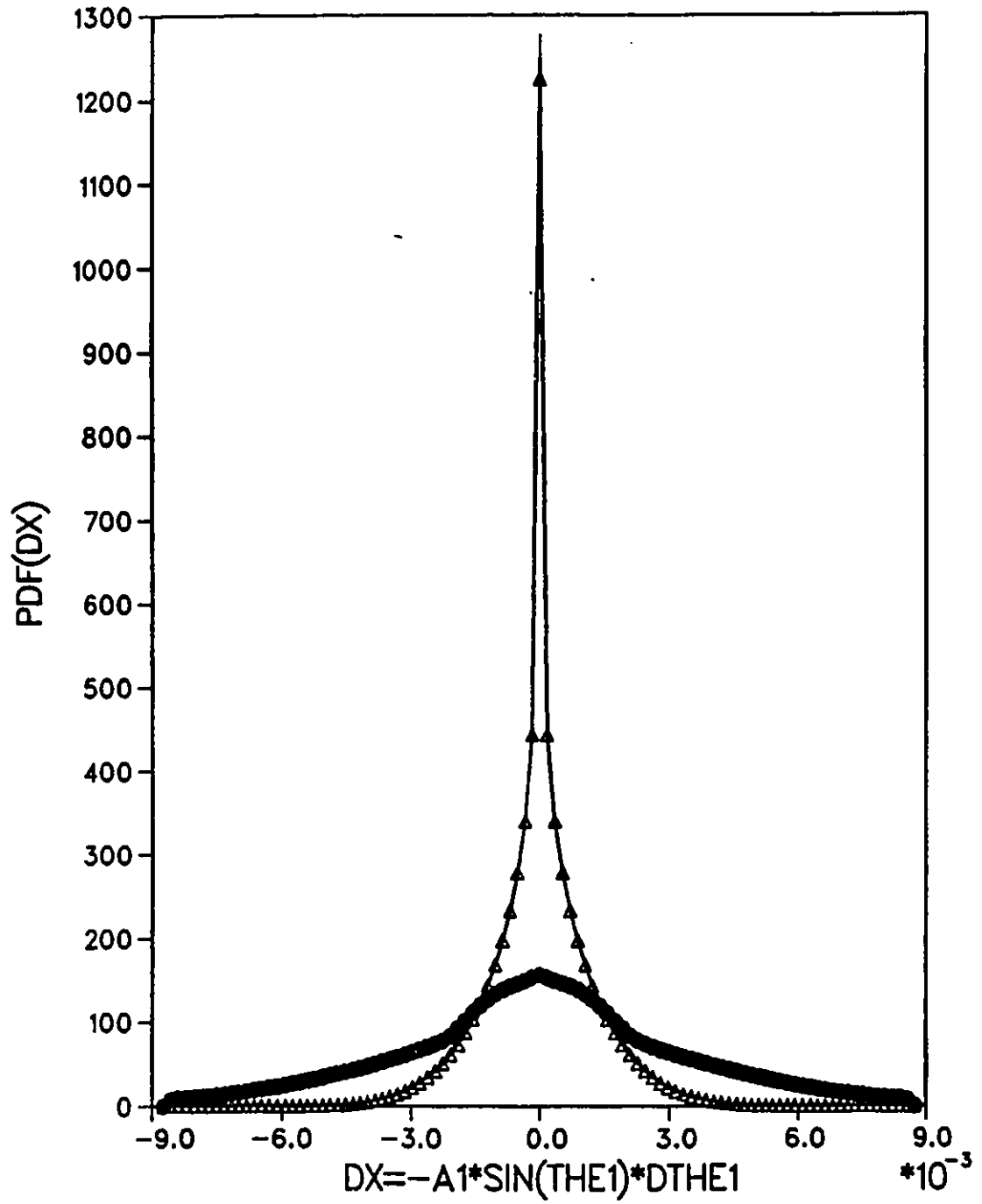


Figure 4.6: dX distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$ (the whole domain)

DY1D1

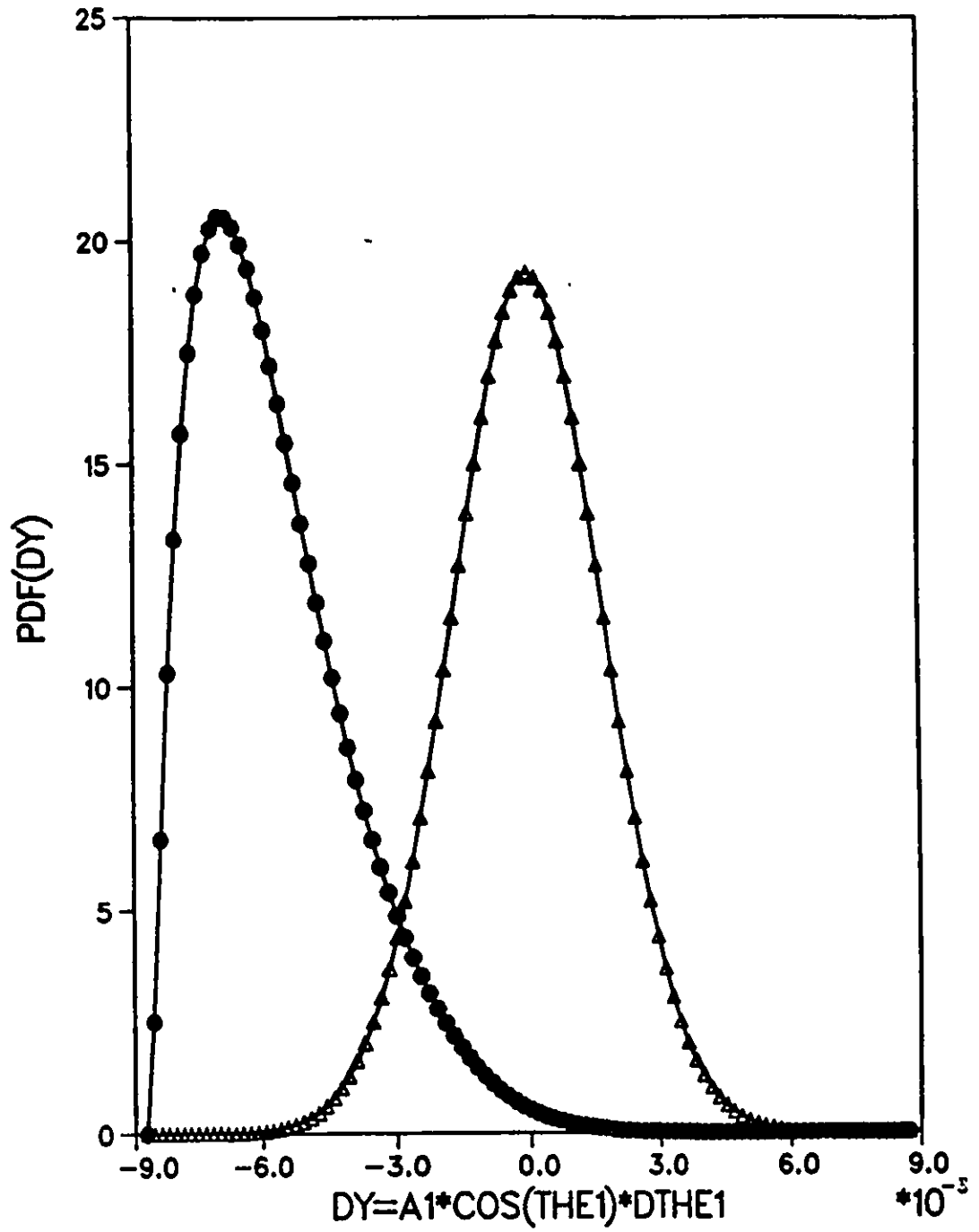


Figure 4.7: dY distribution for which $\theta_1 \in [0^\circ, 15^\circ]$.

DY1D2

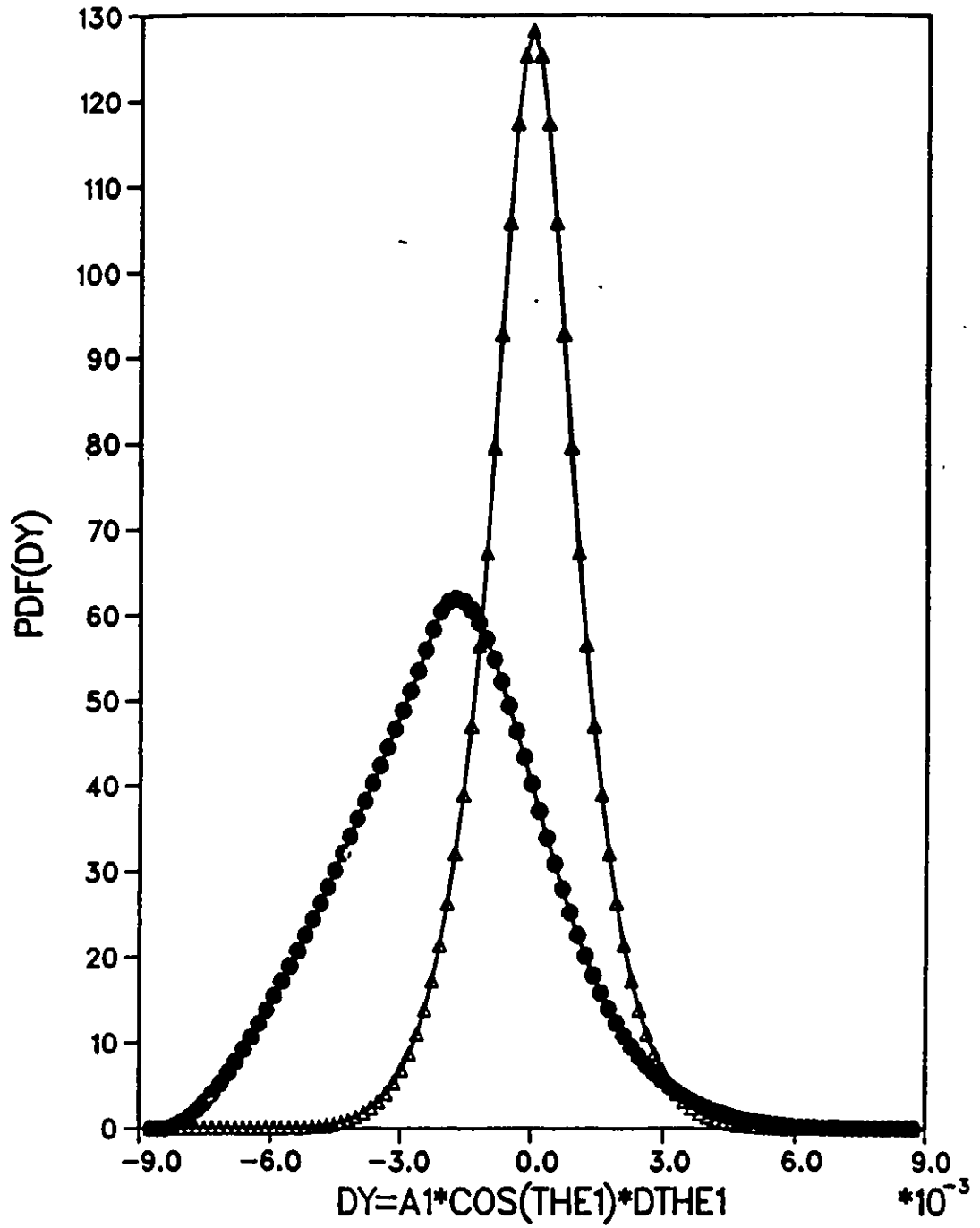


Figure 4.8: dY distribution for which $\theta_1 \in (15^\circ, 75^\circ)$.

DY1D3

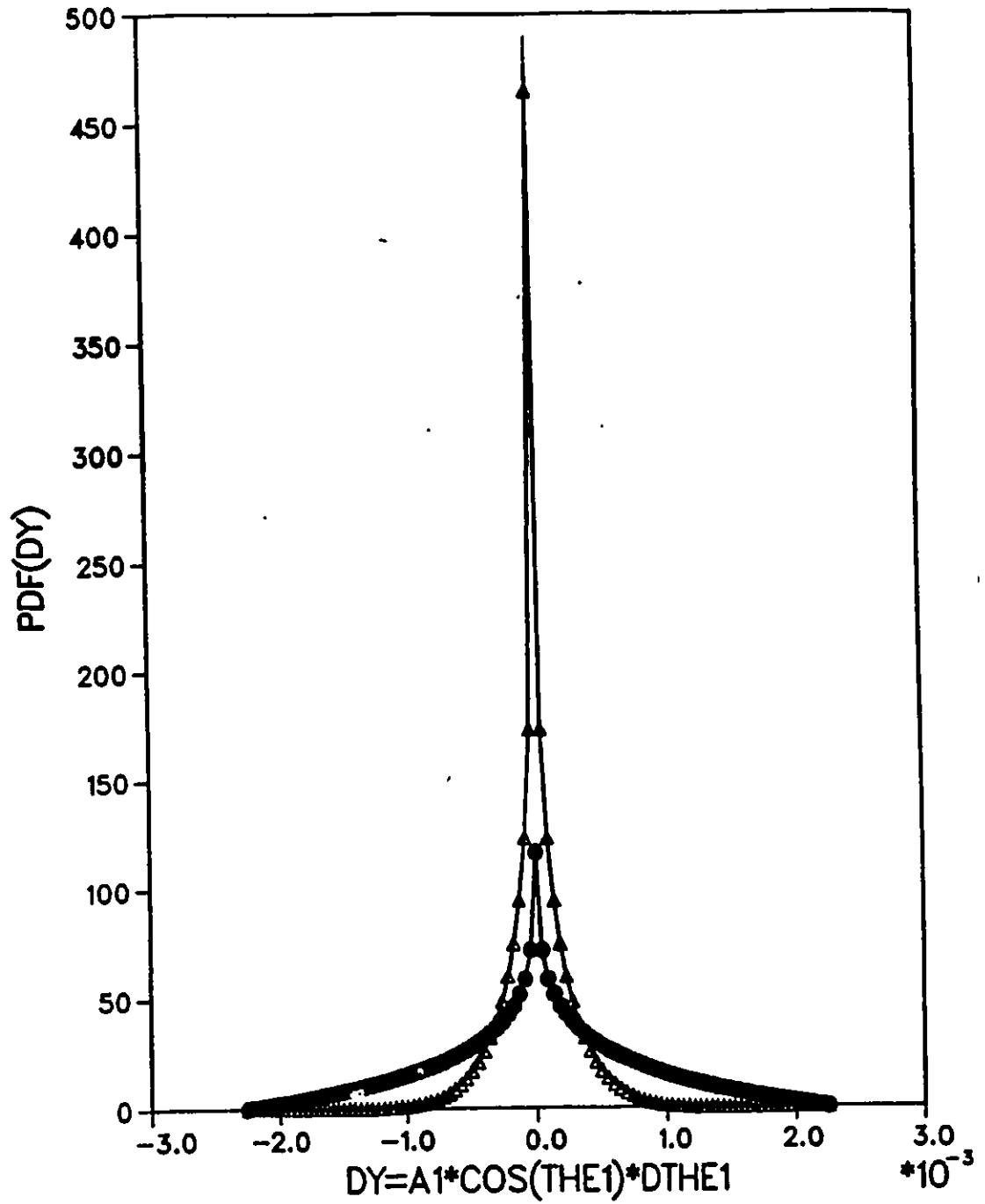


Figure 4.9: dY distribution for which $\theta_1 \in [75^\circ, 90^\circ]$.

DY1D

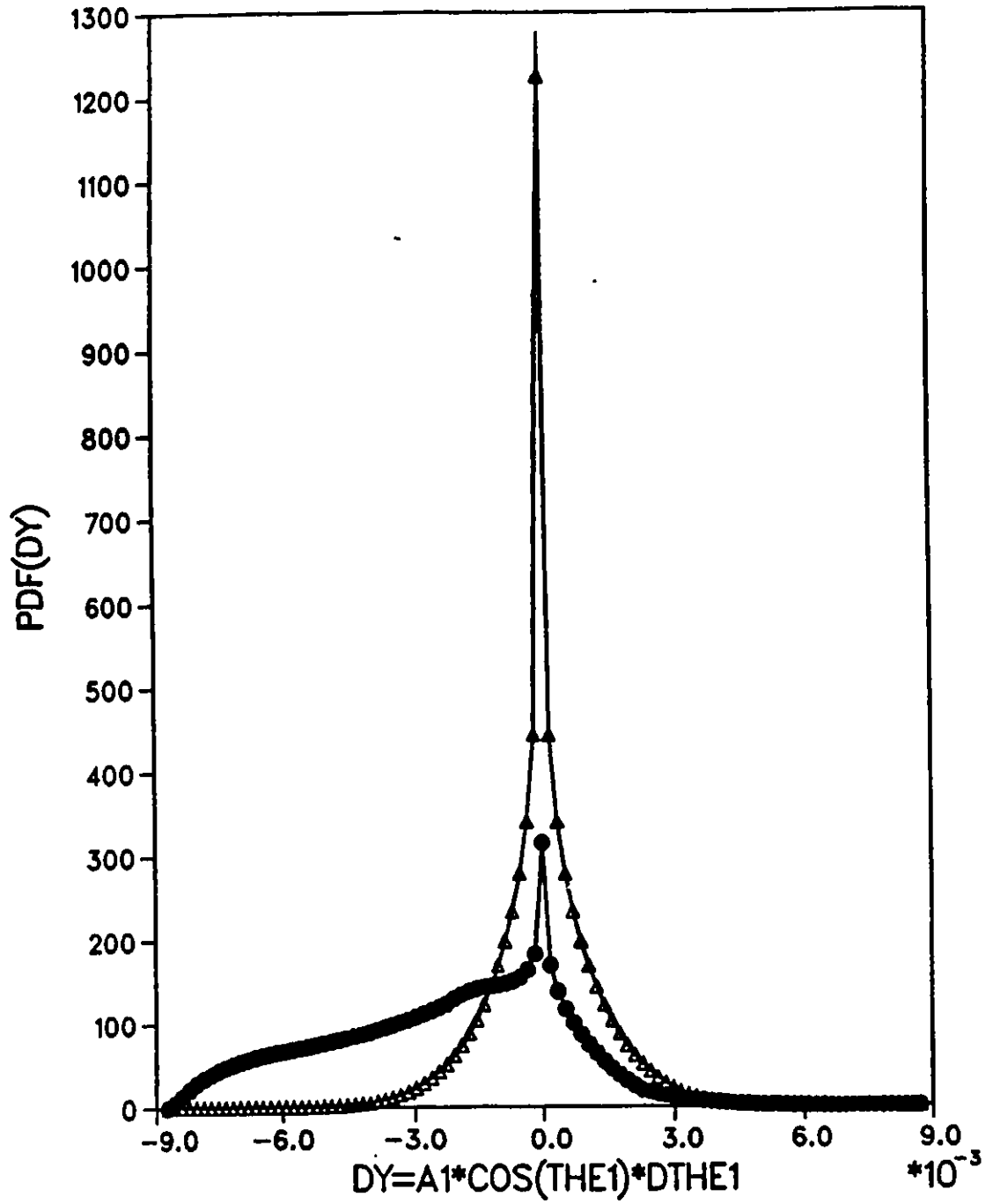


Figure 4.10: dY distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$ (the whole domain)

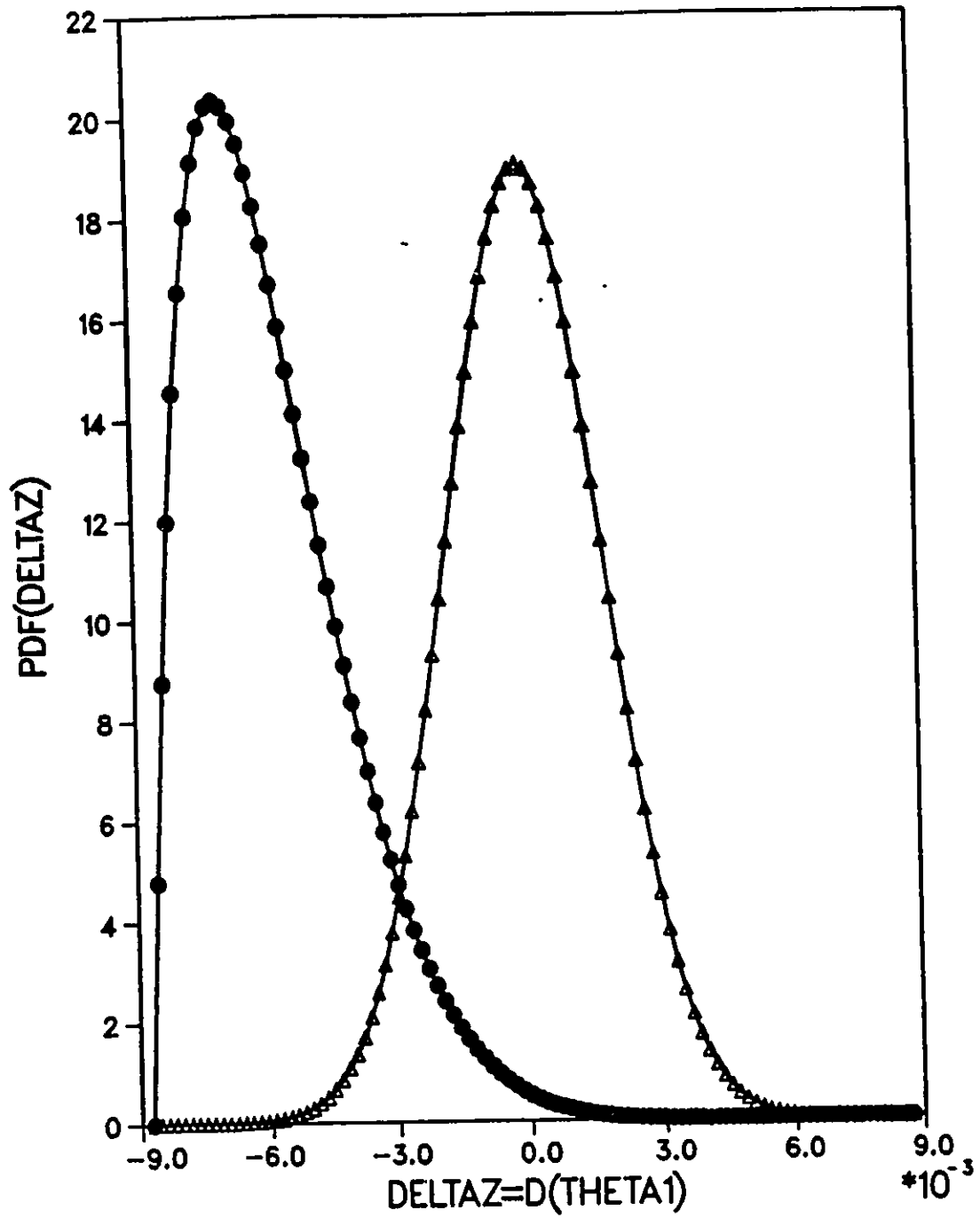


Figure 4.11: δ_z distribution for which $\theta_1 \in [0^\circ, 15^\circ]$.

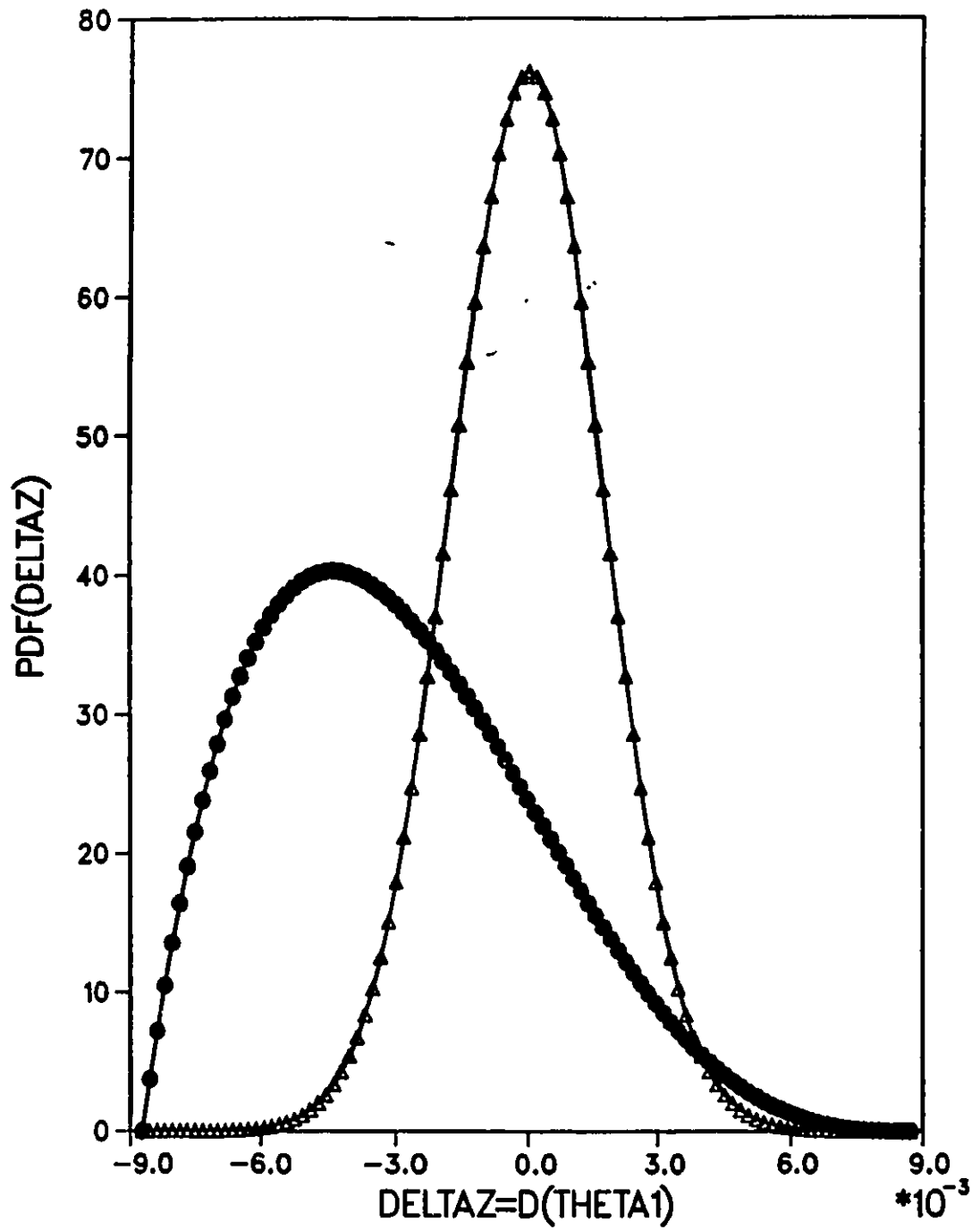


Figure 4.12: δ_z distribution for which $\theta_1 \in (15^\circ, 75^\circ)$.

TZ1D3

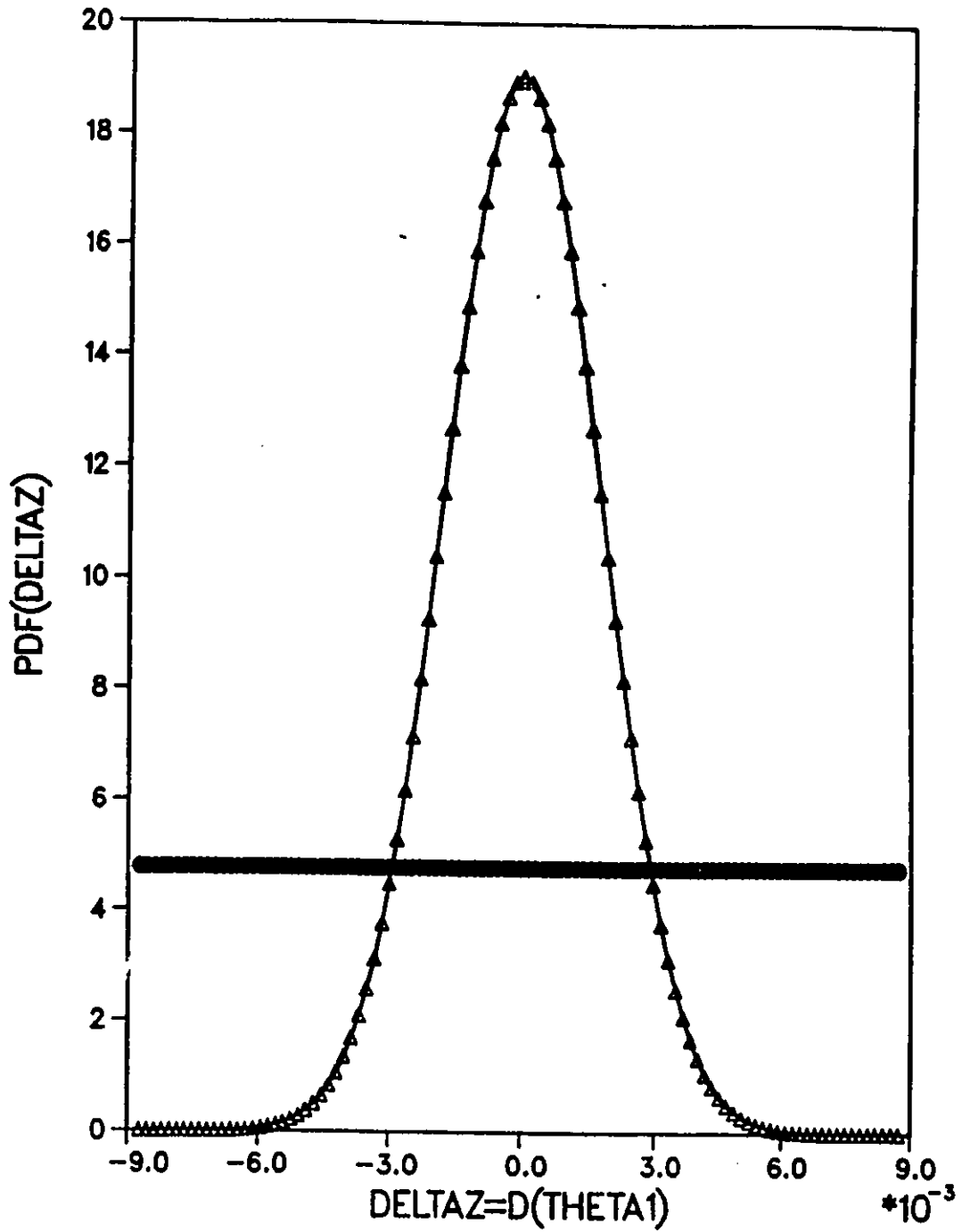


Figure 4.13: δ_z distribution for which $\theta_1 \in [75^\circ, 90^\circ]$.

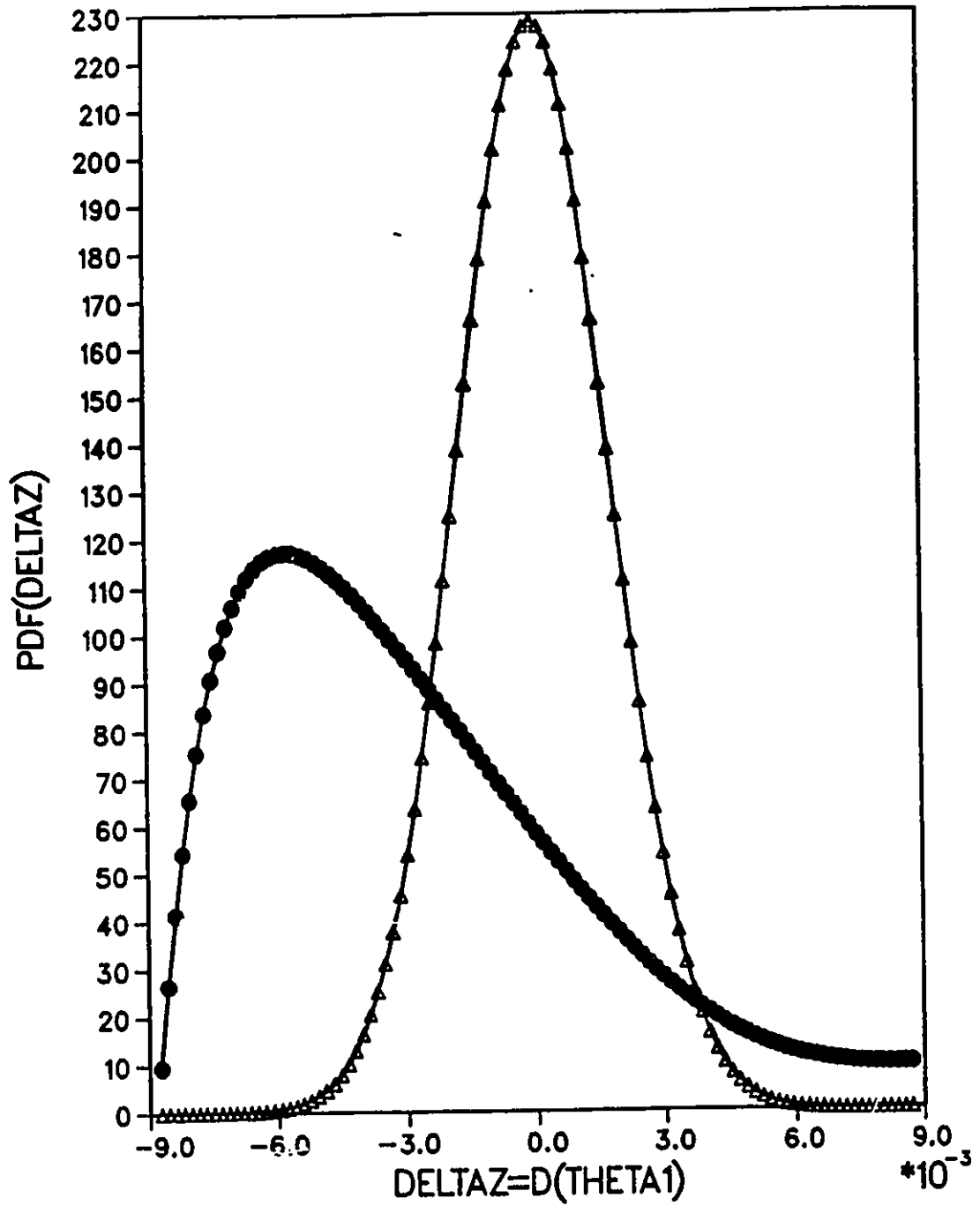


Figure 4.14: δ_z distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$ (the whole domain of θ_1)

Before we advance the proposed stochastic error propagation analysis and numerical simulations to higher degrees-of-freedom robot manipulator, some comments can be made :

The one degree-of-freedom robot manipulator used in these simulations are chosen because the analytical model, presented in chapter 3, is very complex. The one degree-of-freedom simulation permitted to clarify the features of the presented general stochastic model for the simplest case for which it can be applied. From the simulation results, Figure 3 to Figure 14, it is obvious that the presented model possesses the following features :

- the capability of representing non-symmetric error distributions in a rigorous pattern.
- the capability of comparing results from normal and non-normal probability density function.
- the richness in shapes of the beta distribution, chosen for these simulations in representing non-symmetric and symmetric (including uniform distribution) error distributions.

Chapter 5

Stochastic Error Propagation Analysis of a Two Degrees-of-Freedom Robot Manipulator

The error propagation model of chapter 2 and the general stochastic model of chapter 3 must be combined and applied to the modeling of more than one degree-of-freedom structures. A revolute two degrees-of-freedom planar robot manipulator operating only in a plane perpendicular to the gravity field is considered in this chapter.

In this case, gravity has effect on both joint variable errors $d\theta_1$ and $d\theta_2$. Thus, the joint variable errors become the dominant factor in determining the output positioning and orientational errors of the end-effector and are configuration dependent. A general stochastic analysis based on this fact is presented here. A comparison between the previous work based only on the normal distribution approach and this general stochastic error propagation approach is also presented, and the results show the importance of developing a general stochastic error propagation model to approach the arbitrarily mixed distributions existing in the link parameter errors of a robot manipulator.

5.1 Error Propagation Model of a Two Degrees-of-Freedom Robot Manipulator

The structure of the planar two degrees-of-freedom robot manipulator considered in this chapter is schematically shown in Figure 5.1, and the link coordinate frames are assigned according to the D-H convention. From Figure 5.1, we can establish the following link parameter table 5.1.

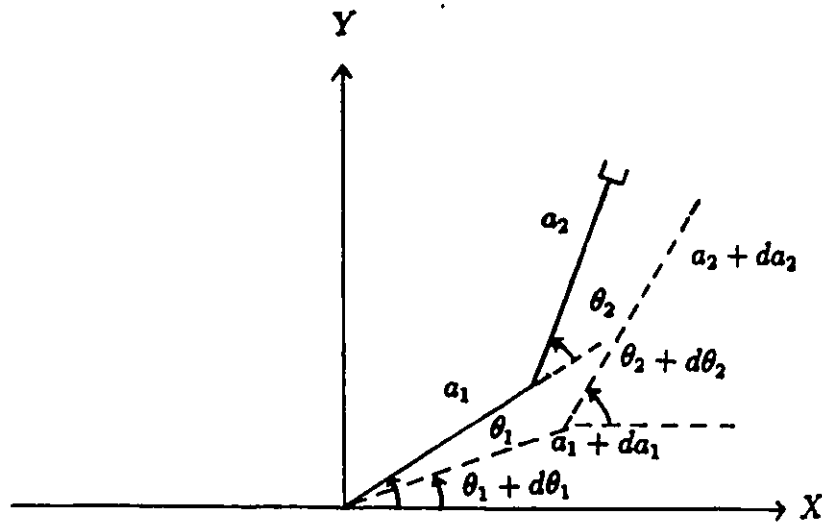


Figure 5.1: schematic drawing of a planar two degrees-of-freedom robot manipulator

Table 5.1: link parameter table

link n	variable	α_n	a_n	d_n	β_n
1	θ_1	0	a_1	0	0
2	θ_2	0	a_2	0	0

Using the general A transformation formula defined by (A.3), the A transformation matrix for this manipulator can be derived as

$$A_1 = \begin{bmatrix} C_1 & -S_1 & 0 & a_1 C_1 \\ S_1 & C_1 & 0 & a_1 S_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.1)$$

$$A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2 C_2 \\ S_2 & C_2 & 0 & a_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.2)$$

and the transformation matrix between the base frame and end-effector can be defined as

$$\begin{aligned} T_2 &= A_1 A_2 \\ &= \begin{bmatrix} C_{12} & -S_{12} & 0 & a_1 C_1 + a_2 C_{12} \\ S_{12} & C_{12} & 0 & a_1 S_1 + a_2 S_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (5.3)$$

From the coefficient matrix equations defined by equation (2.67) to (2.74), the coefficient matrices , R_θ, \dots, L_β defined by (2.60) and (2.61) can be calculated as

$$\begin{aligned} R_{\theta,1} &= R_0^0 * K1_n \\ &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned} \quad (5.4)$$

$$\begin{aligned} R_{\theta,2} &= R_1^0 * K1_2 \\ &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned} \quad (5.5)$$

$$\begin{aligned} R_{\alpha,1} &= R_0^0 * K2_1 \\ &= \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix} \end{aligned} \quad (5.6)$$

$$\begin{aligned}
R_{\alpha,2} &= R_1^0 * K2_2 \\
&= \begin{bmatrix} C_{12} \\ S_{12} \\ 0 \end{bmatrix}
\end{aligned} \tag{5.7}$$

$$\begin{aligned}
R_{\beta,1} &= R_0^0 * K5_1 \\
&= \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{5.8}$$

$$\begin{aligned}
R_{\beta,2} &= R_1^0 * K5_2 \\
&= \begin{bmatrix} -S_{12} \\ C_{12} \\ 0 \end{bmatrix}
\end{aligned} \tag{5.9}$$

$$\begin{aligned}
L_{\theta,1} &= (R_{\theta,1} \times P_2^0) + P_0^0 \times (R_0^0 * K1_1) \\
&= \begin{bmatrix} -a_1 S_1 - a_2 S_{12} \\ a_1 C_1 + a_2 C_{12} \\ 0 \end{bmatrix}
\end{aligned} \tag{5.10}$$

$$\begin{aligned}
L_{\theta,2} &= (R_{\theta,2} \times P_2^0) + P_1^0 \times (R_1^0 * K1_2) \\
&= \begin{bmatrix} -a_2 S_{12} \\ a_2 C_{12} \\ 0 \end{bmatrix}
\end{aligned} \tag{5.11}$$

$$L_{d,1} = R_{\theta,1} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{5.12}$$

$$L_{d,2} = R_{\theta,2} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{5.13}$$

$$L_{\alpha,1} = R_{\alpha,1} = \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix} \tag{5.14}$$

$$L_{\alpha,2} = R_{\alpha,2} = \begin{bmatrix} C_{12} \\ S_{12} \\ 0 \end{bmatrix} \quad (5.15)$$

$$\begin{aligned} L_{\alpha,1} &= (R_{\alpha,1} \times P_2^0) + P_0^0 \times (R_0^0 * K2_1) + (R_0^0 * K3_1) \\ &= \begin{bmatrix} 0 \\ 0 \\ a_2 S_2 \end{bmatrix} \end{aligned} \quad (5.16)$$

$$\begin{aligned} L_{\alpha,2} &= (R_{\alpha,2} \times P_2^0) + P_1^0 \times (R_1^0 * K2_2) + (R_1^0 * K3_2) \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \end{aligned} \quad (5.17)$$

$$\begin{aligned} L_{\beta,1} &= (R_{\beta,1} \times P_2^0) + (R_0^0 * k4_1) \\ &= \begin{bmatrix} 0 \\ 0 \\ -a_2 C_2 \end{bmatrix} \end{aligned} \quad (5.18)$$

$$\begin{aligned} L_{\beta,2} &= (R_{\beta,2} \times P_2^0) + P_1^0 \times (R_1^0 * K5_2) + (R_1^0 * K4_2) \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \end{aligned} \quad (5.19)$$

The orientational error for the end-effector of the manipulator shown in Figure 5.1 can be calculated by substituting (5.4) to (5.9) into (2.60) as

$$\begin{aligned} \delta &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \Delta\theta_1 \\ \Delta\theta_2 \end{bmatrix} + \begin{bmatrix} C_1 & C_{12} \\ S_1 & S_{12} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\alpha_1 \\ \Delta\alpha_2 \end{bmatrix} + \begin{bmatrix} -S_1 & -S_{12} \\ C_1 & C_{12} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\beta_1 \\ \Delta\beta_2 \end{bmatrix} \\ &= \begin{bmatrix} C_1\Delta\alpha_1 + C_{12}\Delta\alpha_2 - S_1\Delta\beta_1 - S_{12}\Delta\beta_2 \\ S_1\Delta\alpha_1 + S_{12}\Delta\alpha_2 + C_1\Delta\beta_1 + C_{12}\Delta\beta_2 \\ \Delta\theta_1 + \Delta\theta_2 \end{bmatrix} \end{aligned} \quad (5.20)$$

and the positioning error of the end-effector can be calculated by substituting (5.10) to (5.19) into (2.61) as

$$dP = \begin{bmatrix} -a_1 S_1 - a_2 S_{12} & -a_2 S_{12} \\ a_1 C_1 + a_2 C_{12} & a_2 C_{12} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\theta_1 \\ \Delta\theta_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \Delta d_1 \\ \Delta d_2 \end{bmatrix}$$

$$\begin{aligned}
& + \begin{bmatrix} C_1 & C_{12} \\ S_1 & S_{12} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta a_1 \\ \Delta a_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ a_2 S_2 & 0 \end{bmatrix} \begin{bmatrix} \Delta \alpha_1 \\ \Delta \alpha_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -a_2 C_2 & 0 \end{bmatrix} \begin{bmatrix} \Delta \beta_1 \\ \Delta \beta_2 \end{bmatrix} \\
& = \begin{bmatrix} (-a_1 S_1 - a_2 S_{12})\Delta\theta_1 + (-a_2 S_{12})\Delta\theta_2 + C_1\Delta a_1 + C_{12}\Delta a_2 \\ (a_1 C_1 + a_2 C_{12})\Delta\theta_1 + (a_2 C_{12})\Delta\theta_2 + S_1\Delta a_1 + S_{12}\Delta a_2 \\ \Delta d_1 + \Delta d_2 + (a_2 S_2)\Delta\alpha_1 + (-a_2 C_2)\Delta\beta_1 \end{bmatrix} \quad (5.21)
\end{aligned}$$

Equations (5.20) and (5.21) define the linear error propagation model for the revolute planar two degrees-of-freedom robot manipulator shown in Figure 5.1. These equations will be used in the stochastic error propagation analysis of the next section.

5.2 Stochastic Error Propagation Model

For the manipulator shown in Figure 5.1, the error propagation equations for the positioning and orientational errors of the end-effector are defined by equations (5.20) and (5.21). In order to find the distribution functions of the Cartesian positioning and orientational errors, the method to propagate all errors described in chapter 3 will be employed. First, we consider the X component of the Cartesian positioning error. We start with choosing the output random variables as

$$Y_1 = dX = (-a_1 S_1 - a_2 S_{12})\Delta\theta_1 - (a_2 S_{12})\Delta\theta_2 + (C_1)\Delta a_1 + (C_{12})\Delta a_2 \quad (5.22)$$

$$Y_2 = \theta_1 \quad (5.23)$$

$$Y_3 = \theta_2 \quad (5.24)$$

$$Y_4 = \Delta\theta_1 \quad (5.25)$$

$$Y_5 = \Delta\theta_2 \quad (5.26)$$

$$Y_6 = \Delta a_1 \quad (5.27)$$

for which, the solutions for the input random parameters can be written as

$$X_1 = \theta_1 = Y_2 \quad (5.28)$$

$$X_2 = \theta_2 = Y_3 \quad (5.29)$$

$$X_3 = \Delta\theta_1 = Y_4 \quad (5.30)$$

$$X_4 = \Delta\theta_2 = Y_5 \quad (5.31)$$

$$X_5 = \Delta a_1 = Y_6 \quad (5.32)$$

$$\begin{aligned} X_6 &= \Delta a_2 \\ &= \frac{Y_1 - [-a_1 S(Y_2) - a_2 S(Y_2 + Y_3)]Y_4 - [-a_2 S(Y_2 + Y_3)]Y_5 - C(Y_2)Y_6}{C(Y_2 + Y_3)} \end{aligned} \quad (5.33)$$

The joint probability density function of the output random variables Y_1, \dots, Y_6 can be found from the joint probability density function of the input random variables X_1, \dots, X_6 . Therefore, the distribution function of the positioning error dX can be found as the marginal distribution of Y_1, \dots, Y_6 by integrating over the whole domains of the supplementary variables Y_2, \dots, Y_6 . Because the transformation from (Y_1, \dots, Y_6) to (X_1, \dots, X_6) is a one-to-one transformation, in this case the determinant of the Jacobian defined by (3.3) is calculated as

$$\begin{aligned} J(X_1, \dots, X_6) &= \left| \frac{\partial(g_1, \dots, g_6)}{\partial(X_1, \dots, X_6)} \right| \\ &= \begin{vmatrix} J_{11} & J_{12} & J_{13} & J_{14} & J_{15} & C_{12} \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} \\ &= C_{12} \end{aligned} \quad (5.34)$$

where the coefficients J_{11} to J_{15} do not affect the results and are not calculated here and the Jacobian must satisfy the condition $J = C_{12} \neq 0$.

Then, the joint probability function of the output random parameters Y_1, \dots, Y_6 can be found from (3.5) as

$$\begin{aligned} &f_{Y_1, \dots, Y_6}(y_1, \dots, y_6) \\ &= f_{X_1, \dots, X_6}(y_2, \dots, y_5, \frac{y_1 - [-a_1 S y_2 - a_2 S y_{23}] y_4 - [-a_2 S y_{23}] y_5 - [C y_2] y_6}{C y_{23}}) \quad (5.35) \\ &/ | C y_{23} | \quad ; C y_{23} \neq 0 \end{aligned}$$

Using (3.34), the distribution function of the Cartesian positioning error dX can be found by performing the integration of the above equation over the domains of

all the supplementary random parameters Y_2, \dots, Y_6 as

$$\begin{aligned}
f_{Y_1}(y_1) &= f_{dX}(y_1) \\
&= \int_{\theta_1} \int_{\theta_2} \int_{\Delta\theta_1} \int_{\Delta\theta_2} \int_{\Delta\alpha_1} f_{\theta_1, \theta_2, \Delta\theta_1, \Delta\theta_2, \Delta\alpha_1, \Delta\alpha_2}(\theta_1, \theta_2, \Delta\theta_1, \Delta\theta_2, \\
&\quad \Delta\alpha_1, \frac{y_1 - (-a_1 S_1 - a_2 S_{12})\Delta\theta_1 - (-a_2 S_{12})\Delta\theta_2 - (C_1)\Delta\alpha_1}{C_{12}}) \\
&\quad / |C_{12}| \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\Delta\theta_1) \cdot d(\Delta\theta_2) \cdot d(\Delta\alpha_1); C_{12} \neq 0 \quad (5.36)
\end{aligned}$$

In this case, with regard to the base frame, the position of link 1 is determined by joint variable θ_1 and that of link 2 is determined by $\theta_1 + \theta_2$. We consider that the gravity effect is configuration dependent such that each point of the work-volume should take a different degree-of-bias distribution. We simplify the analysis by assuming if the domains of θ_1 and $\theta_1 + \theta_2$ be divided into L_1 and L_2 independent smaller sub-domains respectively such that in each of these sub-domains the gravity effect is approximately the same. Then in each of these sub-domains all the link parameter errors can be considered independent. Therefore, the above error distribution function for dX can be further expanded as

$$\begin{aligned}
f_{dX}(dx) &= \sum_{i_1=1}^{L_1} \sum_{i_2=1}^{L_2} \int_{\theta_{1,i_1}} \int_{\theta_{2,i_2}} \int_{\Delta\theta_1} \int_{\Delta\theta_2} \int_{\Delta\alpha_1} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\Delta\theta_1}(\Delta\theta_1) f_{\Delta\theta_2}(\Delta\theta_2) \\
&\quad f_{\Delta\alpha_1}(\Delta\alpha_1) f_{\Delta\alpha_2}(\frac{dx - (-a_1 S_1 - a_2 S_{12})\Delta\theta_1 - (-a_2 S_{12})\Delta\theta_2 - (C_1)\Delta\alpha_1}{C_{12}}) \\
&\quad / |C_{12}| \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\Delta\theta_1) \cdot d(\Delta\theta_2) \cdot d(\Delta\alpha_1) \quad ; C_{12} \neq 0 \quad (5.37)
\end{aligned}$$

By using the same procedures as above, we can derive the distributions of the errors dY , dZ , δ_X , δ_Y and δ_Z as

$$\begin{aligned}
f_{dY}(dy) &= \sum_{i_1=1}^{L_1} \sum_{i_2=1}^{L_2} \int_{\theta_{1,i_1}} \int_{\theta_{2,i_2}} \int_{\Delta\theta_1} \int_{\Delta\theta_2} \int_{\Delta\alpha_1} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\Delta\theta_1}(\Delta\theta_1) f_{\Delta\theta_2}(\Delta\theta_2) f_{\Delta\alpha_1}(\Delta\alpha_1) \\
&\quad f_{\Delta\alpha_2}(\frac{dy - (a_1 C_1 + a_2 C_{12})\Delta\theta_1 - (a_2 C_{12})\Delta\theta_2 - S_1 \Delta\alpha_1}{S_{12}}) \\
&\quad / |S_{12}| \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\Delta\theta_1) \cdot d(\Delta\theta_2) \cdot d(\Delta\alpha_1); S_{12} \neq 0 \quad (5.38)
\end{aligned}$$

$$\begin{aligned}
f_{dZ}(dz) &= \int_{\theta_2} \int_{\Delta d_1} \int_{\Delta d_2} \int_{\Delta\alpha_1} f_{\theta_2}(\theta_2) f_{\Delta d_1}(\Delta d_1) f_{\Delta d_2}(\Delta d_2) \\
&\quad f_{\Delta\alpha_1}(\Delta\alpha_1) f_{\Delta\beta_1}(\frac{dz - \Delta d_1 - \Delta d_2 - (a_2 S_2)\Delta\alpha_1}{-a_2 C_2}) \\
&\quad / |a_2 C_2| \cdot d(\theta_2) \cdot d(\Delta d_1) \cdot d(\Delta d_2) \cdot d(\Delta\alpha_1); C_2 \neq 0 \quad (5.39)
\end{aligned}$$

$$\begin{aligned}
f_{\delta_x}(\delta_x) &= \int_{\theta_1} \int_{\theta_2} \int_{\Delta\alpha_1} \int_{\Delta\alpha_2} \int_{\Delta\beta_1} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\Delta\alpha_1}(\Delta\alpha_1) \\
&\quad f_{\Delta\alpha_2}(\Delta\alpha_2) f_{\Delta\beta_1}(\Delta\beta_1) f_{\Delta\beta_2} \left(\frac{\delta_x - C_1 \Delta\alpha_1 - C_{12} \Delta\alpha_2 + S_1 \Delta\beta_1}{-S_{12}} \right) / |S_{12}| \\
&\quad \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\Delta\alpha_1) \cdot d(\Delta\alpha_2) \cdot d(\Delta\beta_1); S_{12} \neq 0 \tag{5.40}
\end{aligned}$$

$$\begin{aligned}
f_{\delta_y}(\delta_y) &= \int_{\theta_1} \int_{\theta_2} \int_{\Delta\alpha_1} \int_{\Delta\alpha_2} \int_{\Delta\beta_1} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\Delta\alpha_1}(\Delta\alpha_1) f_{\Delta\alpha_2}(\Delta\alpha_2) \\
&\quad f_{\Delta\beta_1}(\Delta\beta_1) f_{\Delta\beta_2} \left(\frac{\delta_y - S_1 \Delta\alpha_1 - S_{12} \Delta\alpha_2 - C_1 \Delta\beta_1}{C_{12}} \right) / |C_{12}| \\
&\quad \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\Delta\alpha_1) \cdot d(\Delta\alpha_2) \cdot d(\Delta\beta_1); C_{12} \neq 0 \tag{5.41}
\end{aligned}$$

$$\begin{aligned}
f_{\delta_z}(\delta_z) &= \sum_{i_1=1}^{L_1} \sum_{i_2=1}^{L_2} \int_{\theta_{1,i_1}} \int_{\theta_{2,i_2}} \int_{\Delta\theta_1} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) \\
&\quad f_{\Delta\theta_1}(\Delta\theta_1) f_{\Delta\theta_2}(\delta_z - \Delta\theta_1) \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\Delta\theta_1) \tag{5.42}
\end{aligned}$$

5.3 Numerical Stochastic Simulations

The Cartesian positioning and orientational errors of the end-effector are defined by equations (5.37) to (5.42). Since the gravity effect is configuration dependent, to illustrate the results without experimental burden, a numerical simulation based on the following assumptions is presented.

1. Both joint 1 and 2 are limited to $\pm 90^\circ$.
2. When θ_1 is located within $[-15^\circ, 15^\circ]$, $d\theta_1$ is assumed to take a beta distribution with beta parameter $r_1 = 2$ and $s_1 = 10$; when $\theta_1 + \theta_2$ is located within $[-15^\circ, 15^\circ]$, $d\theta_2$ is also assumed to take a beta distribution with beta parameter $r'_1 = 2$ and $s'_1 = 10$.
3. When θ_1 is located within $[75^\circ, 90^\circ]$ or $[-90^\circ, -75^\circ]$, $d\theta_1$ takes a uniform distribution; when $\theta_1 + \theta_2$ is located within $[75^\circ, 105^\circ]$ or $[-105^\circ, -75^\circ]$, $d\theta_2$ takes a uniform distribution, too.
4. When θ_1 is located within $(15^\circ, 75^\circ)$ or $(-75^\circ, -15^\circ)$, $d\theta_1$ takes a beta distribution with beta parameter $r_2 = 2$ and $s_2 = 4$; when $\theta_1 + \theta_2$ is located within $(15^\circ, 75^\circ)$ or $(-75^\circ, -15^\circ)$, $d\theta_2$ takes the same beta distribution with parameter $r'_2 = 2$ and $s'_2 = 4$.

5. Due to the opposite effect of gravity, when $\theta_1 + \theta_2$ is located within $[90^\circ, 180^\circ]$ or $[-180^\circ, -90^\circ]$, $d\theta_2$ takes a beta distribution which is the mirror image about Y axis of the distributions of $d\theta_2$ in the domain $[-90^\circ, 90^\circ]$.
6. $\sigma_{\Delta\theta_1} = \sigma_{\Delta\theta_2} = 0.1^\circ$; $\Delta\theta_1$ and $\Delta\theta_2$ are all located within the 5σ band.
7. All the other link parameter errors are assumed their nominal values with zero variance.
 - $\sigma_{\Delta\alpha_1} = \sigma_{\Delta\alpha_2} = \sigma_{\Delta\beta_1} = \sigma_{\Delta\beta_2} = 0^\circ$.
 - $\sigma_{\Delta a_1} = \sigma_{\Delta a_2} = \sigma_{\Delta d_1} = \sigma_{\Delta d_2} = 0\text{m}$.

The joint (controlled) variable error distributions based on the above assumptions are shown in Figure 5.2 and 5.3. The simulation results are presented in Figure 5.4 to 5.15. All the simulations compare the difference between this presented general stochastic error propagation approach (curves with black dot marker) and the approach based on the assumption that all the link parameter errors take normal distributions with zero mean (curves with triangular marker). From the presented results, it is obvious that it is important to develop this general stochastic error propagation model to deal with the arbitrarily mixed distributions existing in the link parameter errors of a robot manipulator.

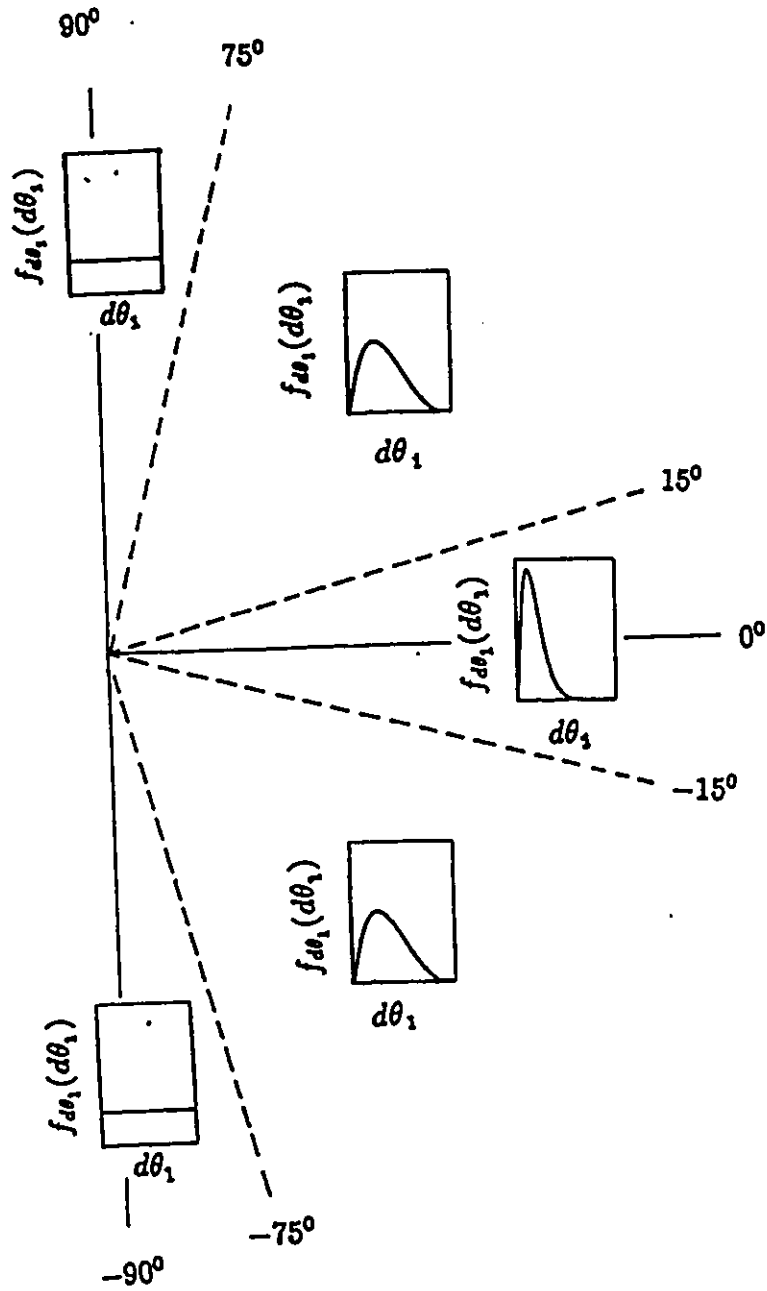


Figure 5.2: $d\theta_1$ distributions in different sub-domains of θ_1 ; $\theta_1 \in [-90^\circ, 90^\circ]$

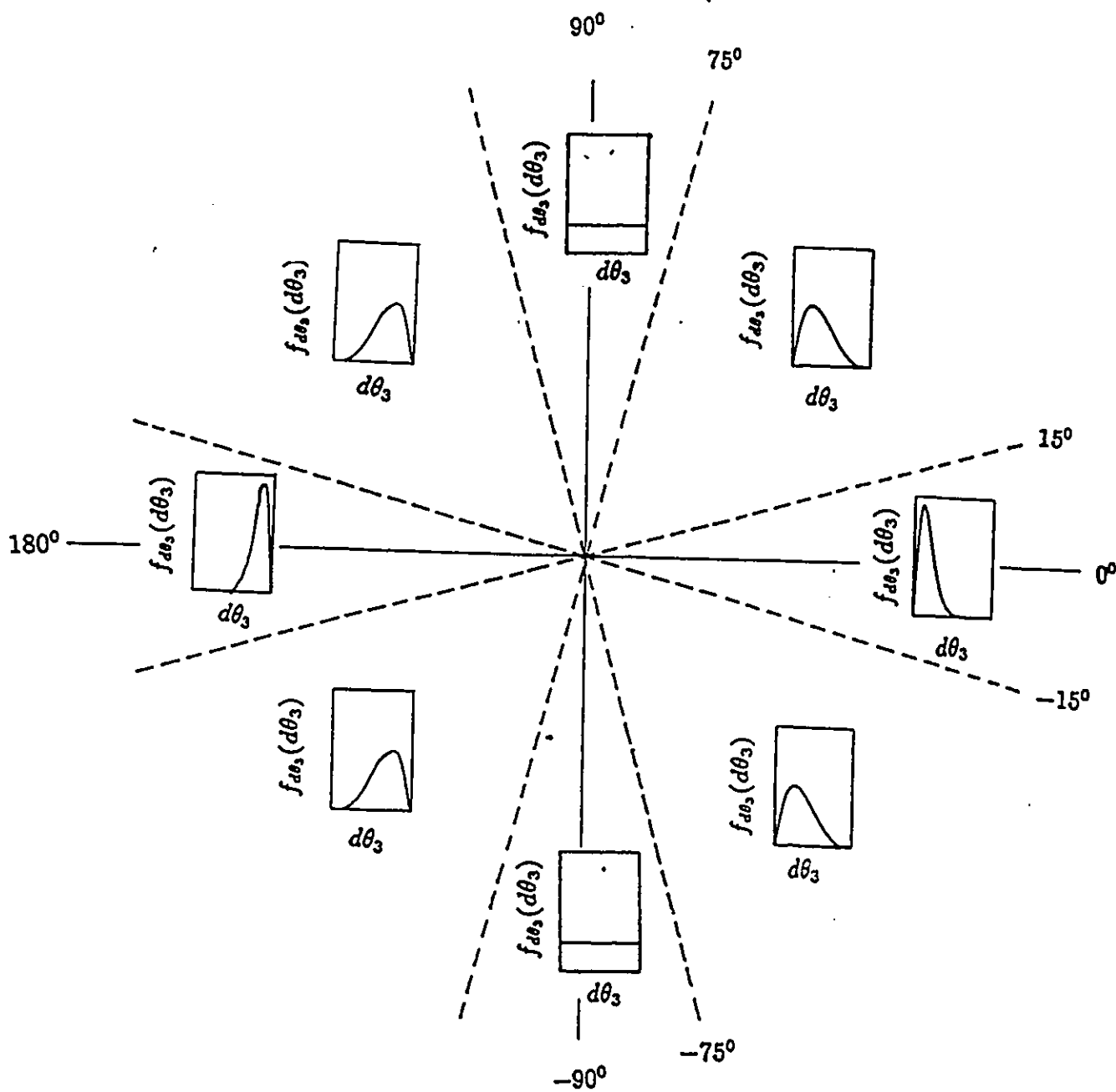


Figure 5.3: $d\theta_2$ distributions in different sub-domains of $(\theta_1 + \theta_2)$; $(\theta_1 + \theta_2) \in [-180^\circ, 180^\circ]$

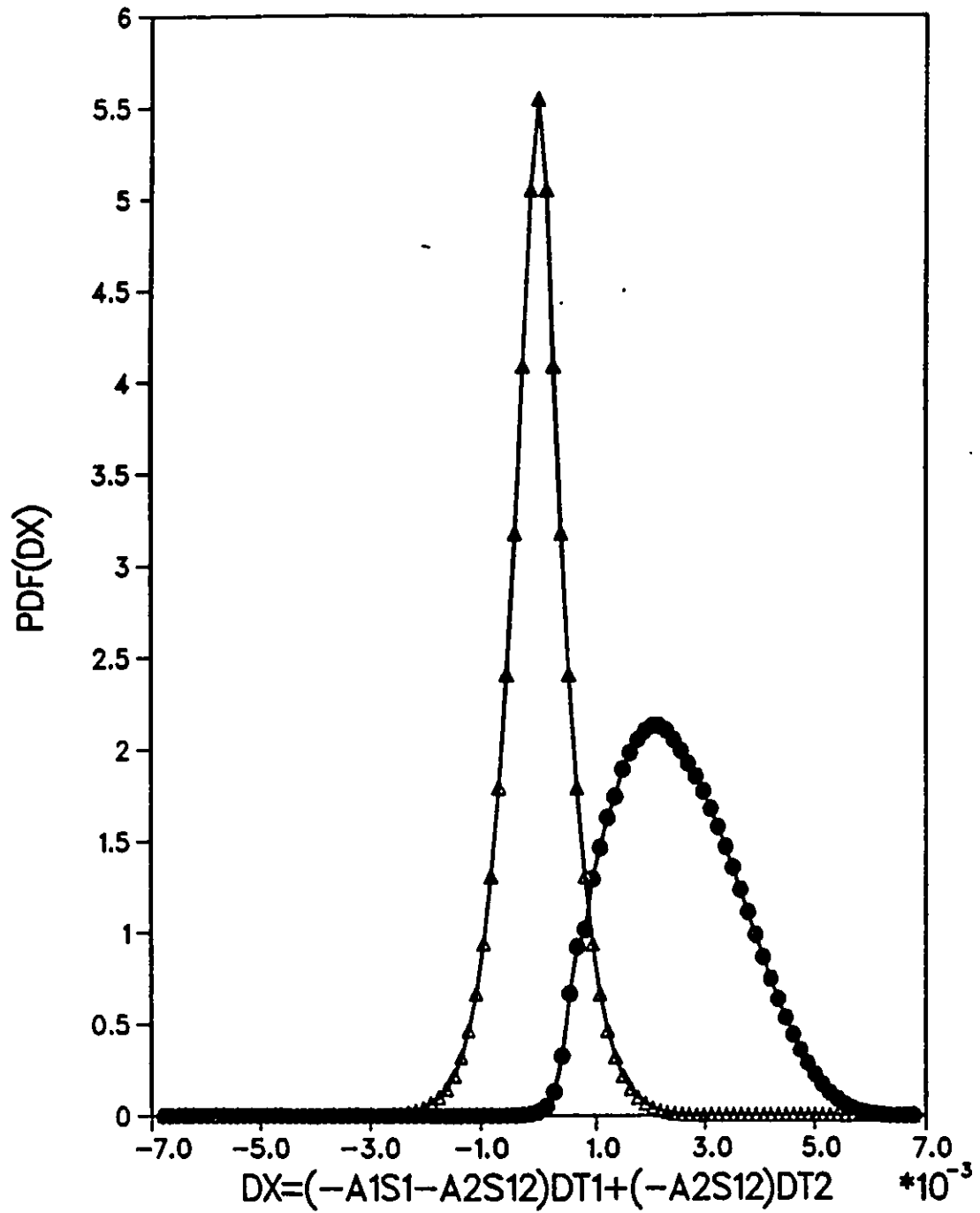


Figure 5.4: dX distribution for which $\theta_1 \in [0^\circ, 15^\circ]$ and $(\theta_1 + \theta_2) \in [0^\circ, 15^\circ]$.

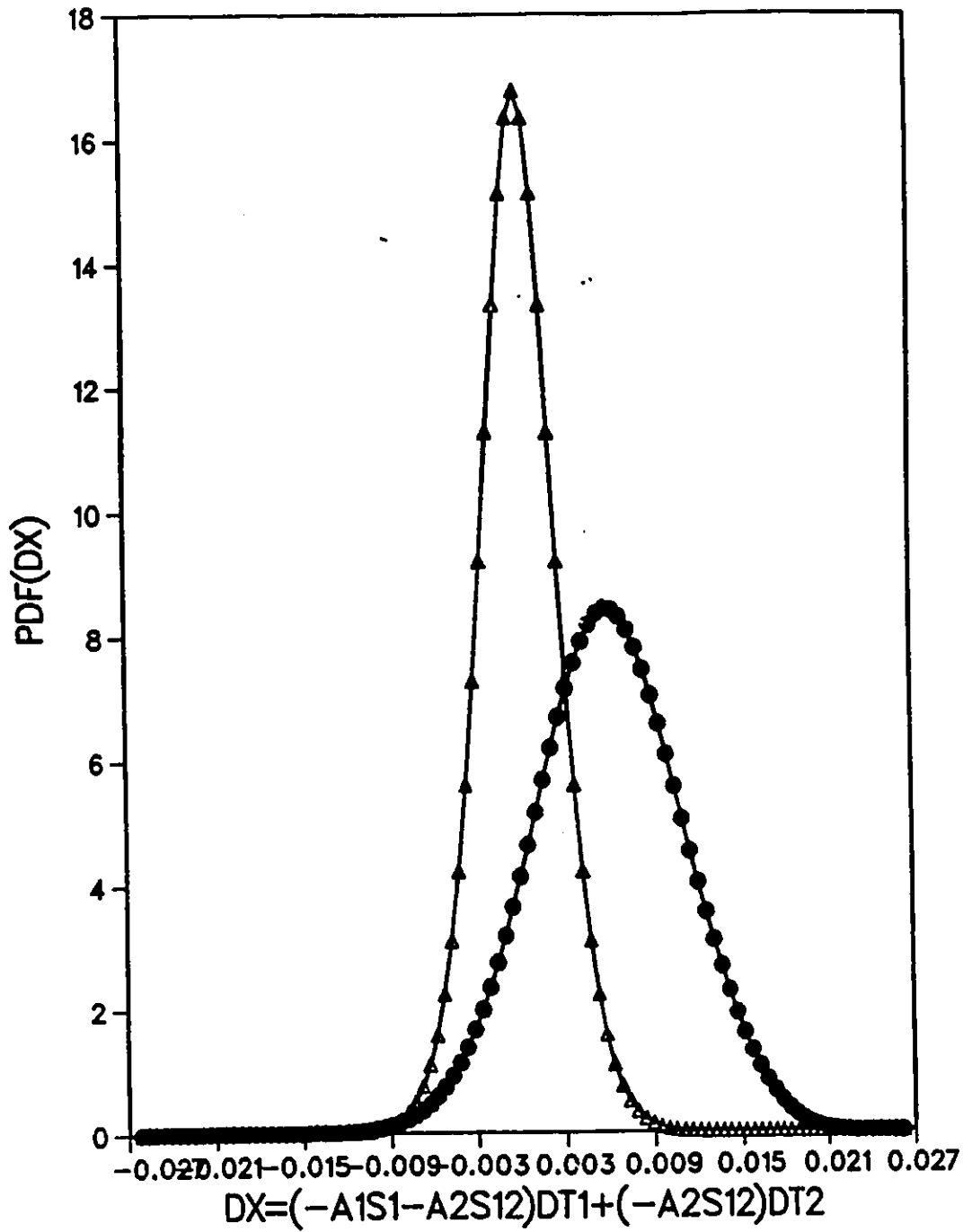


Figure 5.5: dX distribution for which $\theta_1 \in (15^\circ, 75^\circ)$ and $(\theta_1 + \theta_2) \in (15^\circ, 75^\circ)$.

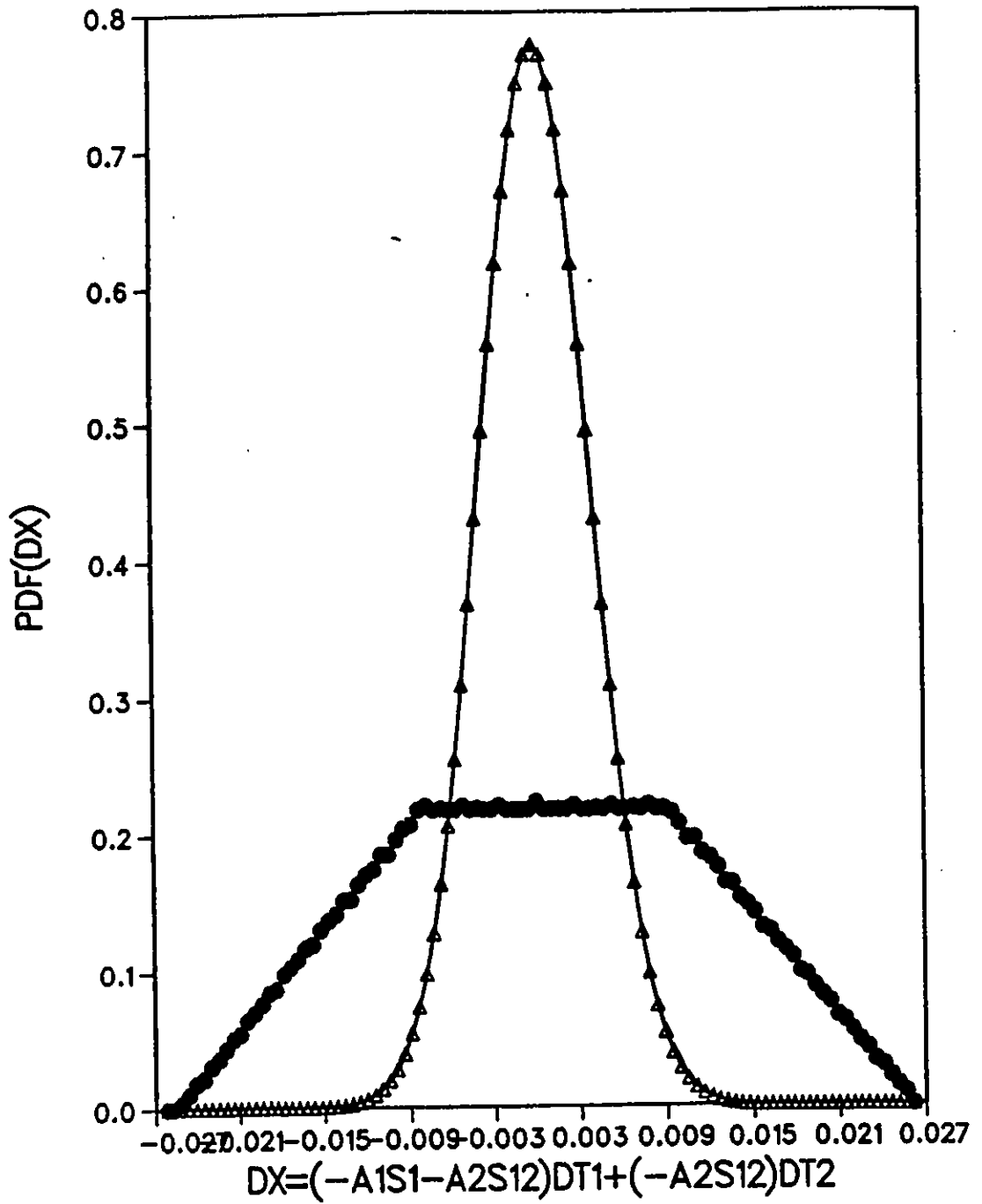


Figure 5.6: dX distribution for which $\theta_1 \in [75^\circ, 90^\circ]$ and $(\theta_1 + \theta_2) \in [75^\circ, 90^\circ]$.

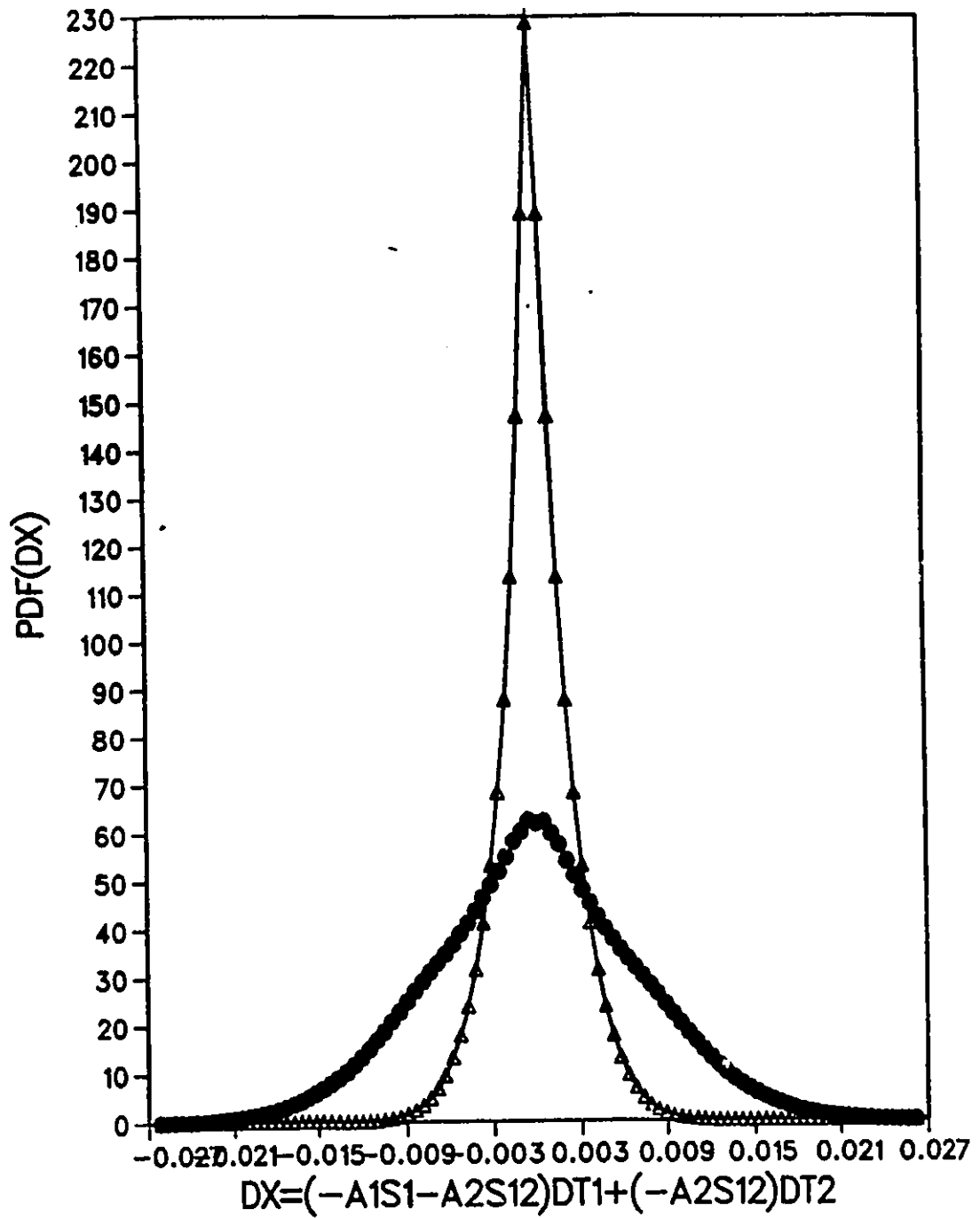


Figure 5.7: dX distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_1) and $(\theta_1 + \theta_2) \in [-180^\circ, 180^\circ]$, (the whole domain of $(\theta_1 + \theta_2)$.)

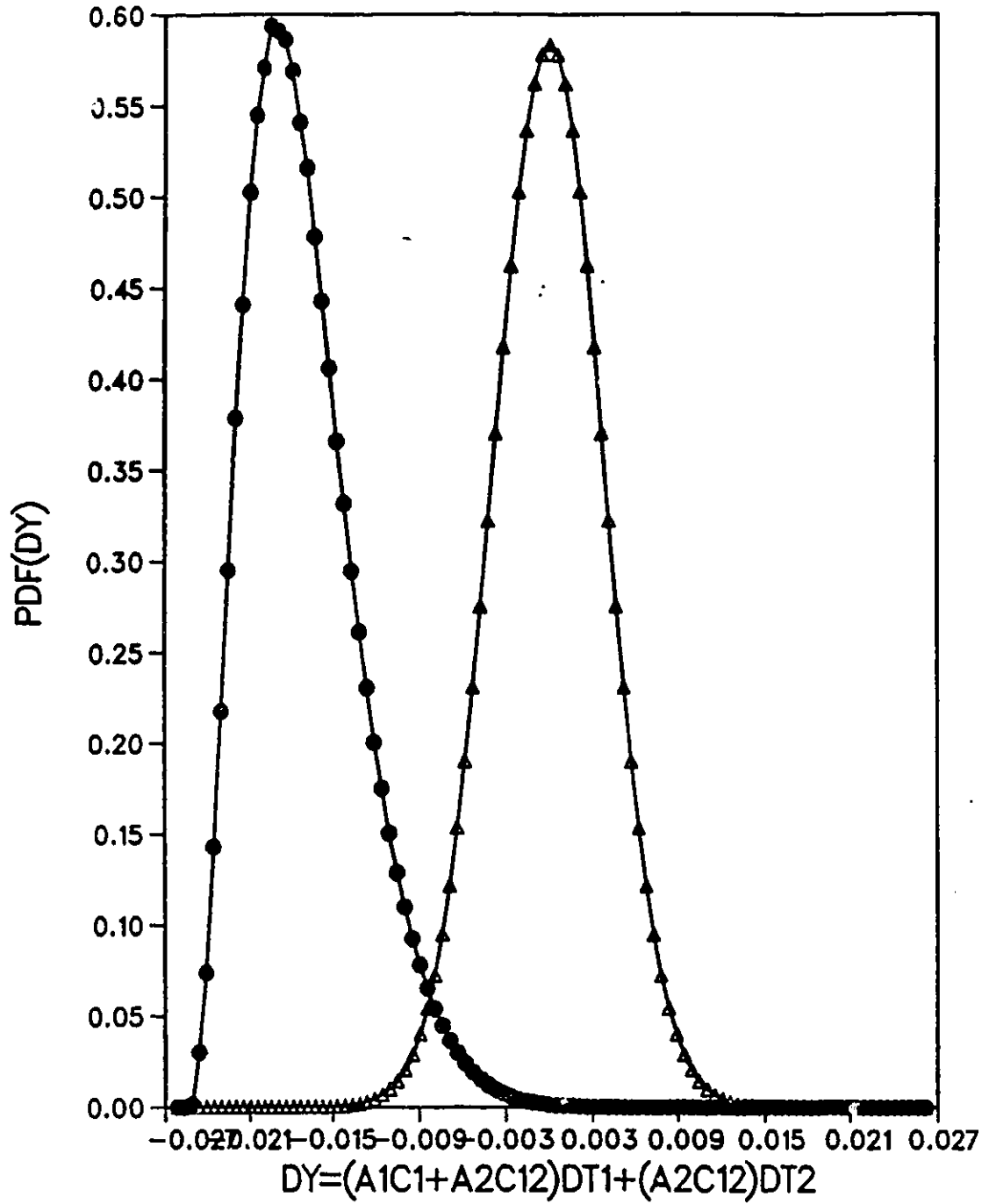


Figure 5.8: dY distribution for which $\theta_1 \in [0^\circ, 15^\circ]$ and $(\theta_1 + \theta_2) \in [0^\circ, 15^\circ]$.

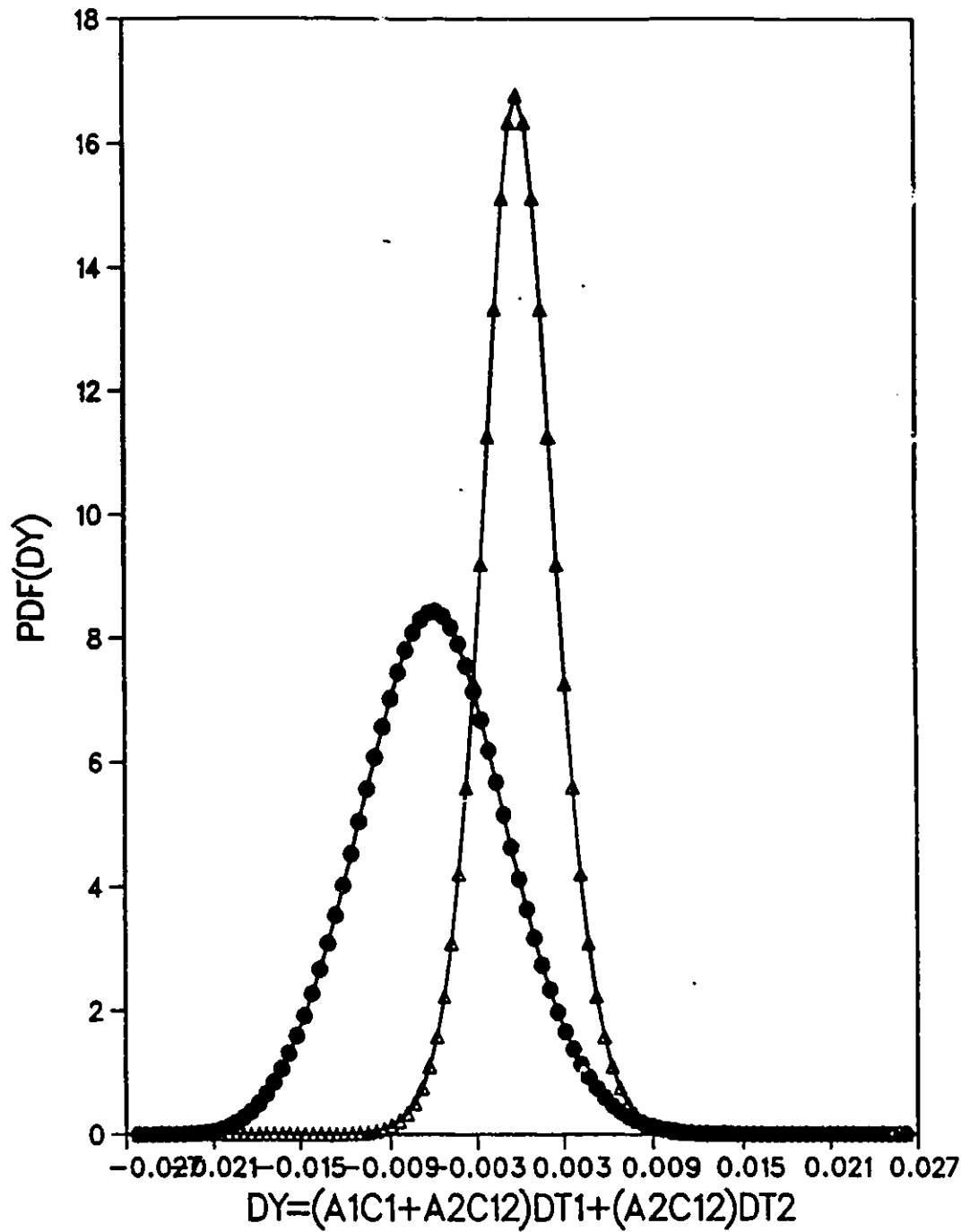


Figure 5.9: dY distribution for which $\theta_1 \in (15^\circ, 75^\circ)$ and $(\theta_1 + \theta_2) \in (15^\circ, 75^\circ)$.

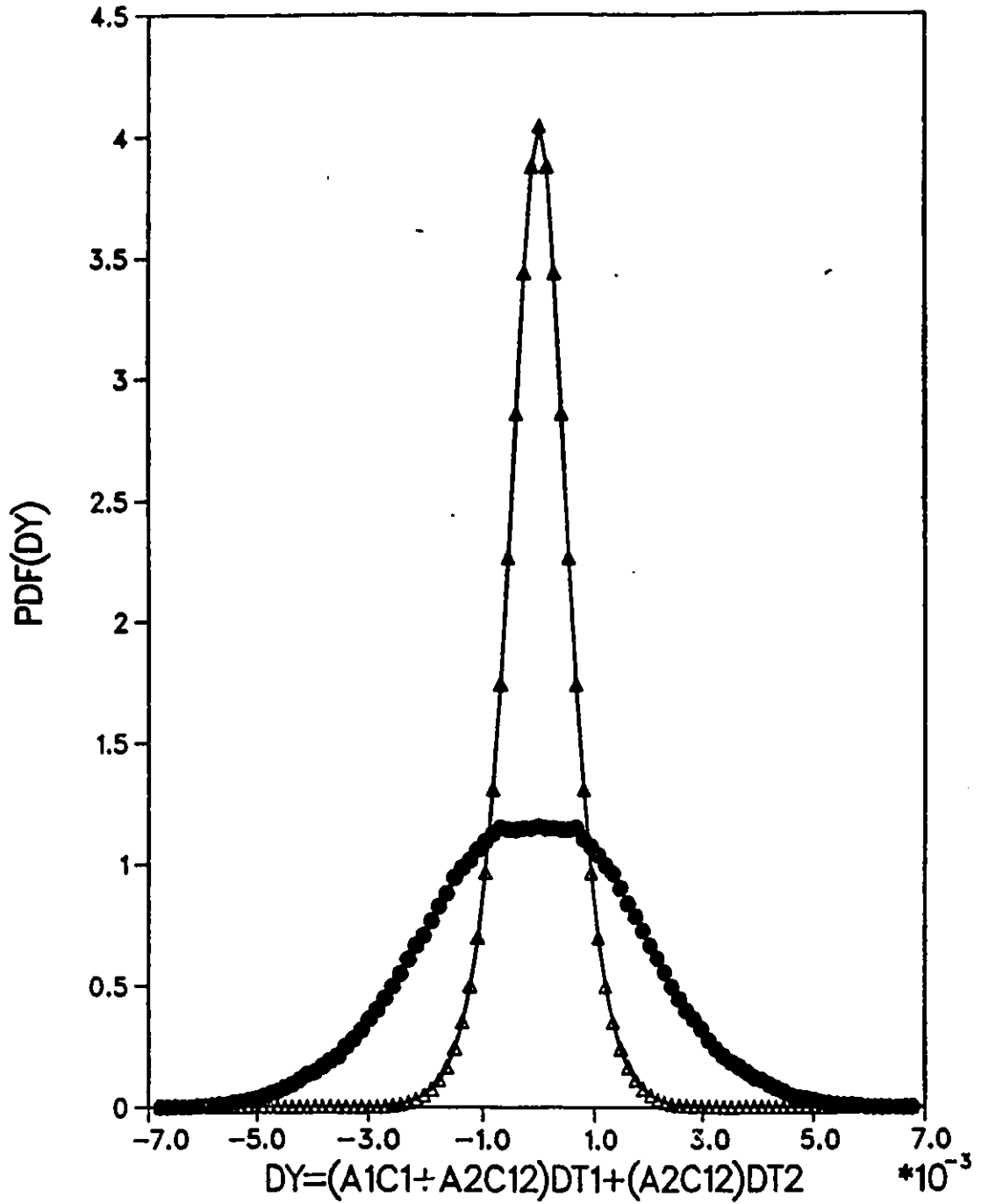


Figure 5.10: dY distribution for which $\theta_1 \in [75^\circ, 90^\circ]$ and $(\theta_1 + \theta_2) \in [75^\circ, 90^\circ]$.

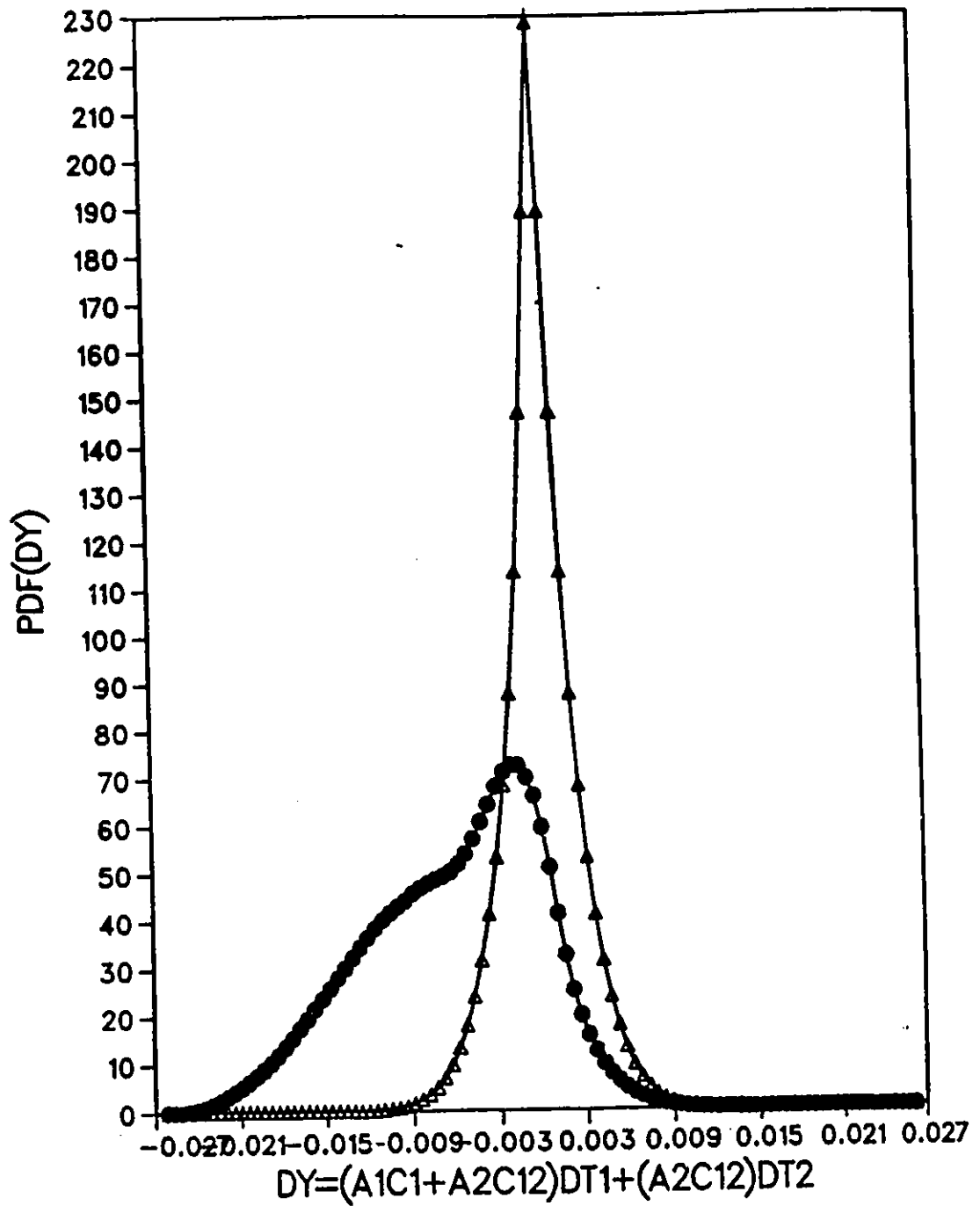


Figure 5.11: dY distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_1) and $(\theta_1 + \theta_2) \in [-180^\circ, 180^\circ]$, (the whole domain of $(\theta_1 + \theta_2)$.)

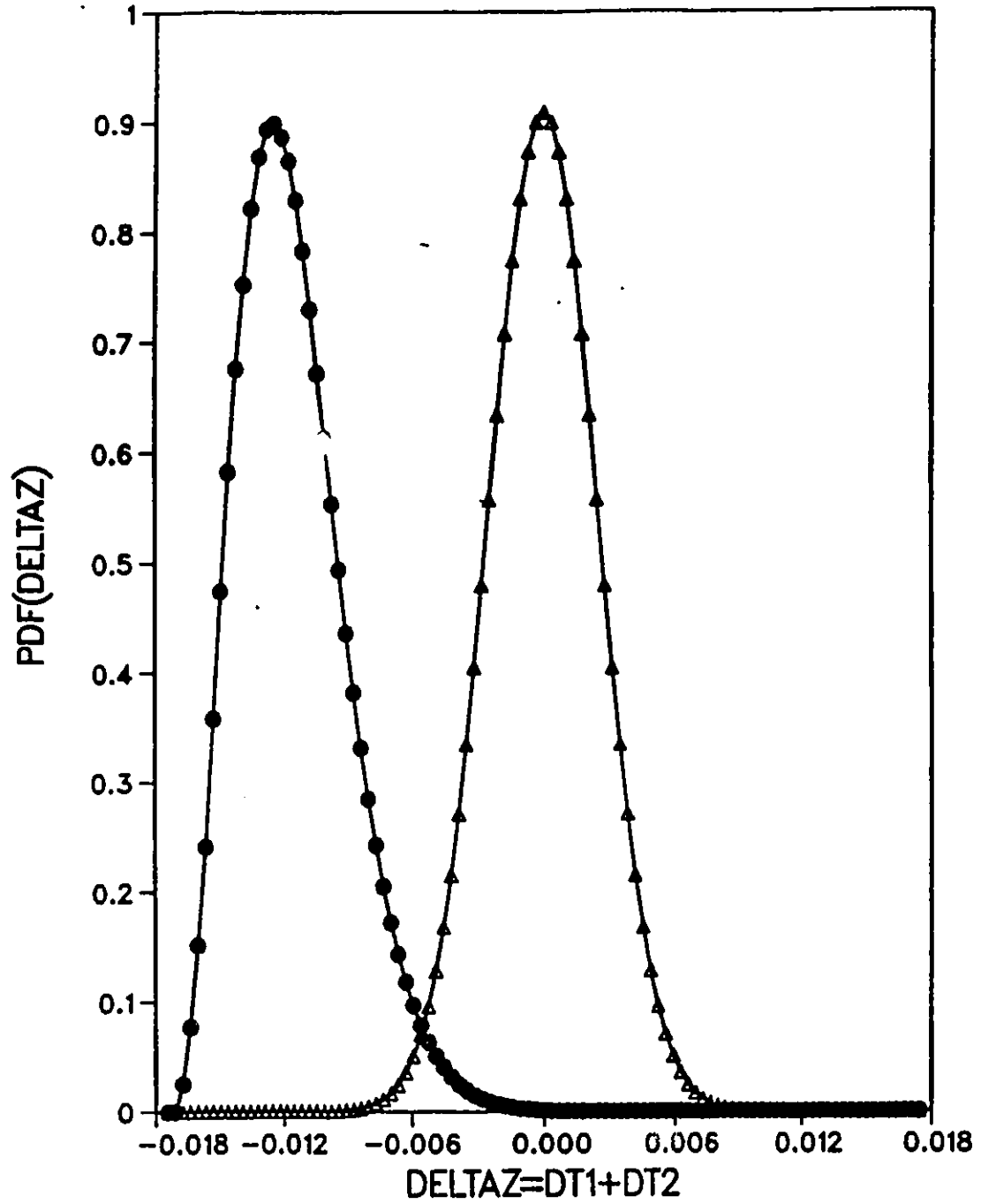


Figure 5.12: δ_z distribution for which $\theta_1 \in [0^\circ, 15^\circ]$ and $(\theta_1 + \theta_2) \in [0^\circ, 15^\circ]$.

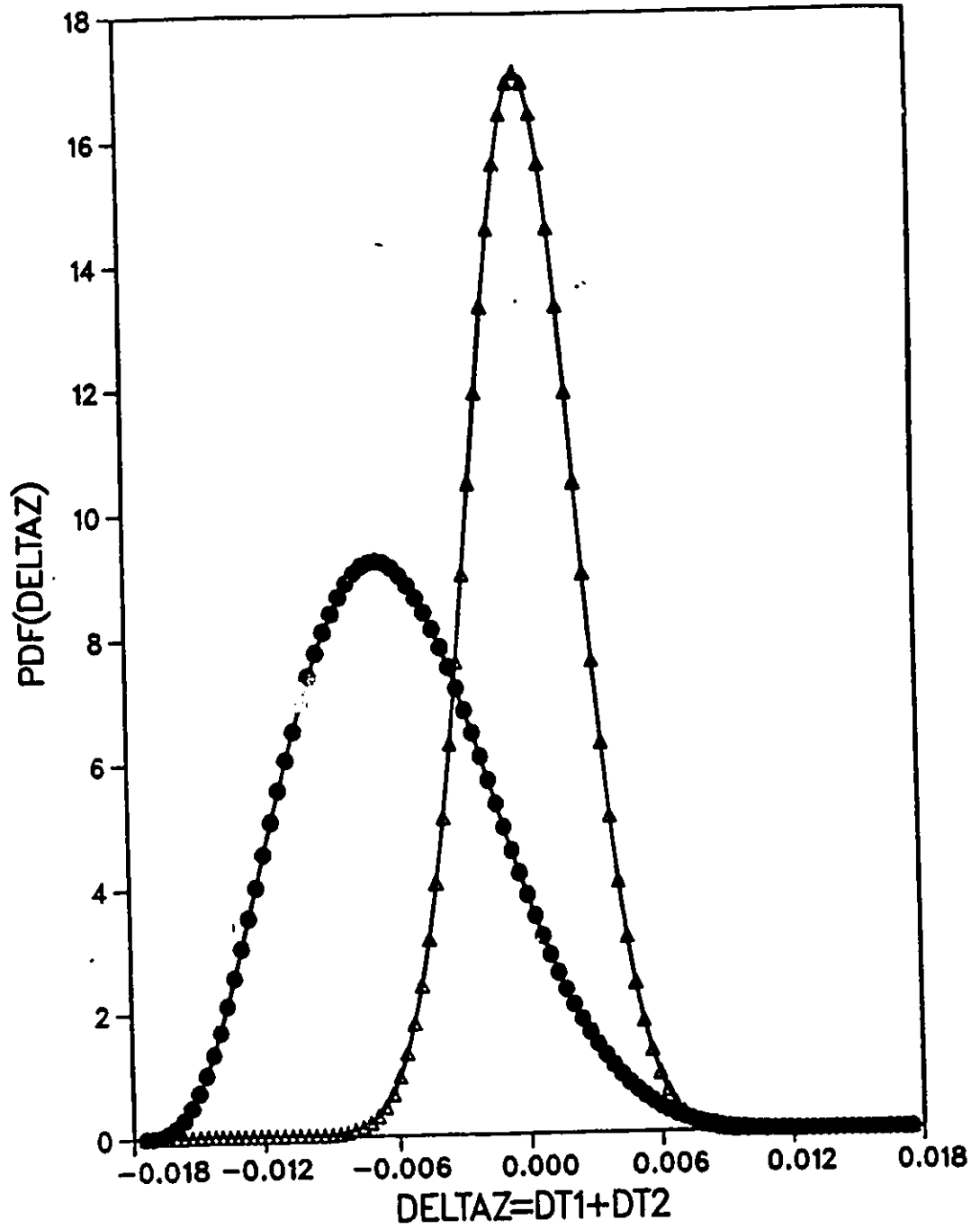


Figure 5.13: δ_z distribution for which $\theta_1 \in (15^\circ, 75^\circ)$ and $(\theta_1 + \theta_2) \in (15^\circ, 75^\circ)$.

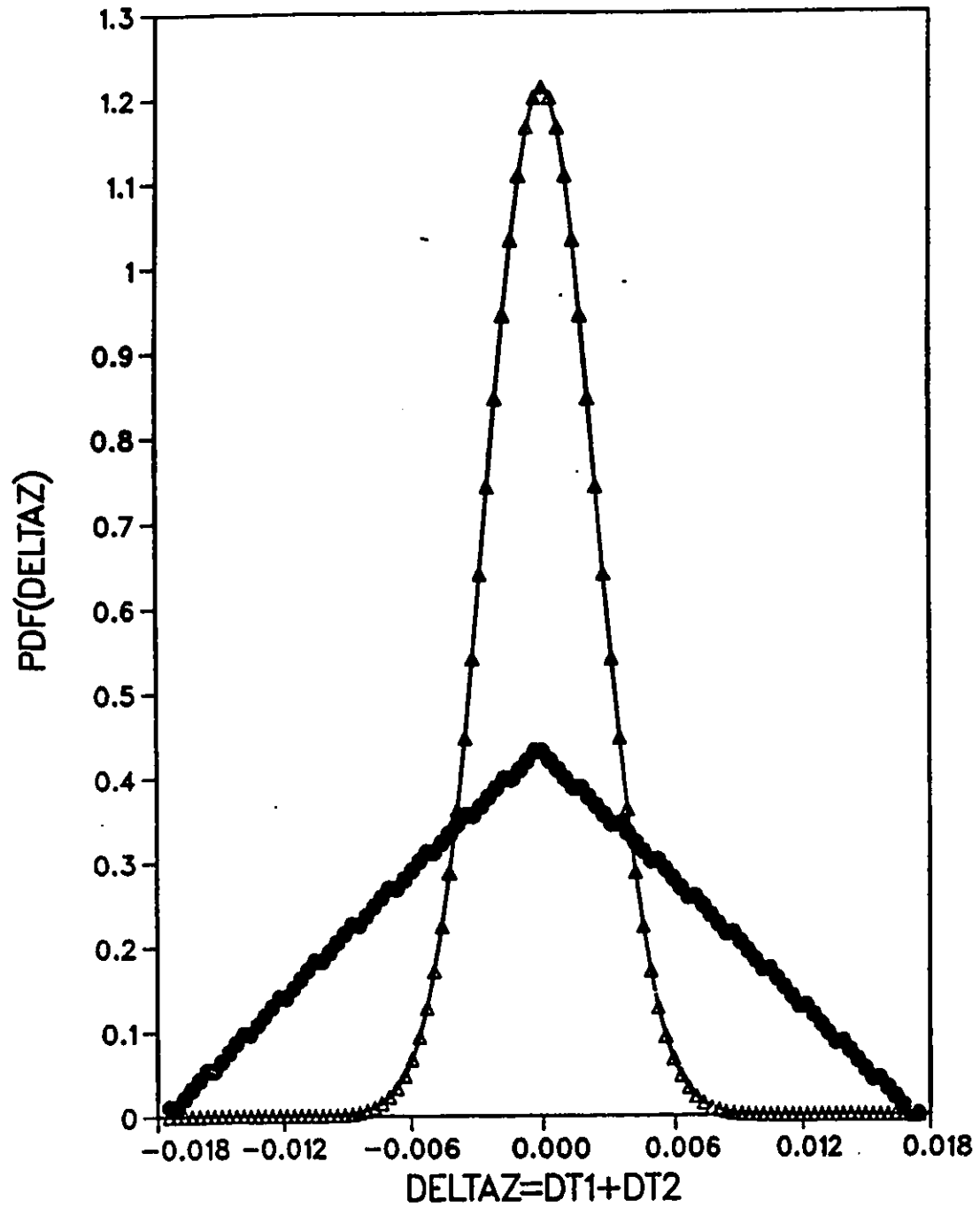


Figure 5.14: δ_z distribution for which $\theta_1 \in [75^\circ, 90^\circ]$ and $(\theta_1 + \theta_2) \in [75^\circ, 90^\circ]$.

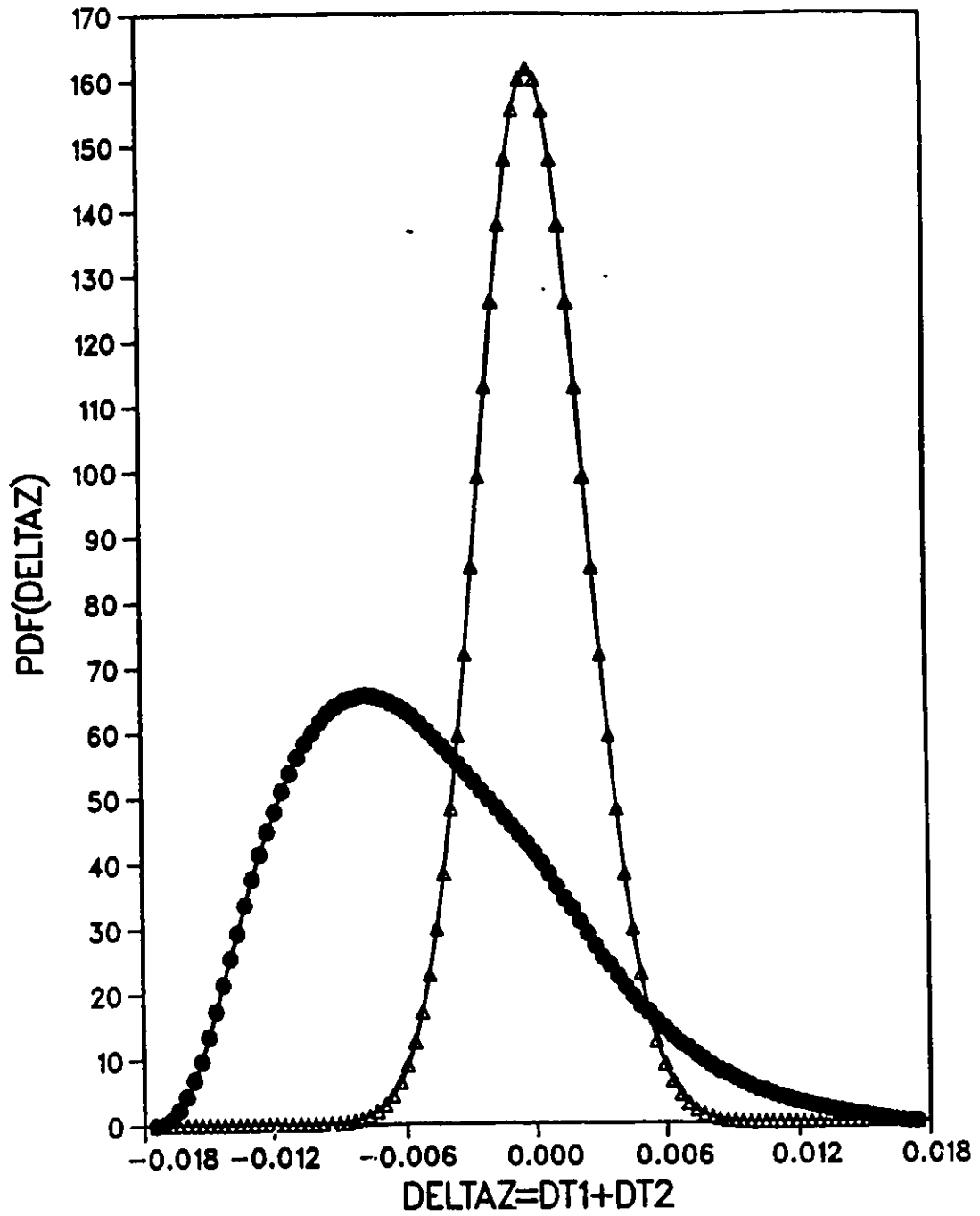


Figure 5.15: δ_z distribution for which $\theta_1 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_1) and $(\theta_1 + \theta_2) \in [-180^\circ, 180^\circ]$, (the whole domain of $(\theta_1 + \theta_2)$.)

In summary, detailed simulations for an arbitrary two degrees-of-freedom vertical planar robot were performed in this chapter. The simulation results show the following characteristics :

- The presented model is general and flexible enough to be applied to analyze any kind of mixed error propagations existing in this two degrees-of-freedom vertical planar robot manipulator.
- In a near horizontal area, both positioning and orientational errors take more biased distributions, which is resulted from the significant effect of gravity on Cartesian positioning error distribution.(Figures 4.3, 4.7 and 4.11)
- In a near vertical area, the positioning and orientational errors take symmetric distributions such as uniform distributions which can not be modeled accurately by the symmetric normal distributions. (Figures 4.5, 4.9 and 4.13)
- The presented model is generally more realistic and is easy to be applied to the error propagation analysis of a two degrees-of-freedom robot manipulator.

Chapter 6

Stochastic Error Propagation Analysis of a Three Degrees-of-Freedom Robot Manipulator

The preceding two chapters have provided the basic illustration of the presented general stochastic model. The computations performed for the simulation of chapter 5 have shown that numerical difficulty is experienced in the evaluation of the positioning and orientational error distribution functions of robot's end-effector when the number of the output supplementary random variables are higher than five. Therefore, it is very difficult to include all link parameter errors in the computational approach for obtaining the positioning and orientational error distribution functions of a manipulator's end-effector.

In fact, four of the five link parameters in the presented A transformation matrix are so-called geometric parameters and they are easy to be well calibrated. Consequently, after calibration only the joint variable errors will have a dominant effect in determining the positioning and orientational error distribution functions of the end-effector.

A general stochastic approach and simulations based on the previously mentioned facts are presented in this chapter for a three degrees-of-freedom spatial

robot manipulator.

6.1 Error Propagation Model

The schematic drawing of the three degrees-of-freedom robot manipulator considered in this chapter is shown in Figure 6.1 [18].

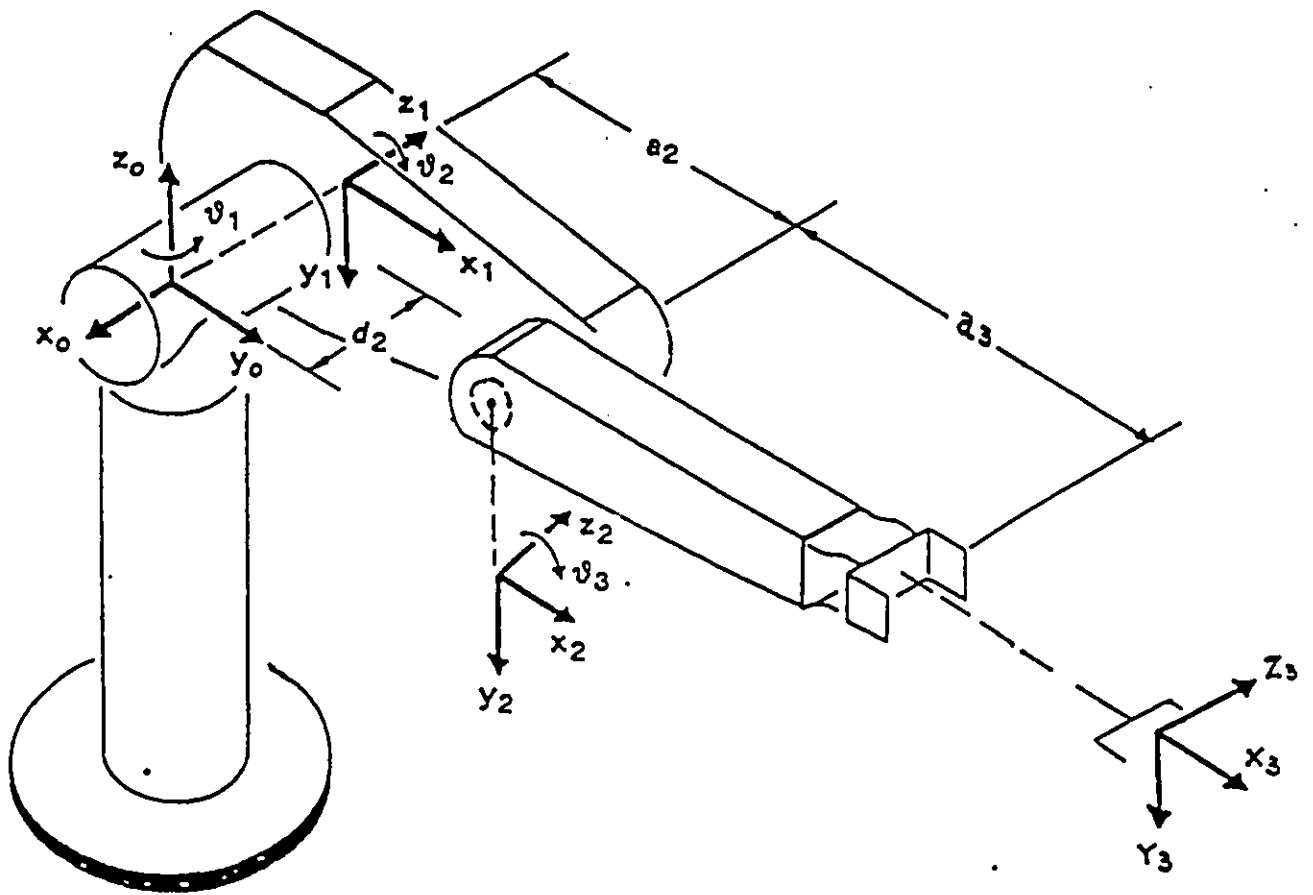


Figure 6.1: schematic drawing of the 3 DOF manipulator for simulation

Using the D-H convention, the link parameter table can be established as

Table 6.1: link parameter table

link n	variable	α_n	a_n	d_n	β_n
1	θ_1	-90°	0	0	0°
2	θ_2	0°	a_2	d_2	0°
3	θ_3	0°	a_3	0	0°

From equation (A.3) defining the modified A transformation matrix, the A matrices can be derived as

$$A_1 = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.1)$$

$$A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2 C_2 \\ S_2 & C_2 & 0 & a_2 S_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.2)$$

$$A_3 = \begin{bmatrix} C_3 & -S_3 & 0 & a_3 C_3 \\ S_3 & C_3 & 0 & a_3 S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.3)$$

Basing on the A transformation matrices calculated beforehand, the T matrices defining the transformation between each coordinate frame and base frame can be obtained as

$$\begin{aligned} T_1 &= A_1 \\ &= \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (6.4)$$

$$\begin{aligned}
T_2 &= A_1 \cdot A_2 \\
&= \begin{bmatrix} C_1 C_2 & -C_1 S_2 & -S_1 & a_2 C_1 C_2 - d_2 S_1 \\ S_1 C_2 & -S_1 S_2 & C_1 & a_2 S_1 C_2 + d_2 C_1 \\ -S_2 & -C_2 & 0 & -a_2 S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{6.5}$$

$$\begin{aligned}
T_3 &= A_1 \cdot A_2 \cdot A_3 \\
&= \begin{bmatrix} C_1 C_{23} & -C_1 S_{23} & -S_1 & a_3 C_1 C_{23} + a_2 C_1 C_2 - d_2 S_1 \\ S_1 C_{23} & -S_1 S_{23} & C_1 & a_3 S_1 C_{23} + a_2 S_1 C_2 + d_2 C_1 \\ -S_{23} & -C_{23} & 0 & -a_3 S_{23} - a_2 S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{6.6}$$

The coefficient matrices in the general error propagation equations defined by (2.60) and (2.61), $R_\theta \cdots L_\beta$, can be calculated as

$$\begin{aligned}
R_{\theta,1} &= R_0^0 * K1_1 \\
&= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\end{aligned} \tag{6.7}$$

$$\begin{aligned}
R_{\theta,2} &= R_1^0 * K1_2 \\
&= \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{6.8}$$

$$\begin{aligned}
R_{\theta,3} &= R_2^0 * K1_3 \\
&= \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{6.9}$$

$$\begin{aligned}
R_{\alpha,1} &= R_0^0 * K2_1 \\
&= \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{6.10}$$

$$\begin{aligned}
R_{\alpha,2} &= R_1^0 * K2_2 \\
&= \begin{bmatrix} C_1 C_2 \\ S_1 C_2 \\ -S_2 \end{bmatrix}
\end{aligned} \tag{6.11}$$

$$\begin{aligned}
R_{\alpha,3} &= R_2^0 * K2_3 \\
&= \begin{bmatrix} C_1 C_{23} \\ S_1 C_{23} \\ -S_{23} \end{bmatrix}
\end{aligned} \tag{6.12}$$

$$\begin{aligned}
R_{\beta,1} &= R_0^0 * K5_1 \\
&= \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}
\end{aligned} \tag{6.13}$$

$$\begin{aligned}
R_{\beta,2} &= R_1^0 * K5_2 \\
&= \begin{bmatrix} -C_1 S_2 \\ -S_1 S_2 \\ -C_2 \end{bmatrix}
\end{aligned} \tag{6.14}$$

$$\begin{aligned}
R_{\beta,3} &= R_2^0 * K5_3 \\
&= \begin{bmatrix} -C_1 S_{23} \\ -S_1 S_{23} \\ -C_{23} \end{bmatrix}
\end{aligned} \tag{6.15}$$

$$\begin{aligned}
L_{\theta,1} &= (R_{\theta,1} \times P_3^0) + P_0^0 \times (R_0^0 * K1_1) \\
&= \begin{bmatrix} -a_3 S_1 C_{23} - a_2 S_1 C_2 - d_2 C_1 \\ a_3 C_1 C_{23} + a_2 C_1 C_2 - d_2 S_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{6.16}$$

$$\begin{aligned}
L_{\theta,2} &= (R_{\theta,2} \times P_3^0) + P_1^0 \times (R_1^0 * K1_2) \\
&= \begin{bmatrix} -a_3 C_1 S_{23} - a_2 C_1 S_2 \\ -a_3 S_1 S_{23} - a_2 S_1 S_2 \\ -a_3 C_{23} - a_2 C_2 \end{bmatrix}
\end{aligned} \tag{6.17}$$

$$\begin{aligned}
L_{\theta,3} &= (R_{\theta,3} \times P_3^0) + P_2^0 \times (R_2^0 * K_{13}) \\
&= \begin{bmatrix} -a_3 C_1 S_{23} \\ -a_3 S_1 S_{23} \\ -a_3 C_{23} \end{bmatrix}
\end{aligned} \tag{6.18}$$

$$\begin{aligned}
L_{d,1} &= R_{\theta,1} \\
&= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\end{aligned} \tag{6.19}$$

$$\begin{aligned}
L_{d,2} &= R_{\theta,2} \\
&= \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{6.20}$$

$$\begin{aligned}
L_{d,3} &= R_{\theta,3} \\
&= \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{6.21}$$

$$\begin{aligned}
L_{a,1} &= R_{\alpha,1} \\
&= \begin{bmatrix} C_1 \\ S_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{6.22}$$

$$\begin{aligned}
L_{a,2} &= R_{\alpha,2} \\
&= \begin{bmatrix} C_1 C_2 \\ S_1 C_2 \\ -S_2 \end{bmatrix}
\end{aligned} \tag{6.23}$$

$$\begin{aligned}
L_{a,3} &= R_{\alpha,3} \\
&= \begin{bmatrix} C_1 C_{23} \\ S_1 C_{23} \\ -S_{23} \end{bmatrix}
\end{aligned} \tag{6.24}$$

$$\begin{aligned}
L_{\alpha,1} &= (R_{\alpha,1} \times P_3^0) + P_0^0 \times (R_0^0 * K2_1) + (R_0^0 * K3_1) \\
&= \begin{bmatrix} -a_3 S_1 S_{23} - a_2 S_1 S_2 \\ a_3 C_1 S_{23} + a_2 C_1 S_2 \\ d_2 \end{bmatrix} \tag{6.25}
\end{aligned}$$

$$\begin{aligned}
L_{\alpha,2} &= (R_{\alpha,2} \times P_3^0) + P_1^0 \times (R_1^0 * K2_2) + (R_1^0 * K3_2) \\
&= \begin{bmatrix} -a_3 S_1 S_3 + d_2 C_1 S_2 \\ a_3 C_1 S_3 + d_2 S_1 S_2 \\ d_2 C_2 \end{bmatrix} \tag{6.26}
\end{aligned}$$

$$\begin{aligned}
L_{\alpha,3} &= (R_{\alpha,3} \times P_3^0) + P_2^0 \times (R_2^0 * K2_2) + (R_2^0 * K3_3) \\
&= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.27}
\end{aligned}$$

$$\begin{aligned}
L_{\beta,1} &= (R_{\beta,1} \times P_3^0) + (R_0^0 * K4_1) \\
&= \begin{bmatrix} a_3 S_1 C_{23} + a_2 S_1 C_2 + d_2 C_1 \\ -a_3 C_1 C_{23} - a_2 C_1 C_2 + d_2 S_1 \\ 0 \end{bmatrix} \tag{6.28}
\end{aligned}$$

$$\begin{aligned}
L_{\beta,2} &= (R_{\beta,2} \times P_3^0) + (R_1^0 * K4_2) + P_1^0 \times (R_1^0 * K5_2) \\
&= \begin{bmatrix} a_3 S_1 C_3 \\ -a_3 C_1 C_3 \\ 0 \end{bmatrix} \tag{6.29}
\end{aligned}$$

$$\begin{aligned}
L_{\beta,3} &= (R_{\beta,3} \times P_3^0) + (R_2^0 * K4_3) + P_2^0 \times (R_2^0 * K5_3) \\
&= \begin{bmatrix} a_3 S_1 \\ -a_3 C_1 \\ 0 \end{bmatrix} \tag{6.30}
\end{aligned}$$

By substituting (6.7) to (6.15) into (2.60), the orientational error of the end-effector can be calculated as

δ

$$\begin{aligned}
&= \begin{bmatrix} 0 & -S_1 & -S_1 \\ 0 & C_1 & C_1 \\ 1 & 0 & 0 \end{bmatrix} \Delta\theta + \begin{bmatrix} C_1 & C_1C_2 & C_1C_{23} \\ S_1 & S_1C_2 & S_1C_{23} \\ 0 & -S_2 & -S_{23} \end{bmatrix} \Delta\alpha + \begin{bmatrix} 0 & -C_1S_2 & -C_1S_{23} \\ 0 & -S_1S_2 & -S_1S_{23} \\ -1 & -C_2 & -C_{23} \end{bmatrix} \Delta\beta \\
&\quad \begin{bmatrix} -S_1\Delta\theta_2 - S_1\Delta\theta_3 + C_1\Delta\alpha_1 + C_1C_2\Delta\alpha_2 + C_1C_{23}\Delta\alpha_3 - C_1S_2\Delta\beta_2 - C_1S_{23}\Delta\beta_3 \\ C_1\Delta\theta_2 + C_1\Delta\theta_3 + S_1\Delta\alpha_1 + S_1C_2\Delta\alpha_2 + S_1C_{23}\Delta\alpha_3 - S_1S_2\Delta\beta_2 - S_1S_{23}\Delta\beta_3 \\ \Delta\theta_1 - S_2\Delta\alpha_2 - S_{23}\Delta\alpha_3 - \Delta\beta_1 - C_2\Delta\beta_2 - C_{23}\Delta\beta_3 \end{bmatrix} \quad (6.31)
\end{aligned}$$

and by substituting (6.16) to (6.30) into (2.61), the positioning error of the end-effector can be found as

$$\begin{aligned}
dP &= \begin{bmatrix} -a_3S_1C_{23} - a_2S_1C_2 - d_2C_1 & -a_3C_1S_{23} - a_2C_1S_2 & -a_3C_1S_{23} \\ a_3C_1C_{23} + a_2C_1C_2 - d_2S_1 & -a_3S_1S_{23} - a_2S_1S_2 & -a_3S_1S_{23} \\ 0 & -a_3C_{23} - a_2C_2 & -a_3C_{23} \end{bmatrix} \Delta\theta \\
&+ \begin{bmatrix} 0 & -S_1 & -S_1 \\ 0 & C_1 & C_1 \\ 1 & 0 & 0 \end{bmatrix} \Delta d \\
&+ \begin{bmatrix} C_1 & C_1C_2 & C_1C_{23} \\ S_1 & S_1C_2 & S_1C_{23} \\ 0 & -S_2 & -S_{23} \end{bmatrix} \Delta a \\
&+ \begin{bmatrix} -a_3S_1S_{23} - a_2S_1S_2 & -a_3S_1S_3 & 0 \\ a_3C_1S_{23} + a_2C_1S_2 & a_3C_1S_3 + d_2S_1S_2 & 0 \\ d_2 & d_2C_2 & 0 \end{bmatrix} \Delta\alpha \\
&+ \begin{bmatrix} a_3S_1C_{23} + a_2S_1C_2 + d_2C_1 & a_3S_1C_3 & a_3S_1 \\ -a_3C_1C_{23} - a_2C_1C_2 + d_2S_1 & -a_3C_1C_3 & -a_3C_1 \\ 0 & 0 & 0 \end{bmatrix} \Delta\beta
\end{aligned}$$

$$\begin{aligned}
& \left[\begin{aligned}
& (-a_3 S_1 C_{23} - a_2 S_1 C_2 - d_2 C_1) \Delta \theta_1 + (-a_3 C_1 S_{23} - a_2 C_1 S_2) \Delta \theta_2 \\
& -a_3 C_1 S_{23} \Delta \theta_3 - S_1 \Delta d_2 - S_1 \Delta d_3 + C_1 \Delta a_1 + C_1 C_2 \Delta a_2 \\
& + C_1 C_{23} \Delta a_3 + (-a_3 S_1 S_{23} - a_2 S_1 S_2) \Delta \alpha_1 + (-a_3 S_1 S_3 + d_2 C_1 S_2) \Delta \alpha_2 \\
& + (a_3 S_1 C_{23} + a_2 S_1 C_2 + d_2 C_1) \Delta \beta_1 + a_3 S_1 C_3 \Delta \beta_2 + a_3 S_1 \Delta \beta_3 \\
& (a_3 C_1 C_{23} + a_2 C_1 C_2 - d_2 S_1) \Delta \theta_1 + (-a_3 S_1 S_{23} - a_2 S_1 S_2) \Delta \theta_2 \\
& -a_3 S_1 S_{23} \Delta \theta_3 + C_1 \Delta d_2 + C_1 \Delta d_3 + S_1 \Delta a_1 \\
& + S_1 C_2 \Delta a_2 + S_1 C_{23} \Delta a_3 + (a_3 C_1 S_{23} + a_2 C_1 S_2) \Delta \alpha_1 \\
& + (a_3 C_1 S_3 + d_2 S_1 S_2) \Delta \alpha_2 + (-a_3 C_1 C_{23} - a_2 C_1 C_2 + d_2 S_1) \Delta \beta_1 \\
& -a_3 C_1 C_3 \Delta \beta_2 - a_3 C_1 \Delta \beta_3 \\
& (-a_3 C_{23} - a_2 C_2) \Delta \theta_2 - a_3 C_{23} \Delta \theta_3 + \Delta d_1 - S_2 \Delta a_2 - S_{23} \Delta a_3 \\
& + d_2 \Delta \alpha_1 + d_2 C_2 \Delta \alpha_2
\end{aligned} \right] \quad (6.32)
\end{aligned}$$

Equation (6.31) and (6.32) completely define the error propagation model based on the five link parameter A transformation matrix for the three degrees-of-freedom spatial robot manipulator shown in Figure 6.1. The stochastic analysis based on these two equations will be given in the next section.

6.2 Stochastic Error Propagation Model

For the manipulator shown in Figure 6.1, the error propagation model is defined by equations (6.31) and (6.32). In order to find the distribution functions of the Cartesian positioning and orientational errors, the method described in equation (3.34) will be used to find the positioning and orientational error distribution functions of the end-effector.

The deriving procedures for finding the error distribution functions for dX , dY , dZ , δ_X , δ_Y , and δ_Z are structurally the same. Consequently, a detailed deriving procedure will be given here for dX only. From (6.32), the error propagation model for dX can be written in the matrix form of (2.61) as

$$\begin{aligned}
dX = & [L_{\theta,x}(\theta)] \Delta \theta + [L_{d,x}(\theta)] \Delta d + [L_{a,x}(\theta)] \Delta a \\
& + [L_{\alpha,x}(\theta)] \Delta \alpha + [L_{\beta,x}(\theta)] \Delta \beta \quad (6.33)
\end{aligned}$$

where

$$[L_{\theta,X}(\theta)] = [-a_3S_1C_{23} - a_2S_1C_2 - d_2C_1, -a_3C_1S_{23} - a_2C_1S_2, -a_3C_1S_{23}] \quad (6.34)$$

$$[L_{d,X}(\theta)] = [0, -S_1, -S_1] \quad (6.35)$$

$$[L_{a,x}(\theta)] = [C_1, C_1C_2, C_1C_{23}] \quad (6.36)$$

$$[L_{\alpha,X}(\theta)] = [-a_3S_1S_{23} - a_2S_1S_2, -a_3S_1S_3 + d_2C_1S_2, 0] \quad (6.37)$$

$$[L_{\beta,X}(\theta)] = [a_3S_1C_{23} + a_2S_1C_2 + d_2C_1, a_3S_1C_3, a_3S_1] \quad (6.38)$$

Due to computational limitation, it is very difficult to include all the link parameter errors in the numerical simulations. In fact, among the five link parameter errors, four of them are geometrical link parameter errors and they are easy to be well calibrated out. Consequently, for a well calibrated robot manipulator, only the joint variable errors will become the dominant factor in determining the positioning and orientational error distributions of the end-effector.

For a well calibrated robot, the positioning error dX of the end-effector can thus be approximated by assuming that the geometric link parameter errors are negligible. Equation (6.33) can be simplified to

$$\begin{aligned} dX &= (-a_1S_1C_{23} - a_2S_1C_2 - d_2C_1)\Delta\theta_1 + (-a_3C_1S_{23} - a_2C_1S_2)\Delta\theta_2 \\ &\quad - a_3C_1S_{23}\Delta\theta_3 \end{aligned} \quad (6.39)$$

we choose the output random parameters as

$$\begin{aligned} Y_1 &= dX \\ &= (-a_1S_1C_{23} - a_2S_1C_2 - d_2C_1)\Delta\theta_1 + (-a_3C_1S_{23} - a_2C_1S_2)\Delta\theta_2 \\ &\quad - a_3C_1S_{23}\Delta\theta_3 \end{aligned} \quad (6.40)$$

$$Y_2 = \theta_1 \quad (6.41)$$

$$Y_3 = \theta_2 \quad (6.42)$$

$$Y_4 = \theta_3 \quad (6.43)$$

$$Y_5 = \Delta\theta_1 \quad (6.44)$$

$$Y_6 = \Delta\theta_2 \quad (6.45)$$

For which, the input random variables are the solutions of equations (6.40) to (6.45)

$$X_1 = Y_2 = \theta_1 = h_1(Y_1, \dots, Y_6) \quad (6.46)$$

$$X_2 = Y_3 = \theta_2 = h_2(Y_1, \dots, Y_6) \quad (6.47)$$

$$X_3 = Y_4 = \theta_3 = h_3(Y_1, \dots, Y_6) \quad (6.48)$$

$$X_4 = Y_5 = \Delta\theta_1 = h_4(Y_1, \dots, Y_6) \quad (6.49)$$

$$X_5 = Y_6 = \Delta\theta_2 = h_5(Y_1, \dots, Y_6) \quad (6.50)$$

$$\begin{aligned} X_6 &= \Delta\theta_3 \\ &= \frac{Y_1 - (-a_3SY_2CY_{34} - a_2SY_2CY_3 - d_2CY_2)Y_5 - (-a_3CY_2SY_{34} - a_2CY_2SY_3)Y_6}{-a_3CY_2SY_{34}} \\ &= h_6(Y_1, \dots, Y_6) \end{aligned} \quad (6.51)$$

From the above definition of input and output random variables, the transformation between (Y_1, Y_2, \dots, Y_6) and (X_1, X_2, \dots, X_6) is related by the Jacobian. The determinant of the Jacobian is defined as

$$\begin{aligned} J(X_1, \dots, X_6) &= J(h_1(Y_1, \dots, Y_6), \dots, h_6(Y_1, \dots, Y_6)) \\ &= \frac{\partial(Y_1, Y_2, Y_3, Y_4, Y_5, Y_6)}{\partial(\theta_1, \theta_2, \theta_3, \Delta\theta_1, \Delta\theta_2, \Delta\theta_3)} \\ &= \begin{vmatrix} J_{11} & J_{12} & J_{13} & J_{14} & J_{15} & -a_3CY_2SY_{34} \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} \\ &= a_3CY_2SY_{34} \end{aligned} \quad (6.52)$$

where the coefficients J_{11} to J_{15} do not affect the Jacobian's value. Thus, they are not calculated here. And the Jacobian must satisfy the condition $J = a_3CY_2SY_{34} \neq 0$.

It is obvious that the transformation between (Y_1, \dots, Y_6) and (X_1, \dots, X_6) is a one-to-one transformation. From equation (3.5), the joint probability density function $f_{Y_1, \dots, Y_6}(y_1, \dots, y_6)$ of Y_1, \dots, Y_6 is given by

$$f_{Y_1, \dots, Y_6}(y_1, \dots, y_6) = f_{X_1, \dots, X_6}(x_1, \dots, x_6) / |J(X_1, \dots, X_6)| \quad (6.53)$$

where X_1 to X_6 are replaced by the solutions (6.46) to (6.51). Equation (6.53) can be rewritten as

$$\begin{aligned} & f_{Y_1, \theta_1, \theta_2, \theta_3, \Delta\theta_1, \Delta\theta_2}(y_1, \theta_1, \theta_2, \theta_3, \Delta\theta_1, \Delta\theta_2) \\ &= f_{\theta_1, \theta_2, \theta_3, \Delta\theta_1, \Delta\theta_2, \Delta\theta_3}(\theta_1, \theta_2, \theta_3, \Delta\theta_1, \Delta\theta_2, \\ & \quad \frac{y_1 - (-a_3 S_{Y2} C_{Y34} - a_2 S_{Y2} C_{Y3} - d_2 C_{Y2}) y_3 - (-a_3 C_{Y2} S_{Y34} - a_2 C_{Y2} S_{Y3}) y_4}{-a_3 C_{Y2} S_{Y34}}) \cdot \frac{1}{|a_3 C_{Y2} S_{Y34}|} \end{aligned} \quad (6.54)$$

The positioning error dX of the end-effector can be determined from the marginal distribution of (6.50) by integrating on Y_2 to Y_6 over the entire domain of these supplementary output random parameters as

$$\begin{aligned} & f_{dX}(dx) \\ &= \int_{\theta_1} \int_{\theta_2} \int_{\theta_3} \int_{\Delta\theta_1} \int_{\Delta\theta_2} f(\theta_1, \theta_2, \theta_3, \Delta\theta_1, \Delta\theta_2, \\ & \quad \frac{dx - (-a_3 S_1 C_{23} - a_2 S_1 C_2 - d_2 C_1) \Delta\theta_1 - (-a_3 C_1 S_{23} - a_2 C_1 S_2) \Delta\theta_2}{-a_3 C_1 S_{23}}) \\ & \quad / |a_3 C_1 S_{23}| \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\theta_3) \cdot d(\Delta\theta_1) \cdot d(\Delta\theta_2) \quad ; a_3 C_1 S_{23} \neq 0 \end{aligned} \quad (6.55)$$

Due to the special structure of the robot manipulator considered in this chapter, in which the first joint axis is vertical, the first joint parameter error can be modeled as configuration independent. Therefore, if we divide the domains of θ_2 and $(\theta_2 + \theta_3)$ into L_2 and L_3 independent smaller sub-domains respectively such that in each sub-domain the gravity effect can be evaluated approximately the same, then in each sub-domain all the link parameter errors become independent and configuration irrelevant. Based on the afore mentioned fact, the error distribution function for dX can be further expanded as

$$\begin{aligned} & f_{dX}(dx) \\ &= \int_{i_2=1}^{L_2} \int_{i_3=1}^{L_3} \int_{\theta_1} \int_{\theta_2, i_2} \int_{\theta_3, i_3} \int_{\Delta\theta_1} \int_{\Delta\theta_2} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\theta_3}(\theta_3) \\ & \quad f_{\Delta\theta_1}(\Delta\theta_1) f_{\Delta\theta_2}(\Delta\theta_2) f_{\Delta\theta_3} \left(\frac{dx - (-a_3 S_1 C_{23} - a_2 S_1 C_2 - d_2 C_1) \Delta\theta_1 - (-a_3 C_1 S_{23} - a_2 C_1 S_2) \Delta\theta_2}{-a_3 C_1 S_{23}} \right) \\ & \quad \frac{1}{|a_3 C_1 S_{23}|} \cdot d(\theta_1) \cdot d(\theta_2) \cdot d(\theta_3) \cdot d(\Delta\theta_1) \cdot d(\Delta\theta_2) \quad ; a_3 C_1 S_{23} \neq 0 \end{aligned} \quad (6.56)$$

Using the same procedures as above, for the other positioning and orientational error distribution functions, we obtain

$$\begin{aligned} & f_{dY}(dy) \\ &= \sum_{i_2=1}^{L_2} \sum_{i_3=1}^{L_3} \int_{\theta_1} \int_{\theta_2, i_2} \int_{\theta_3, i_3} \int_{\Delta\theta_1} \int_{\Delta\theta_2} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\theta_3}(\theta_3) f_{\Delta\theta_1}(\Delta\theta_1) f_{\Delta\theta_2}(\Delta\theta_2) \\ & \quad f_{\Delta\theta_3} \left(\frac{dy - (a_3 C_1 C_{23} + a_2 C_1 C_2 - d_2 S_1) \Delta\theta_1 - (-a_3 S_1 S_{23} - a_2 S_1 S_2) \Delta\theta_2}{-a_3 S_1 S_{23}} \right) \frac{1}{|a_3 S_1 S_{23}|} \\ & \quad d(\theta_1) \cdot d(\theta_2) \cdot d(\theta_3) \cdot d(\Delta\theta_1) \cdot d(\Delta\theta_2); a_3 S_1 S_{23} \neq 0 \end{aligned} \quad (6.57)$$

$$\begin{aligned}
& f_{dz}(dz) \\
&= \sum_{i_2=1}^{L_2} \sum_{i_3=1}^{L_3} \int_{\theta_2, i_2} \int_{\theta_3, i_3} \int_{\Delta\theta_2} f_{\theta_2}(\theta_2) f_{\theta_3}(\theta_3) f_{\Delta\theta_2}(\Delta\theta_2) \\
& f_{\Delta\theta_3}\left(\frac{dz - (-a_3 C_{23} - a_2 C_2) \Delta\theta_2}{-a_3 C_{23}}\right) \cdot \frac{1}{|a_3 C_{23}|} d(\theta_2) \cdot d(\theta_3) \cdot d(\Delta\theta_2); a_3 C_{23} \neq 0
\end{aligned} \tag{6.58}$$

$$\begin{aligned}
& f_{\delta_x}(\delta_x) \\
&= \sum_{i_2=1}^{L_2} \sum_{i_3=1}^{L_3} \int_{\theta_1} \int_{\theta_2, i_2} \int_{\theta_3, i_3} \int_{\Delta\theta_2} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\theta_3}(\theta_3) \\
& f_{\Delta\theta_2}(\Delta\theta_2) f_{\Delta\theta_3}\left(\frac{\delta_x + S_1 \Delta\theta_2}{-S_1}\right) \cdot \frac{1}{|S_1|} d(\theta_1) \cdot d(\theta_2) \cdot d(\theta_3) \cdot d(\Delta\theta_2); S_1 \neq 0
\end{aligned} \tag{6.59}$$

$$\begin{aligned}
& f_{\delta_y}(\delta_y) \\
&= \sum_{i_2=1}^{L_2} \sum_{i_3=1}^{L_3} \int_{\theta_1} \int_{\theta_2, i_2} \int_{\theta_3, i_3} \int_{\Delta\theta_2} f_{\theta_1}(\theta_1) f_{\theta_2}(\theta_2) f_{\theta_3}(\theta_3) f_{\Delta\theta_2}(\Delta\theta_2) \\
& f_{\Delta\theta_3}\left(\frac{\delta_y - C_1 \Delta\theta_2}{C_1}\right) \cdot \frac{1}{C_1} d(\theta_1) \cdot d(\theta_2) d(\theta_3) d(\Delta\theta_2); C_1 \neq 0
\end{aligned} \tag{6.60}$$

$$f_{\delta_z}(\delta_z) = f_{\Delta\theta_1}(\Delta\theta_1) \tag{6.61}$$

The last six equations represent the general stochastic model for the positioning or orientational error distribution functions of the three degrees-of-freedom spatial robot manipulator shown in Figure 6.1. The numerical simulations based on these six general stochastic error propagation models will be presented in the next section.

6.3 Numerical Stochastic Simulations

The stochastic error propagation models for the positioning or orientational errors of the end-effector are given by the last six equations of the last section. As in chapter 4 and 5, it is known that practically, the effect of gravity is different when the mechanical arm modifies its position. For example, when the arm is in the horizontal position, the arm links would be pulled more downward and the distribution function of errors are most likely shifted towards negative errors. Therefore, it is assumed that in different configurations, the positioning or orientational error distribution functions should take different forms of distributions. If we divide the domain into some smaller independent sub-domains, then in each sub-domain the joint parameter error distribution could be approximated by a fixed distribution function. Consequently, the developed stochastic model can be applied easily to obtain the distribution functions of the positioning or orientational errors of an end-effector. In this section, to illustrate the results, the numerical simulations are based on the following assumptions :

- All joints, 1, 2 and 3, are limited to move in $\pm 90^\circ$.

- Due to this robot's particular structure, the first joint parameter error is configuration independent and is assumed to take a uniform distribution in the whole domain of θ_1 .
- When θ_2 is located within $[-15^\circ, 15^\circ]$, $d\theta_2$ is assumed to take a biased beta distribution with beta parameters $r_1 = 2$ and $s_1 = 10$; when $(\theta_2 + \theta_3)$ is located within $[-15^\circ, 15^\circ]$, $d\theta_3$ is assumed to take a biased beta distribution with beta parameters $r'_1 = 2$ and $s'_1 = 10$.
- When θ_2 is located within $[75^\circ, 90^\circ]$ or $[-90^\circ, -75^\circ]$, $d\theta_2$ is assumed to take a uniform distribution with beta parameters $r_3 = 1$ and $s_3 = 1$; when $(\theta_2 + \theta_3)$ is located within $[75^\circ, 105^\circ]$ or $[-105^\circ, -75^\circ]$, $d\theta_3$ is assumed to take a uniform distribution with beta parameters $r'_3 = 1$ and $s'_3 = 1$.
- When θ_2 is located within $(15^\circ, 75^\circ)$ or $(-75^\circ, -15^\circ)$, $d\theta_1$ is assumed to take a beta distribution with beta parameter $r_2 = 2$ and $s_2 = 4$; when $(\theta_2 + \theta_3)$ is located within $(15^\circ, 75^\circ)$ or $(-75^\circ, -15^\circ)$, $d\theta_3$ is assumed to take a biased beta distribution with beta parameter $r'_2 = 2$ and $s'_2 = 4$.
- When $(\theta_2 + \theta_3)$ is located within $[90^\circ, 180^\circ]$ or $[-180^\circ, -90^\circ]$, $d\theta_3$ is assumed to take a biased beta distribution which is the mirror images about Y axis of the the distributions of $d\theta_3$ in the domain $[-90^\circ, 90^\circ]$ (due to the opposite effect of gravity).
- $\sigma_{\Delta\theta_1} = \sigma_{\Delta\theta_2} = \sigma_{\Delta\theta_3} = 0.1^\circ$; $\Delta\theta_1, \Delta\theta_2$ and $\Delta\theta_3$ are all located within the 5σ band.
- All the other link parameter errors are assumed to take nominal values with zero variance.
 1. $\sigma_{\Delta\alpha_1} = \sigma_{\Delta\alpha_2} = \sigma_{\Delta\alpha_3} = 0^\circ$.
 2. $\sigma_{\Delta\beta_1} = \sigma_{\Delta\beta_2} = \sigma_{\Delta\beta_3} = 0^\circ$.
 3. $\sigma_{\Delta a_1} = \sigma_{\Delta a_2} = \sigma_{\Delta a_3} = 0$.
 4. $\sigma_{\Delta d_1} = \sigma_{\Delta d_2} = \sigma_{\Delta d_3} = 0$.

The joint parameter error distributions based on the above assumption are shown in Figure 6.2, 6.3 and 6.4. The simulation results presented in Figure 6.5 to 6.21 compare the difference between this general stochastic approach which consider the mixed error propagation phenomena existing in a robot manipulator (curves

with black dot markers) and the approach based on the assumption that all the link parameter errors are normal distributions with zero mean (curves with triangular markers). The spikes in Figures 6.5, 6.7, 6.13 and 6.17 are due to limited computation time. The computation is very time consuming in these cases but test on the representative cases have shown that with increasing the computational accuracy the spikes tend to smooth out.

These presented results again show the importance of developing the general stochastic error propagation model to handle the arbitrarily mixed error distributions existing in the error propagation of a robot manipulator.

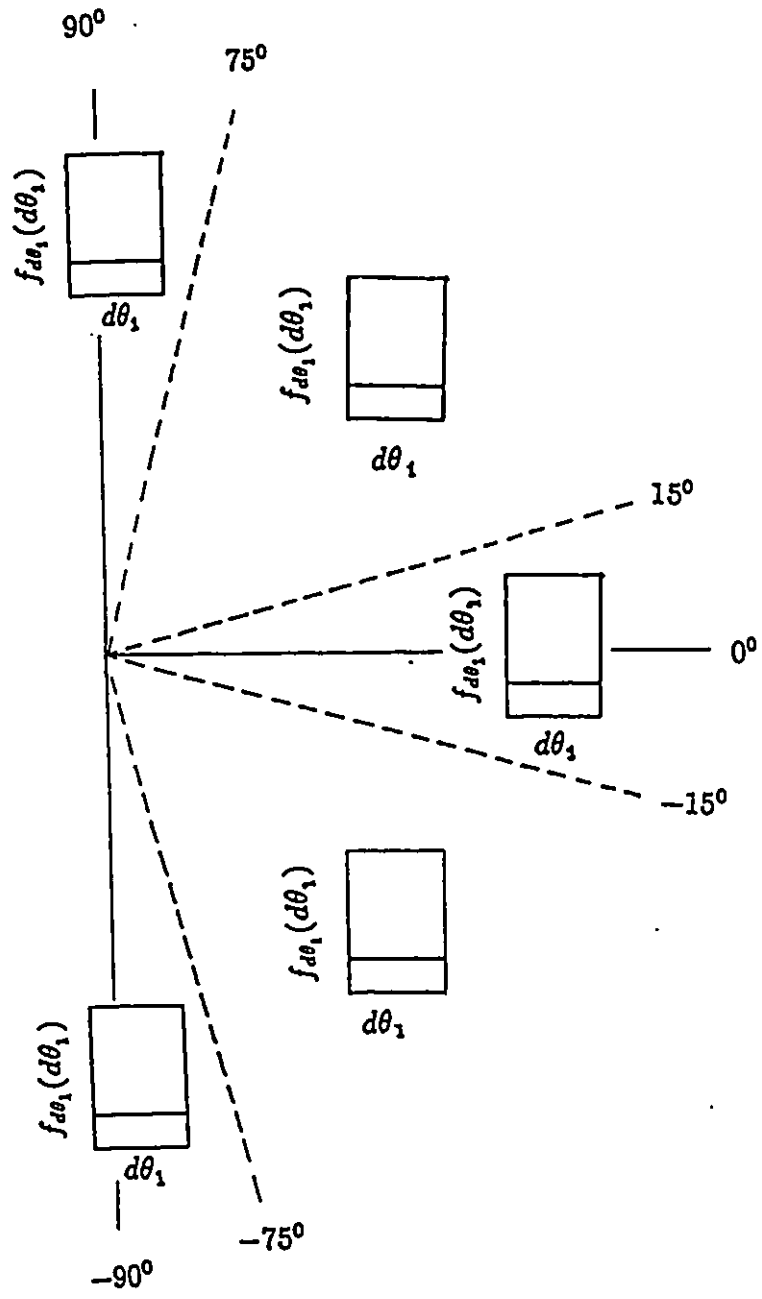


Figure 6.2: $d\theta_1$ distributions in different sub-domains of θ_1 ; $\theta_1 \in [-90^\circ, 90^\circ]$

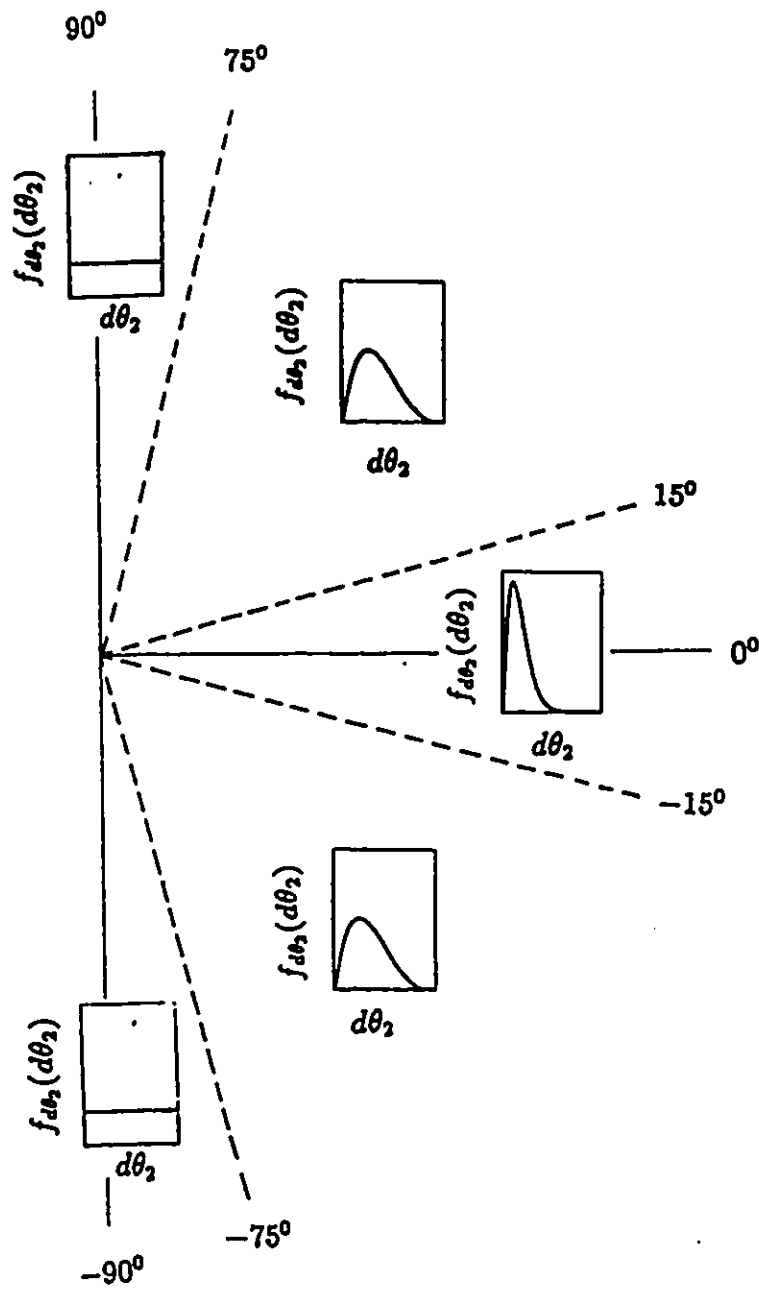


Figure 6.3: $d\theta_2$ distributions in different sub-domains of θ_2 ; $\theta_2 \in [-90^\circ, 90^\circ]$

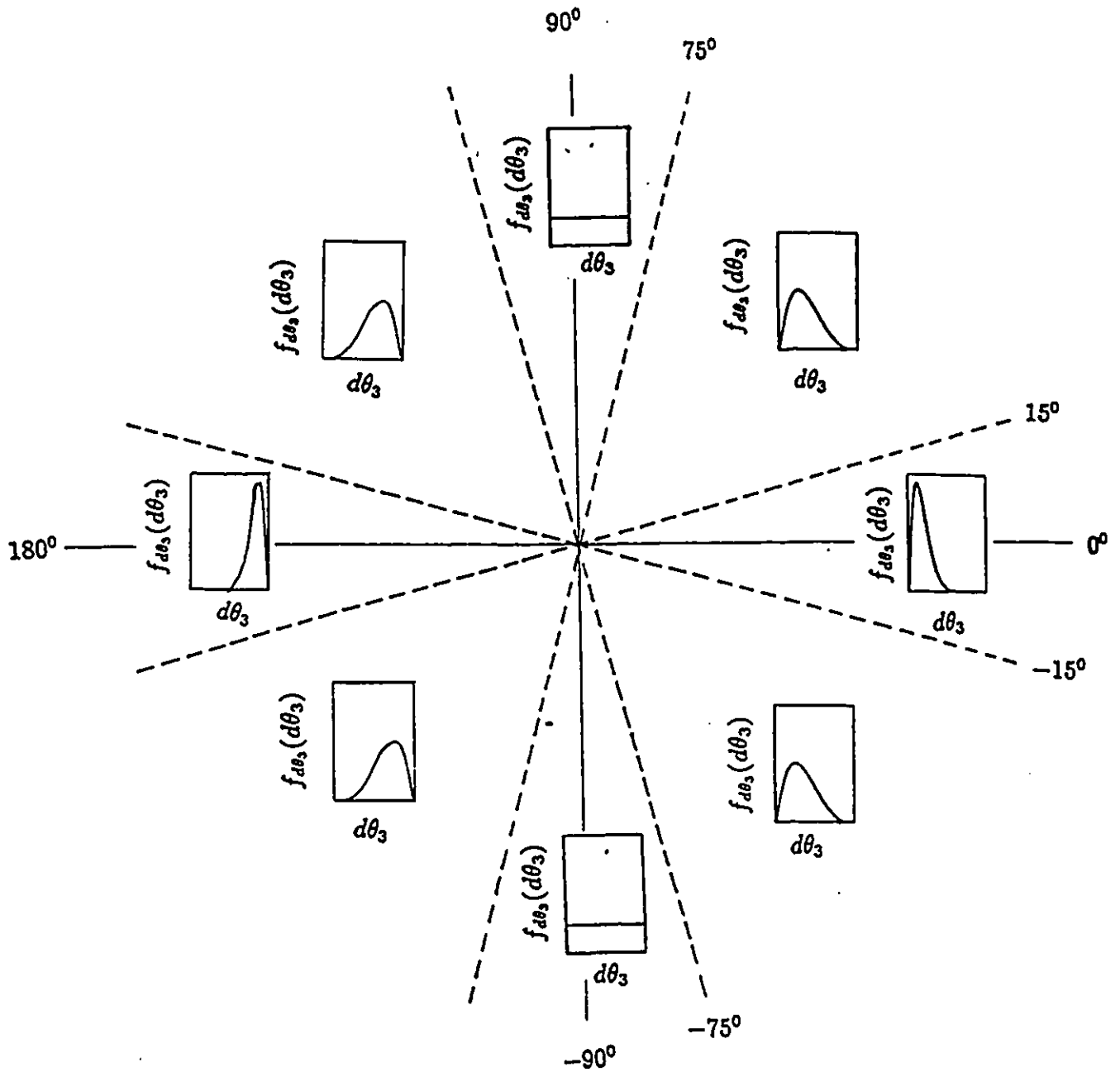


Figure 6.4: $d\theta_3$ distributions in different sub-domains of $(\theta_2 + \theta_3)$; $(\theta_2 + \theta_3) \in [-180^\circ, 180^\circ]$

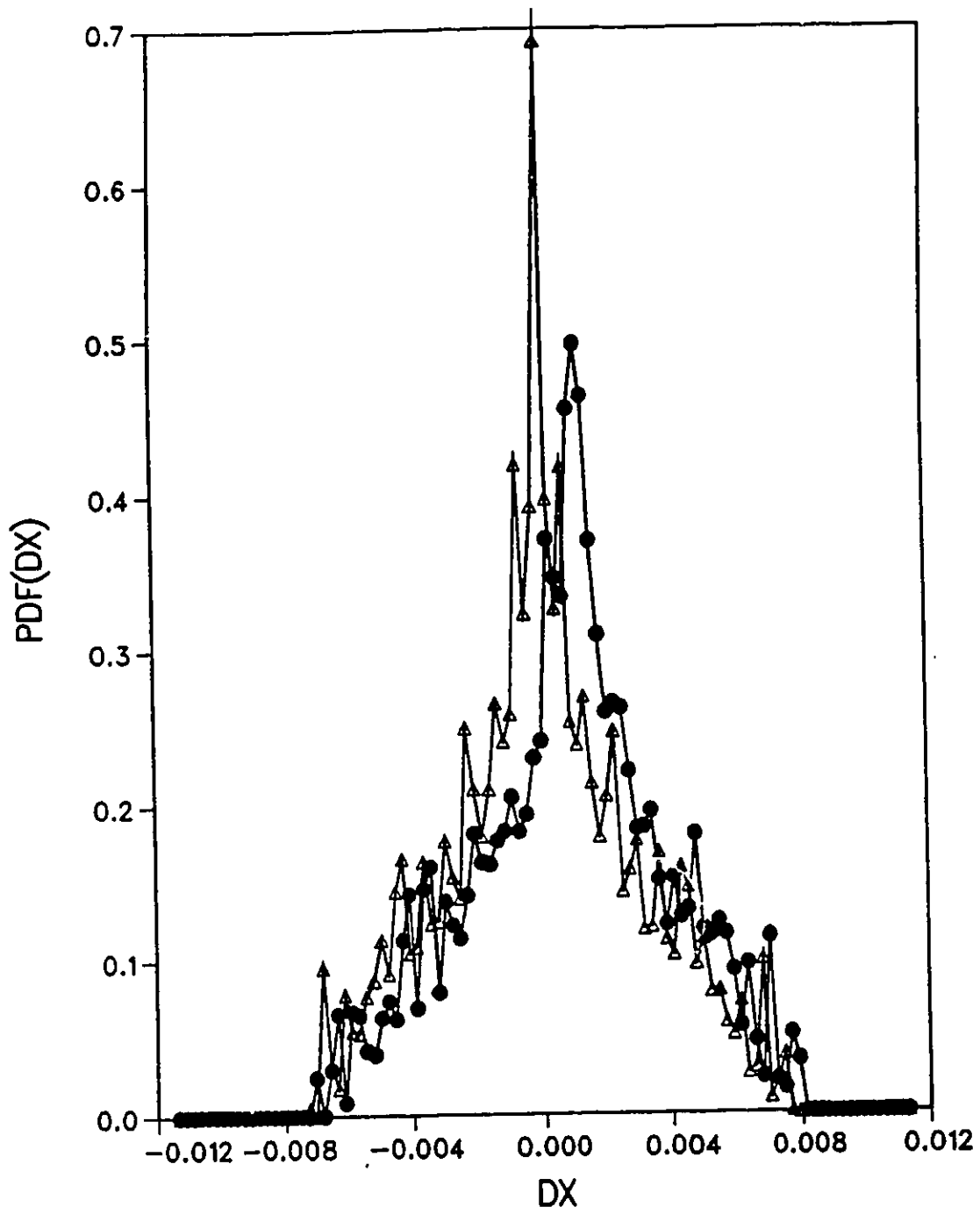


Figure 6.5: dX distribution for which $\theta_2 \in [0^\circ, 15^\circ]$ and $(\theta_2 + \theta_3) \in [0^\circ, 15^\circ]$.

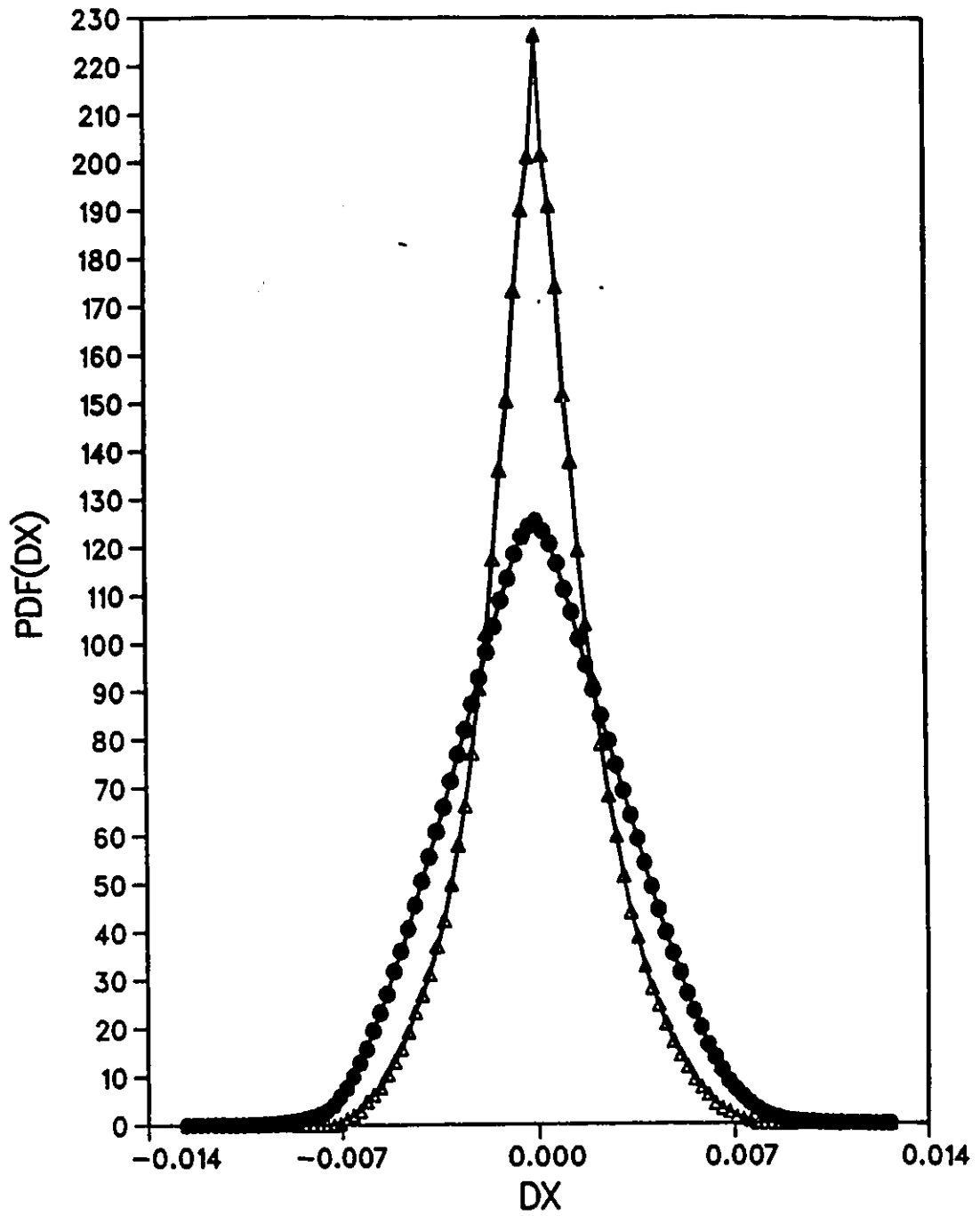


Figure 6.6: dX distribution for which $\theta_2 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^\circ, 180^\circ]$, (the whole domain of $(\theta_2 + \theta_3)$).

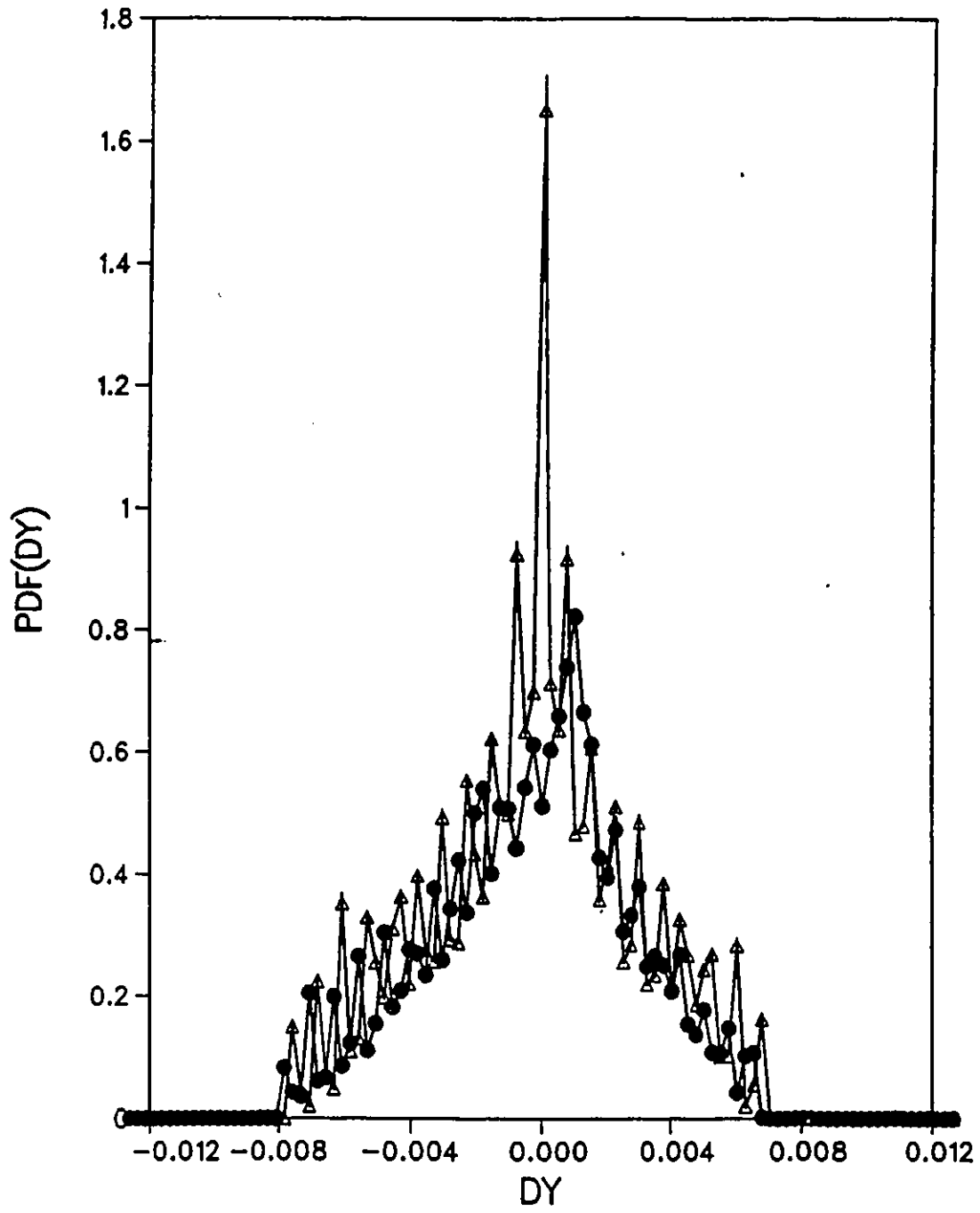


Figure 6.7: dY distribution for which $\theta_2 \in [0^\circ, 15^\circ]$ and $(\theta_2 + \theta_3) \in [0^\circ, 15^\circ]$.

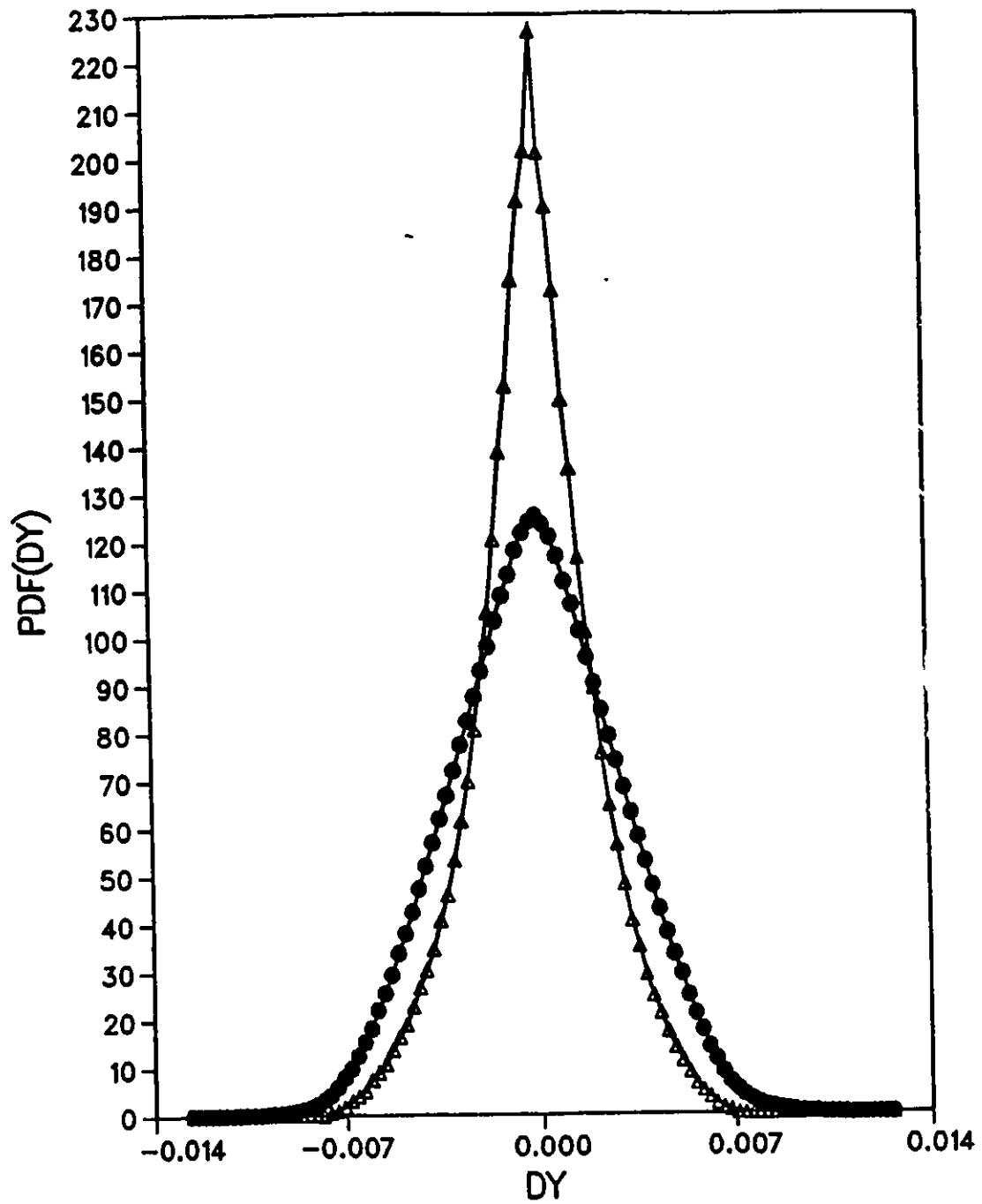


Figure 6.8: dY distribution for which $\theta_2 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_2) and $\theta_2 + \theta_3 \in [-180^\circ, 180^\circ]$, (the whole domain of $\theta_2 + \theta_3$).

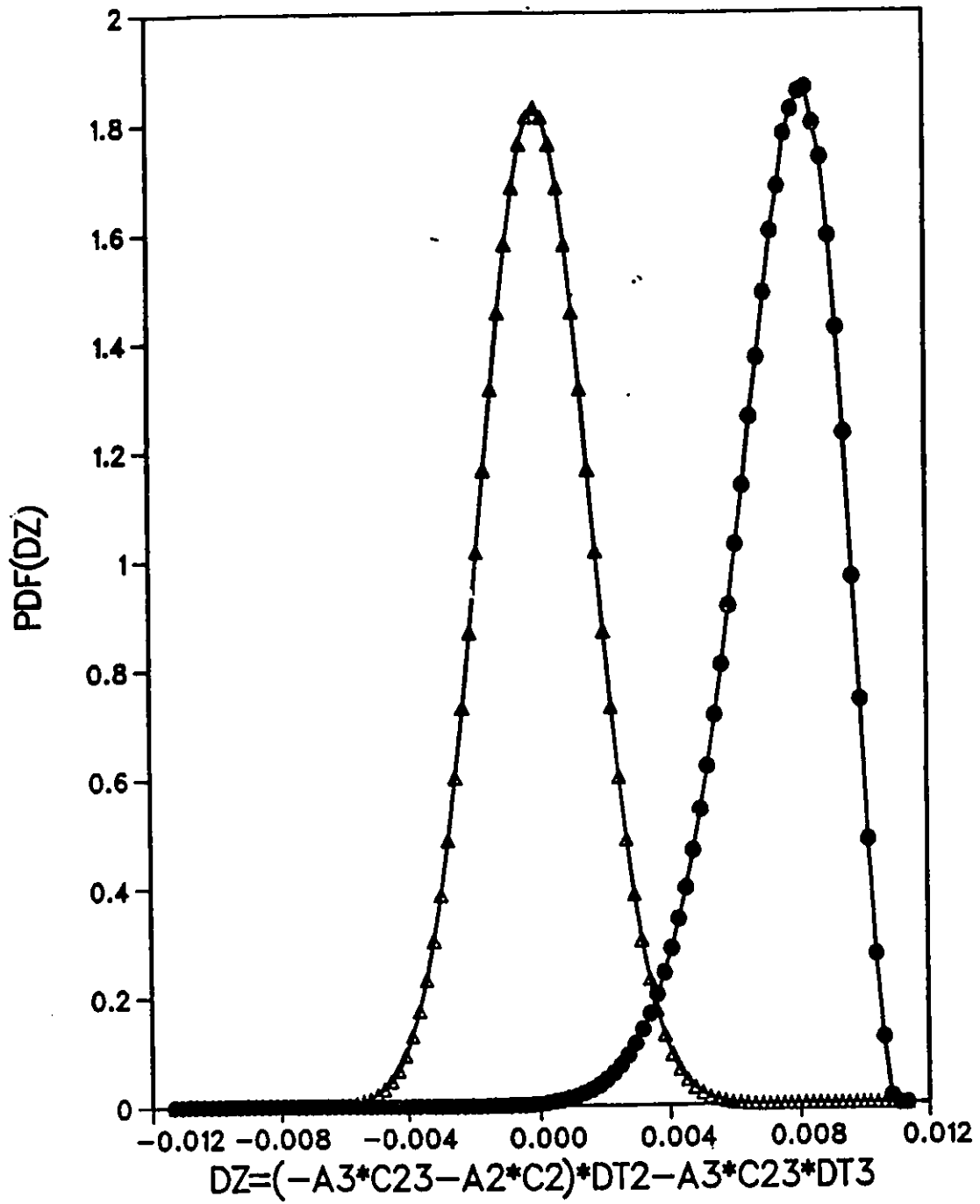


Figure 6.9: dZ distribution for which $\theta_2 \in [0^\circ, 15^\circ]$ and $(\theta_2 + \theta_3) \in [0^\circ, 15^\circ]$.

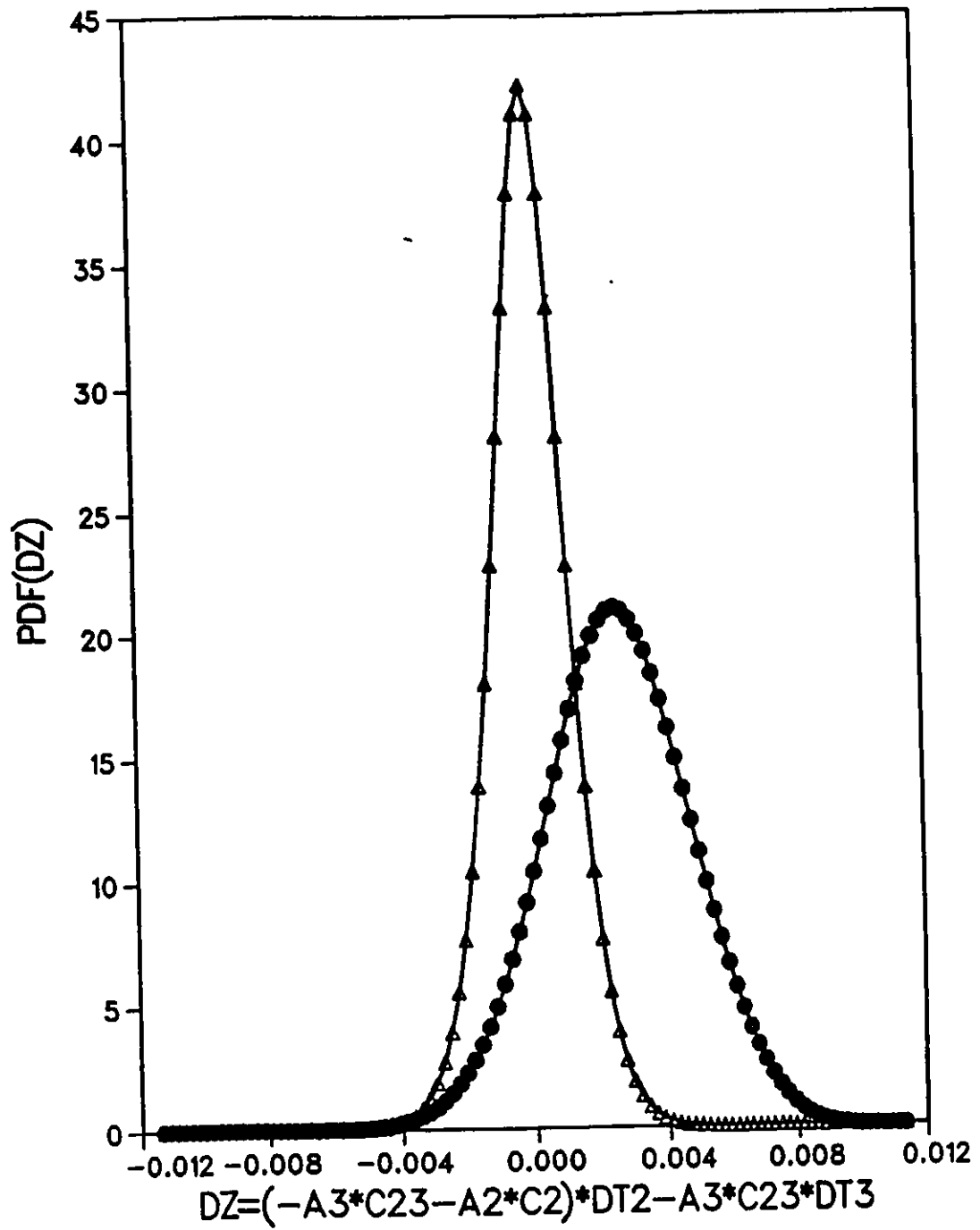


Figure 6.10: dZ distribution for which $\theta_2 \in (15^\circ, 75^\circ)$ and $(\theta_2 + \theta_3) \in (15^\circ, 75^\circ)$.

ZT3D33

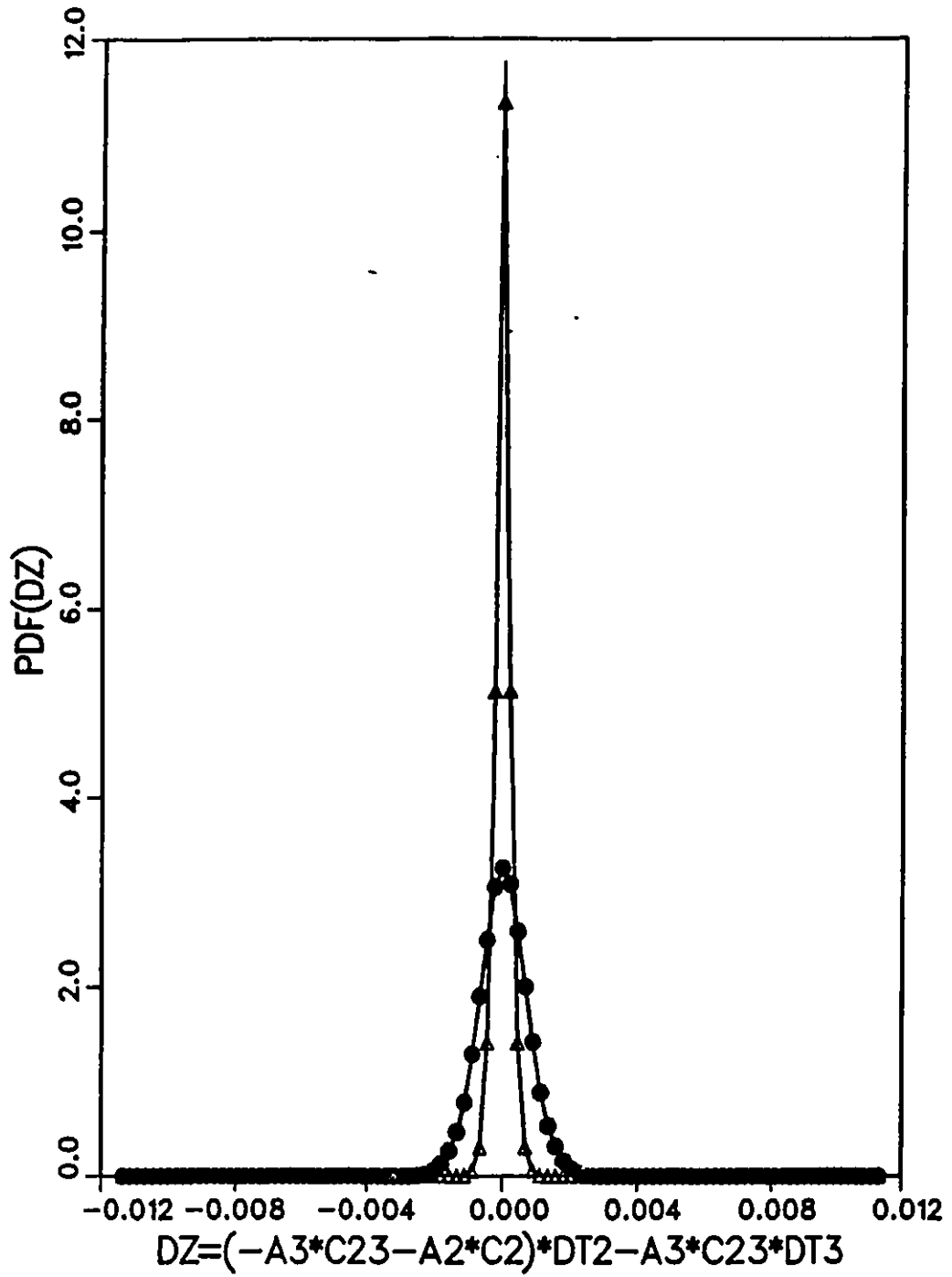


Figure 6.11: dZ distribution for which $\theta_2 \in [75^\circ, 90^\circ]$ and $(\theta_2 + \theta_3) \in [75^\circ, 90^\circ]$.

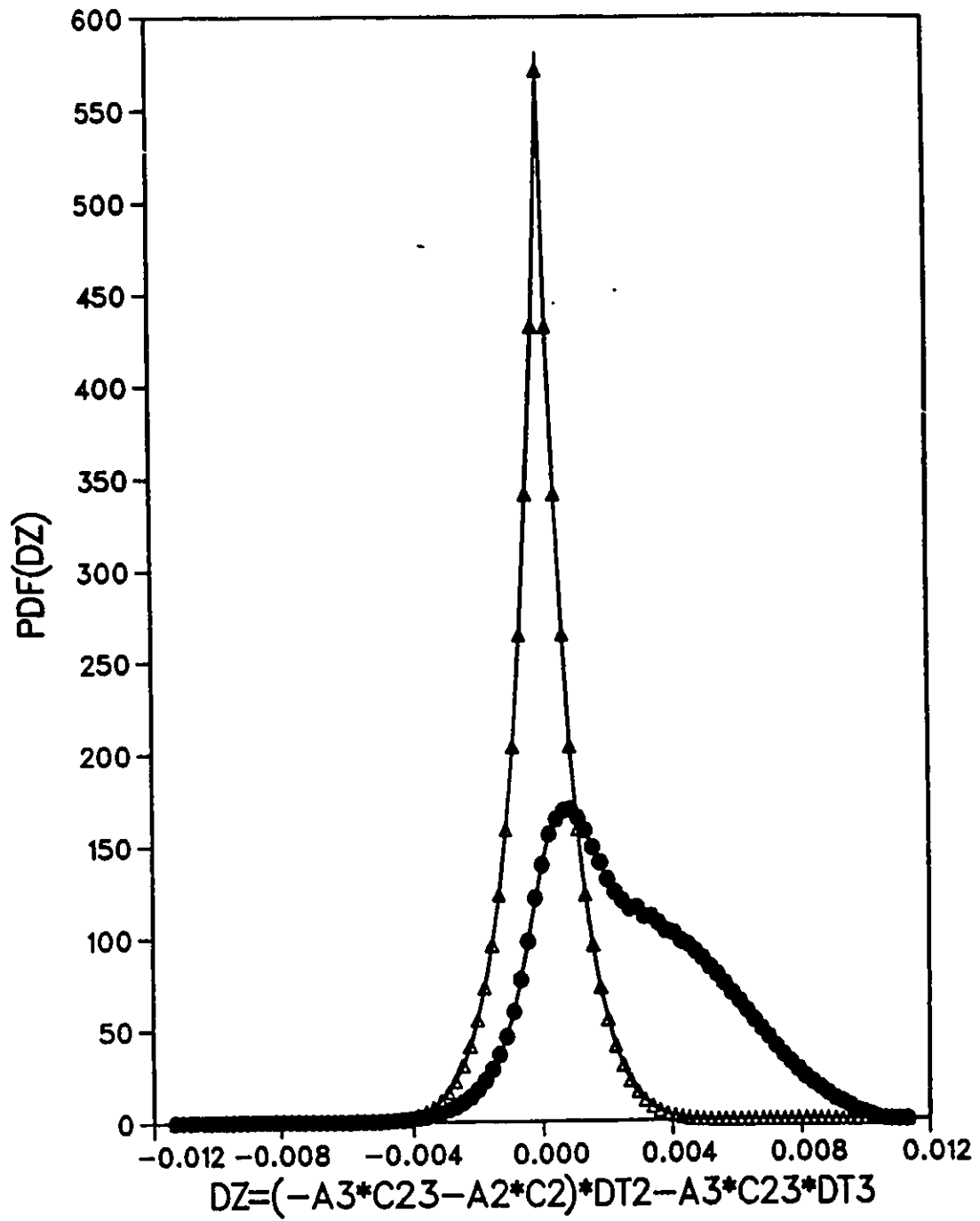


Figure 6.12: dZ distribution for which $\theta_2 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^\circ, 180^\circ]$, (the whole domain of $\theta_2 + \theta_3$).

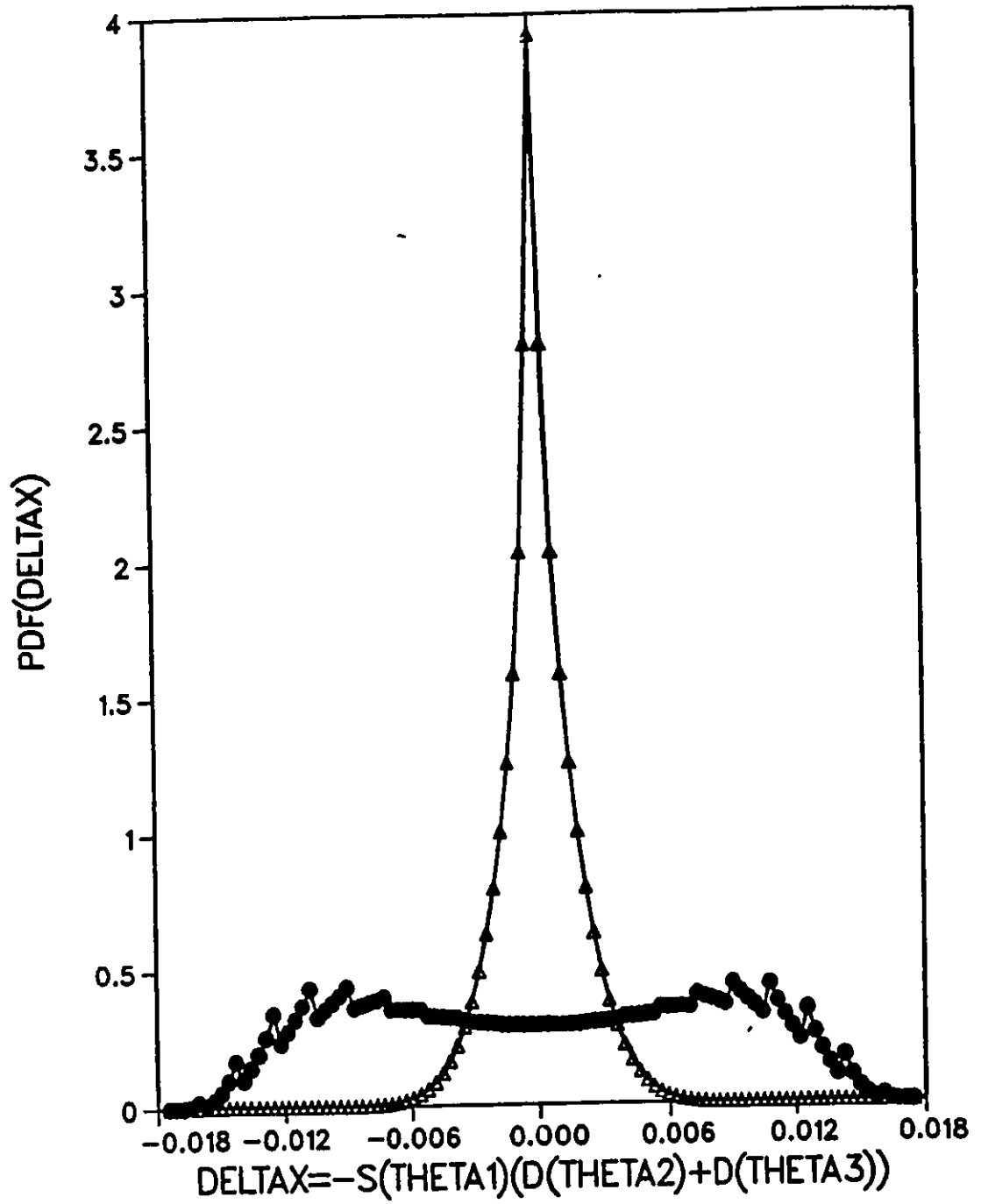


Figure 6.13: δ_x distribution for which $\theta_2 \in [0^\circ, 15^\circ]$ and $(\theta_2 + \theta_3) \in [0^\circ, 15^\circ]$.

TX3D22

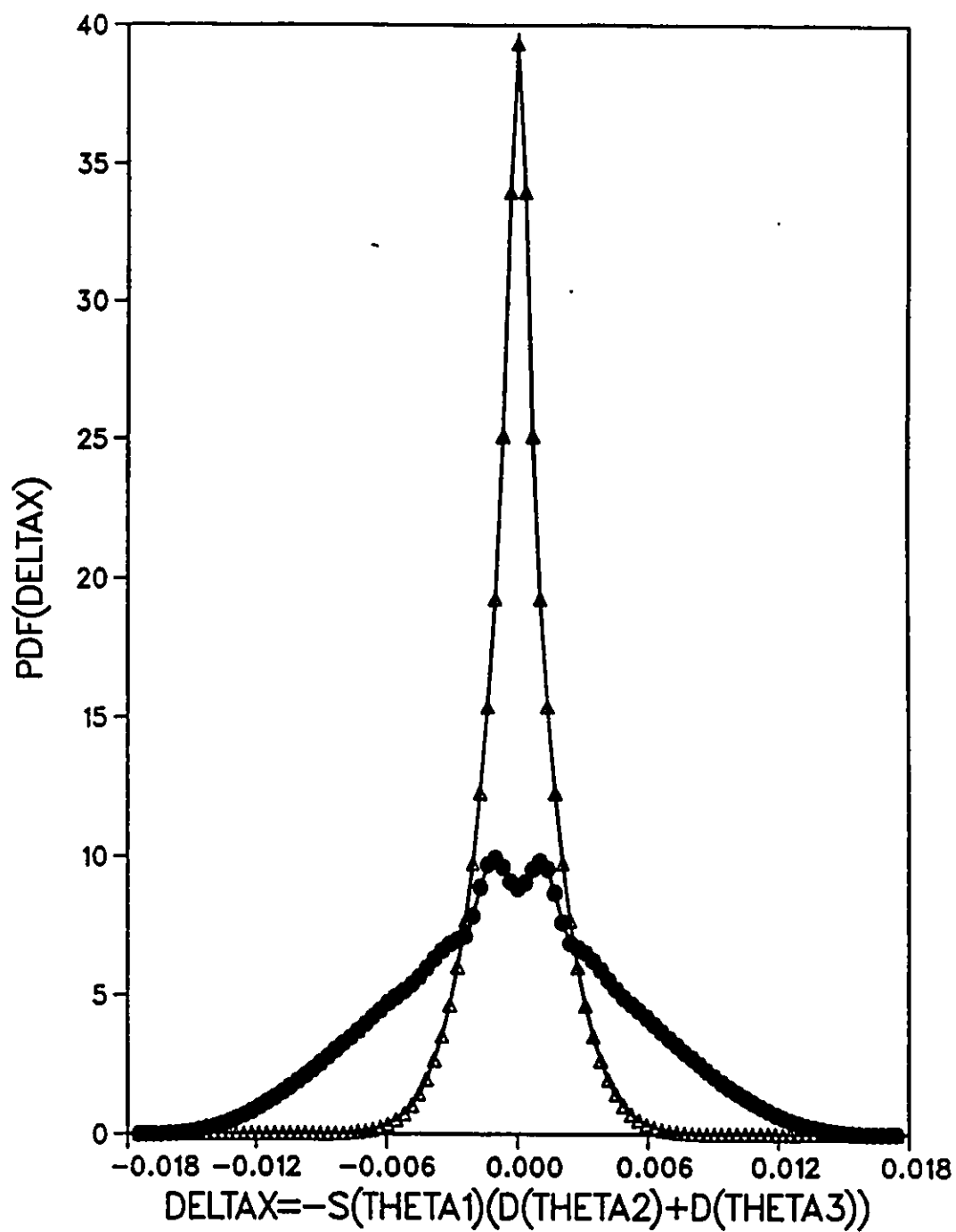


Figure 6.14: δ_X distribution for which $\theta_2 \in (15^\circ, 75^\circ)$ and $(\theta_2 + \theta_3) \in (15^\circ, 75^\circ)$.

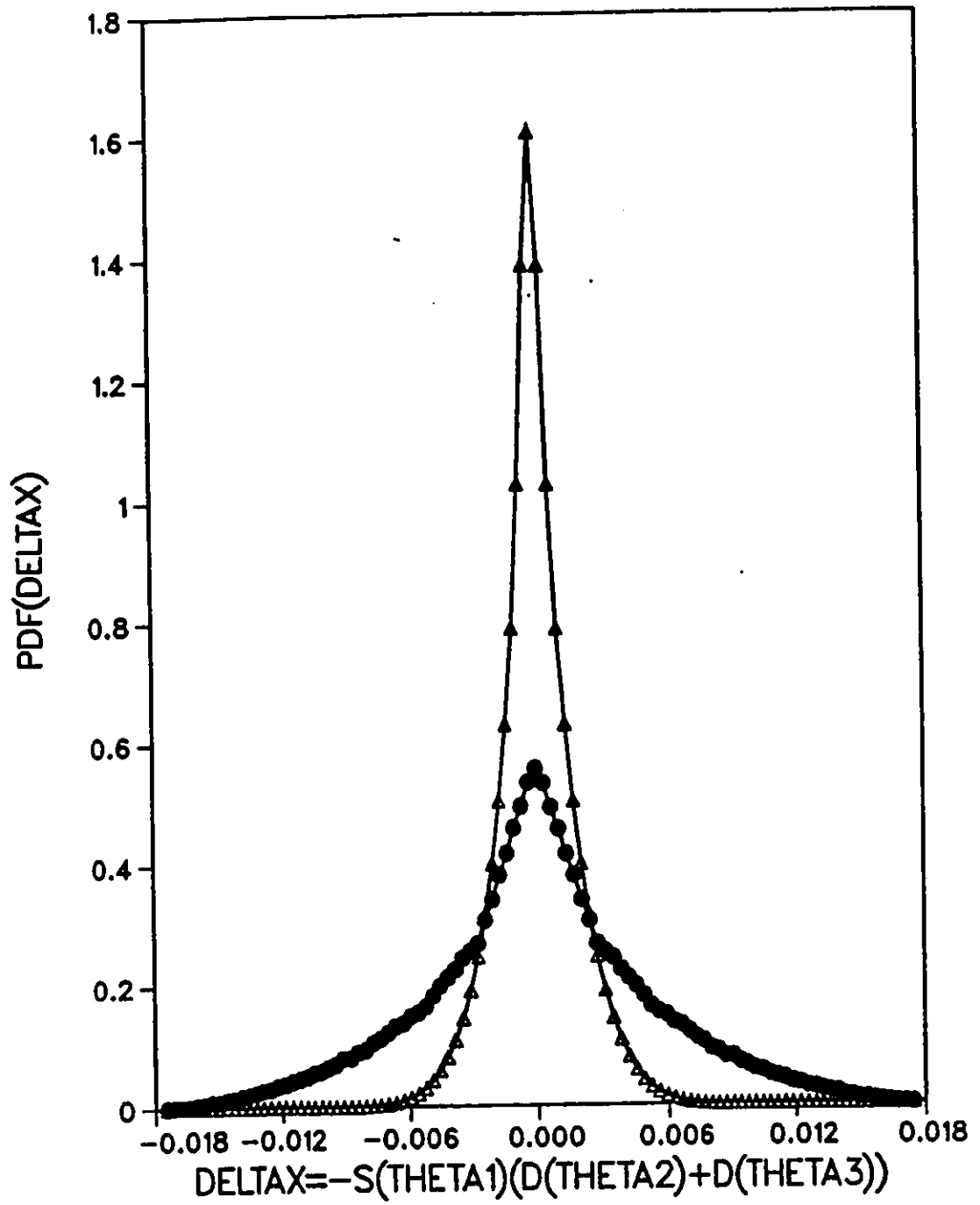


Figure 6.15: δ_x distribution for which $\theta_2 \in [75^\circ, 90^\circ]$ and $(\theta_2 + \theta_3) \in [75^\circ, 90^\circ]$.

TX3D

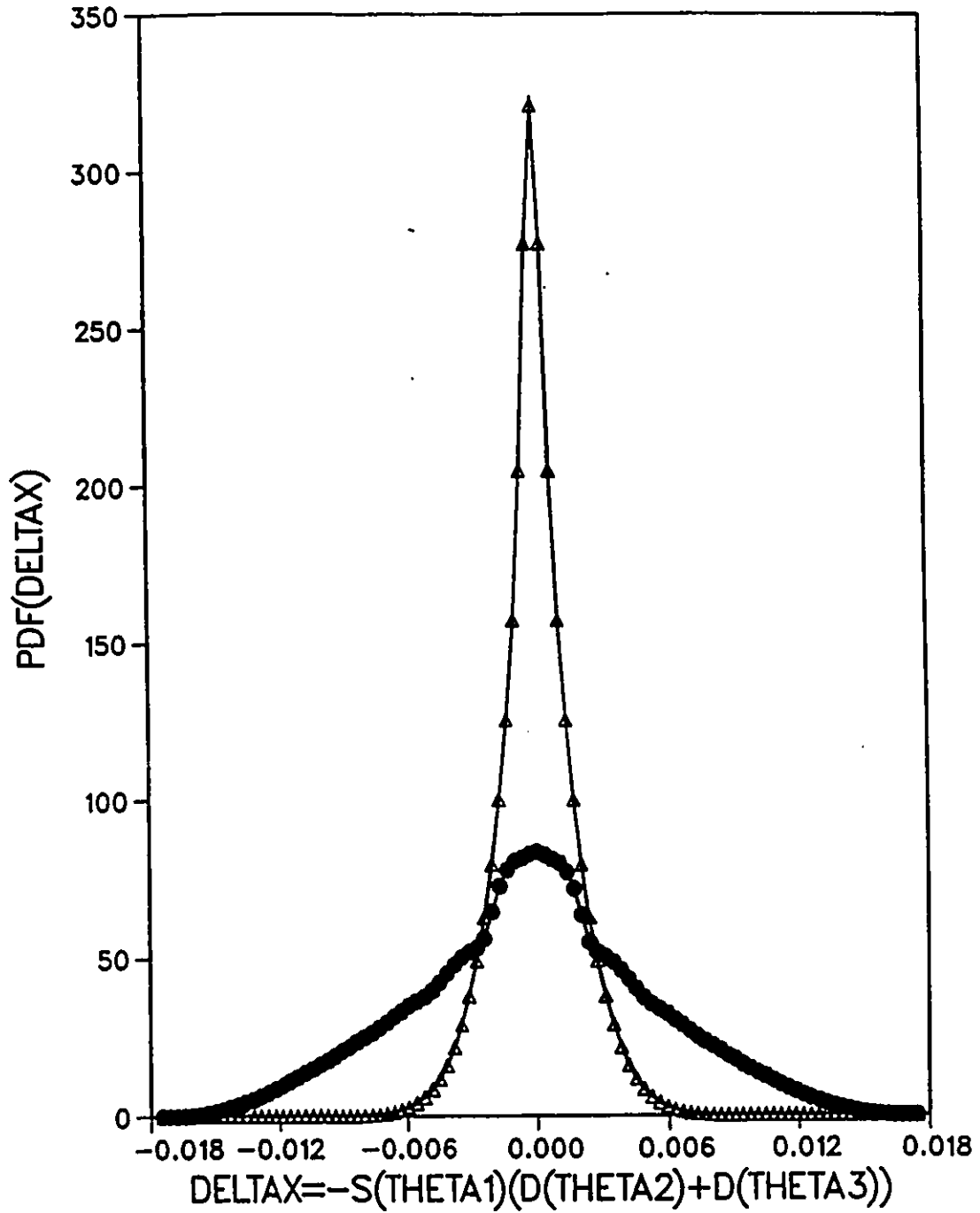


Figure 6.16: δ_X distribution for which $\theta_2 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^\circ, 180^\circ]$, (the whole domain of $\theta_2 + \theta_3$.)

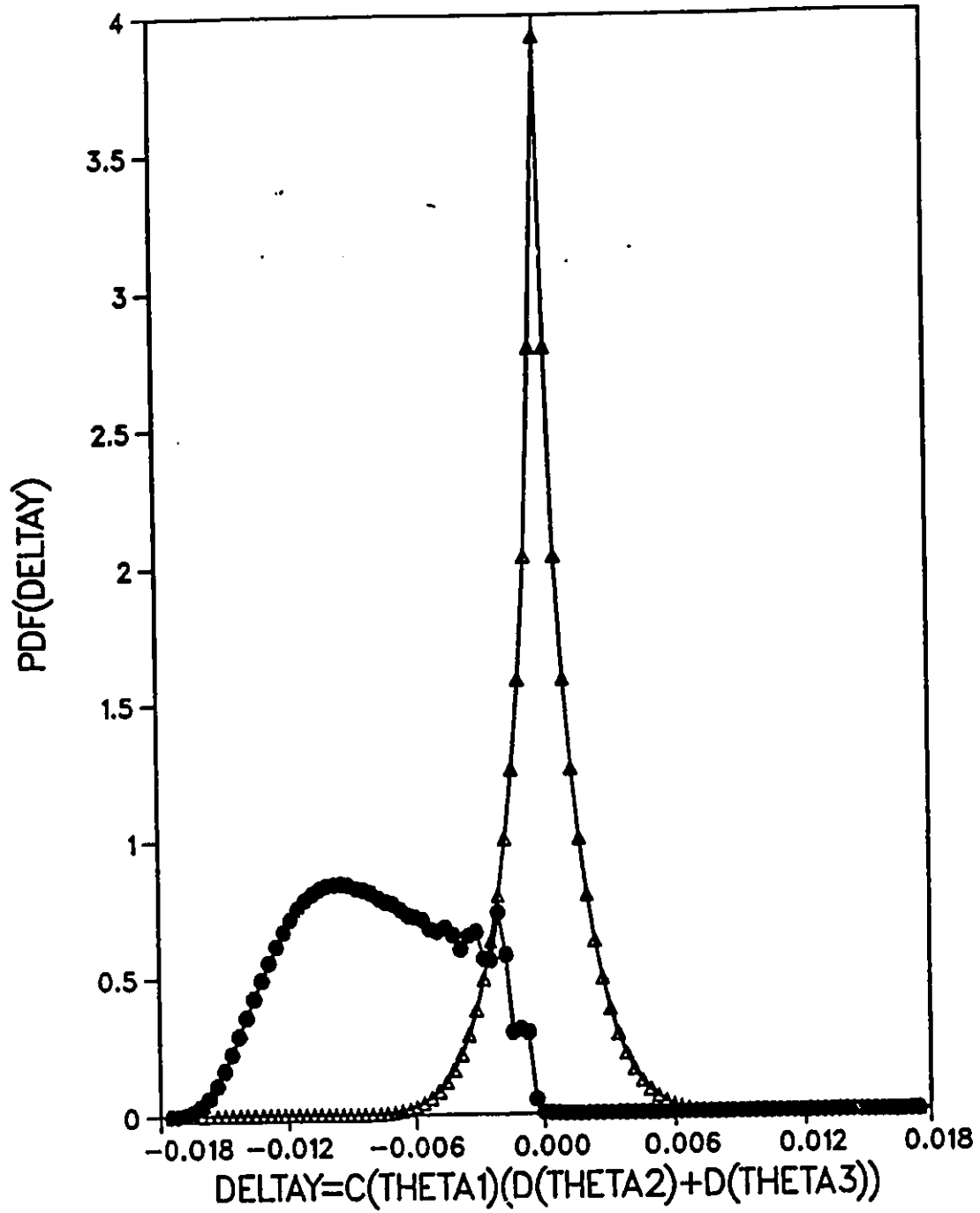


Figure 6.17: δ_γ distribution for which $\theta_2 \in [0^\circ, 15^\circ]$ and $(\theta_2 + \theta_3) \in [0^\circ, 15^\circ]$.

TY3D22

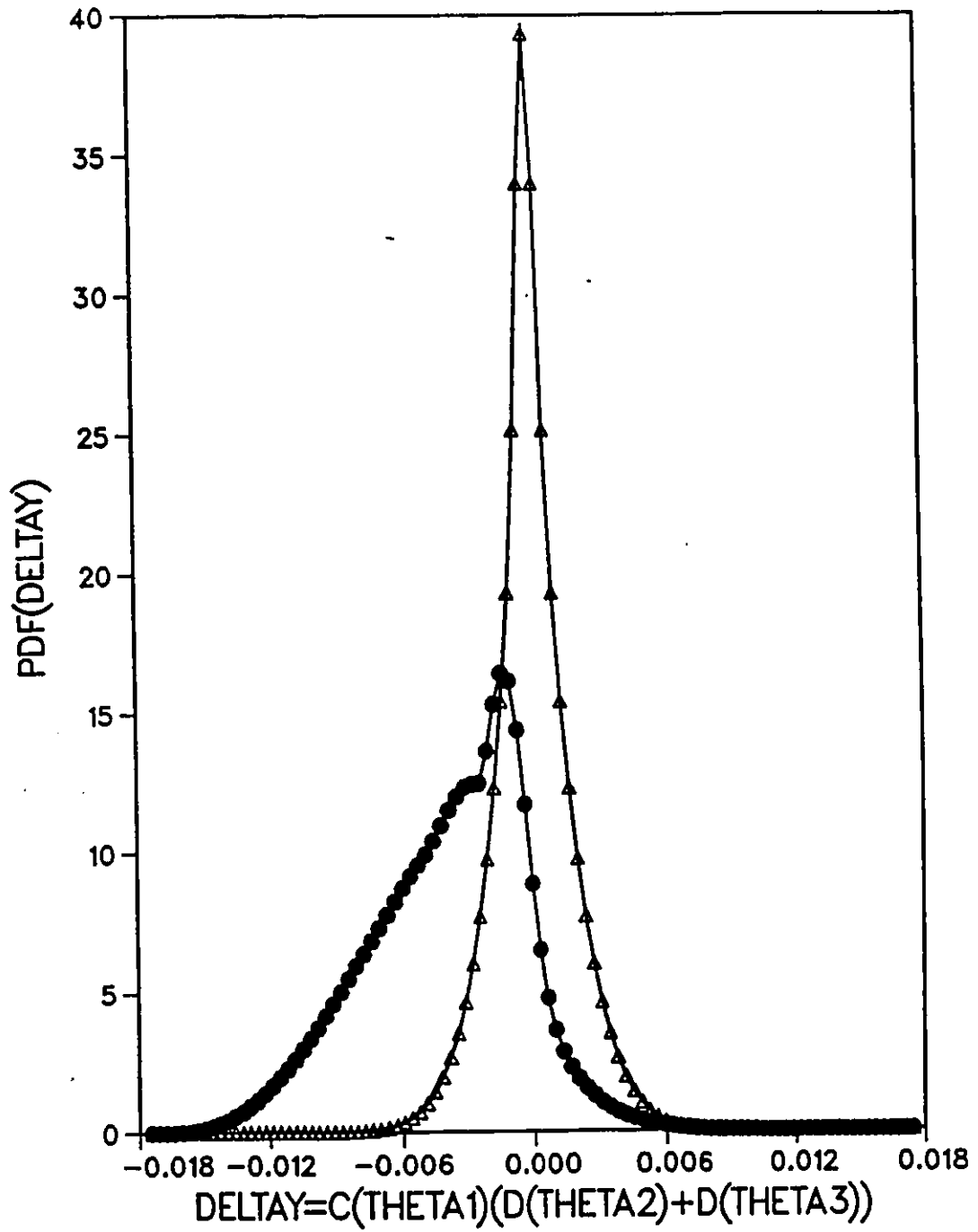


Figure 6.18: $\delta\gamma$ distribution for which $\theta_2 \in (15^\circ, 75^\circ)$ and $(\theta_2 + \theta_3) \in (15^\circ, 75^\circ)$.

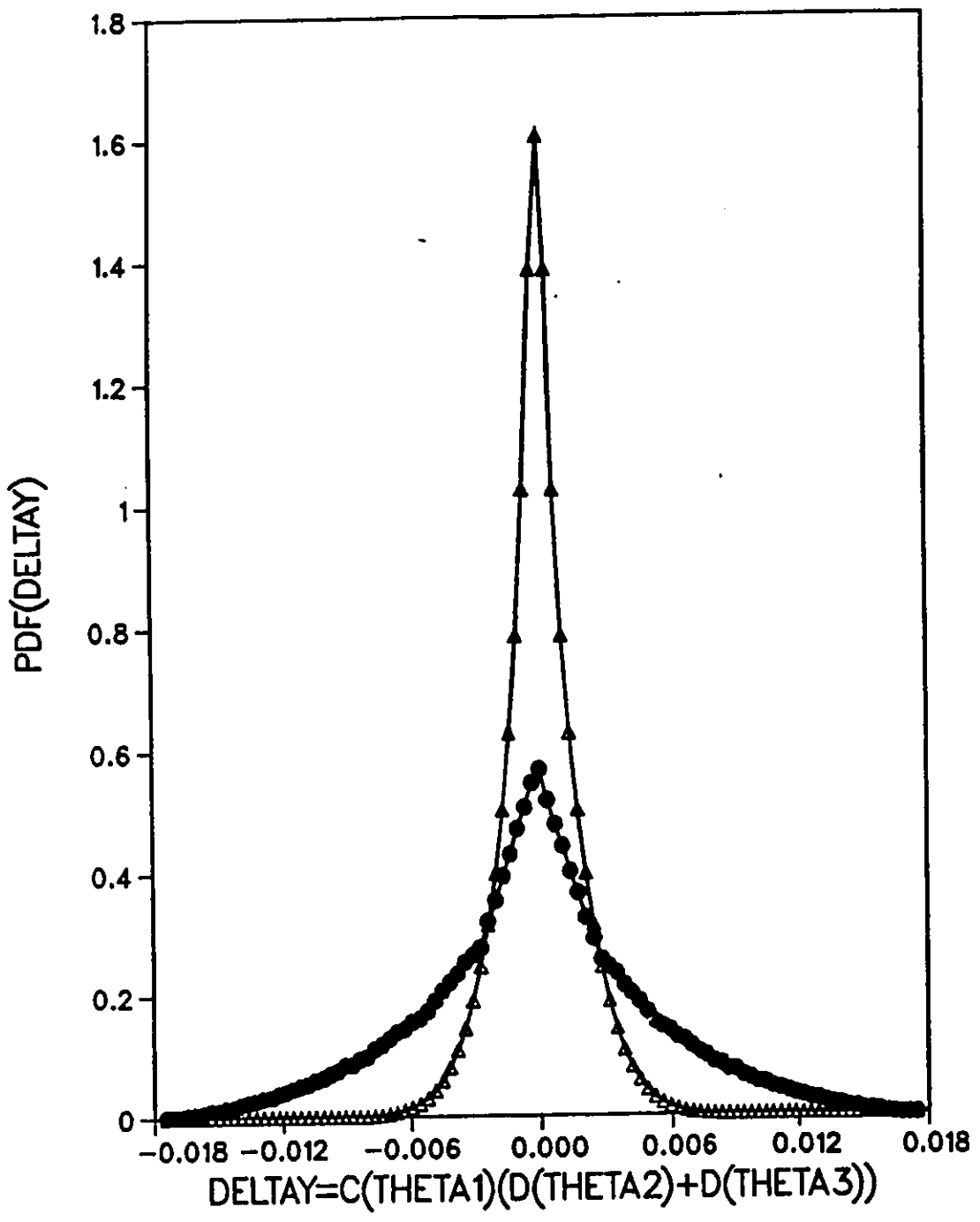


Figure 6.19: δ_Y distribution for which $\theta_2 \in [75^\circ, 90^\circ]$ and $(\theta_2 + \theta_3) \in [75^\circ, 90^\circ]$.

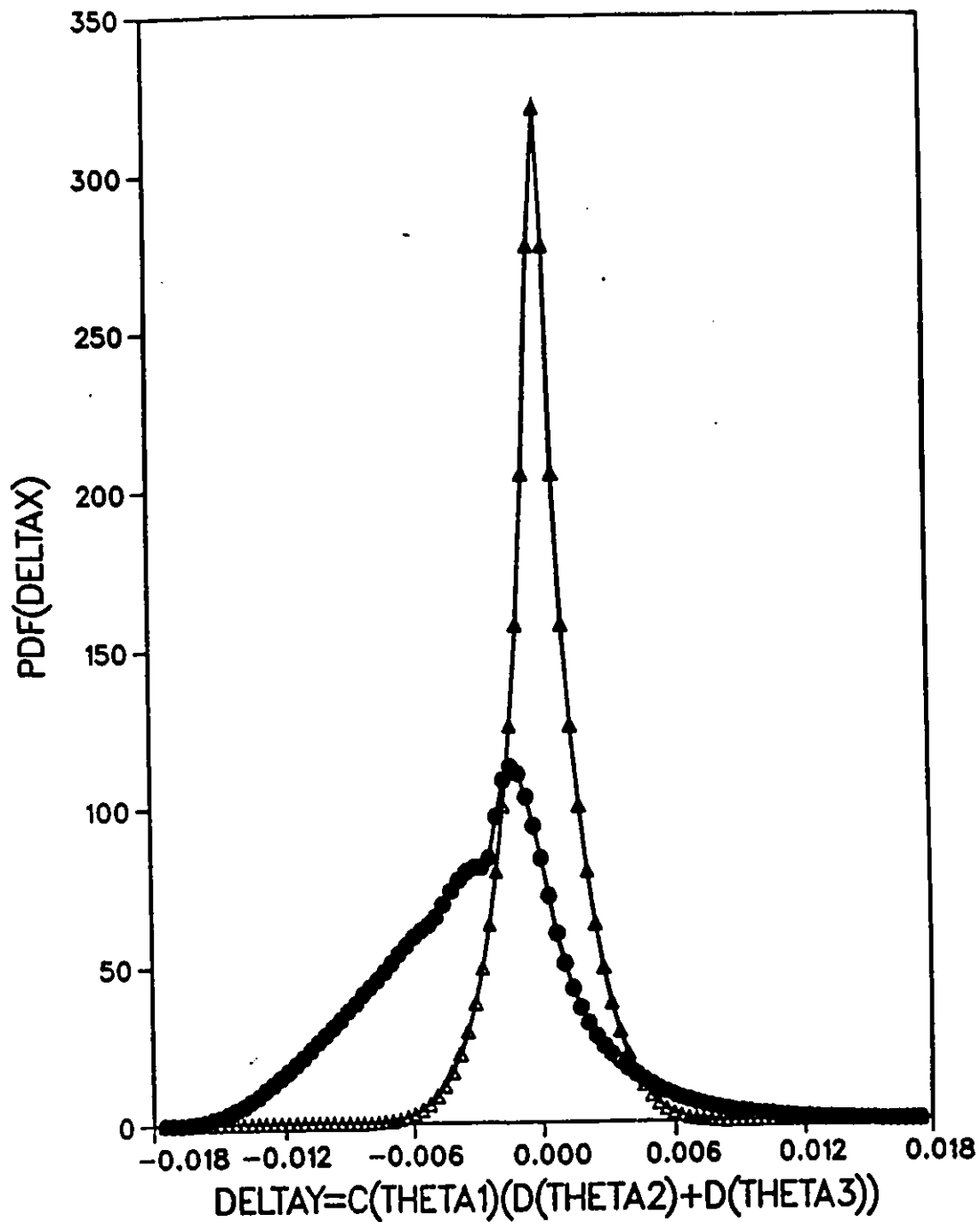


Figure 6.20: δy distribution for which $\theta_2 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^\circ, 180^\circ]$, (the whole domain of $(\theta_2 + \theta_3)$.)

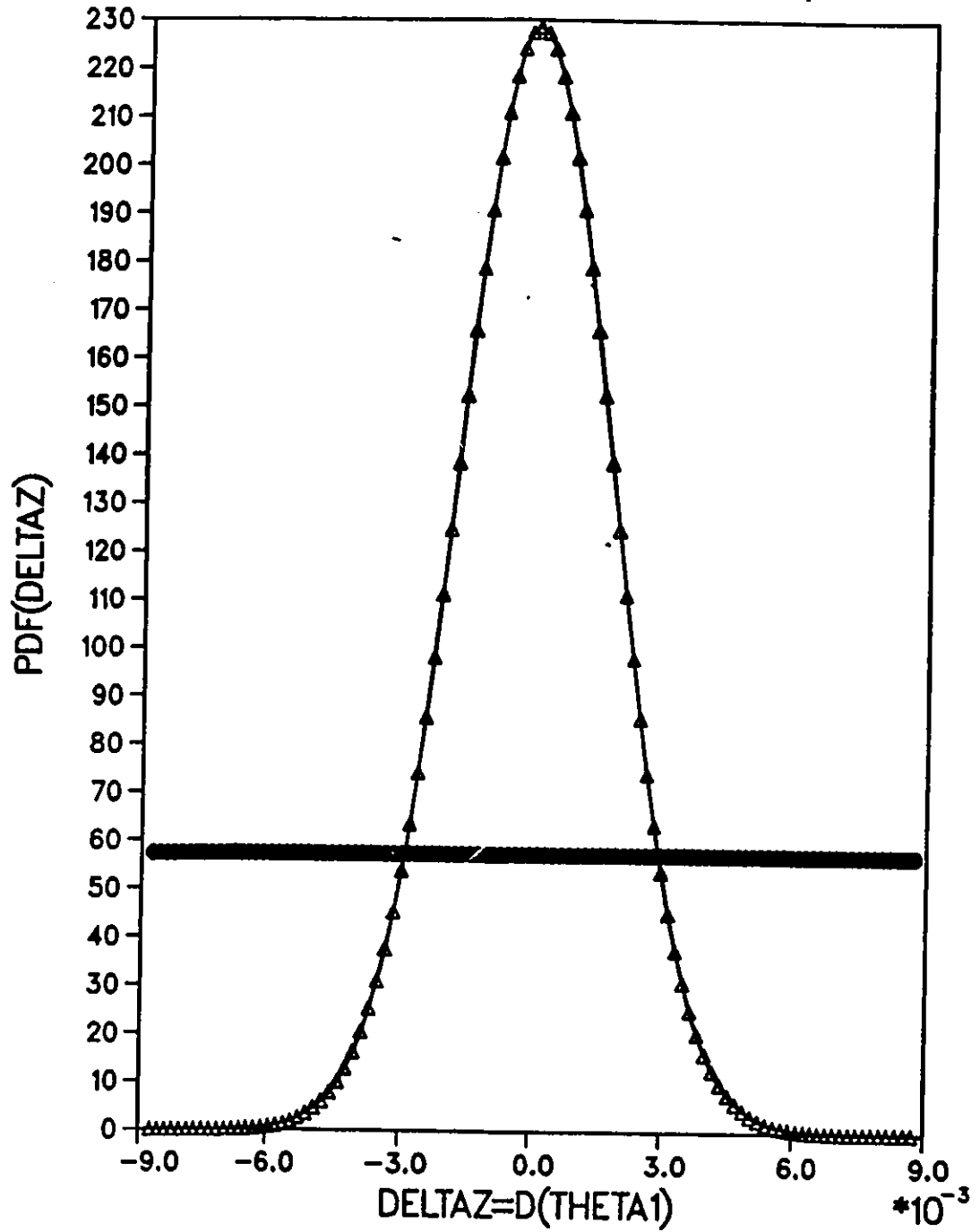


Figure 6.21: δ_z distribution for which $\theta_2 \in [-90^\circ, 90^\circ]$, (the whole domain of θ_2) and $(\theta_2 + \theta_3) \in [-180^\circ, 180^\circ]$, (the whole domain of $(\theta_2 + \theta_3)$.)

In summary, detailed numerical simulations for a spatial three degrees-of-freedom robot manipulator were performed in this chapter and some comments can be made here.

- A three degrees-of-freedom spatial robot manipulator is the maximum extent to which the simulation in this thesis were performed.
- All previous features, mentioned in the presentation of one and two degrees-of-freedom simulations were also found here.
- The three degrees-of-freedom robot simulations represent a realistic result for the positioning and orientation errors of a robot manipulator, although in some cases extensive computational time are needed.
- The sequence of simulations from one to two and three degrees-of-freedom robot manipulator justify our confidence that the presented model is applicable to any higher degrees-of-freedom robot manipulators on the expense of extensive computation.

Chapter 7

Discussion and Conclusion

An analytical model for predicting the positioning or orientational error distribution functions of an end-effector resulting from the arbitrarily mixed error propagation of the 5N link parameter errors is presented in this dissertation. The model is mathematically rigorous and applicable to any degrees-of-freedom non-redundant robot manipulator. This model is computationally intensive in order to obtain the desired positioning or orientational error distribution functions.

Numerical difficulty was experienced in the evaluation of the positioning or orientational error distribution function when the number of the supplementary output random variables are higher than 5. In principle, all the link parameter error sources in the numerical simulation for the general stochastic error propagation model can be included in the numerical computations on the expense of a very large computational burden. In this thesis the numerical simulations are given for up to three degrees-of-freedom in order to prove the validity and the features of the proposed model.

Although the presented model has such a computational limitation, it can still be applied well for real world applications. The other practical reasons for limiting the number of the error sources taken into account in numerical simulations are :

- For each link, four of the five link parameters associated with the link are geometric link parameters. For this kind of parameters, it is possible to use calibration accurately. This fact simplifies the presented model to be composed of only joint parameter values and joint parameter errors.

- In real world applications, the robot manipulator is commanded to follow a pre-designed trajectory to perform some specific or iterative tasks. In this case, the input joint parameters have to take some specific values rather than be random variables. This fact makes the presented model be able to be further simplified to a function of input joint parameter errors only.

Basing on the afore mentioned facts, for a six degrees-of-freedom robot manipulator, in some real world applications the positioning or orientational error distribution functions defined by equation (3.34) can be represented as functions of the six input joint parameter errors only. Consequently, in order to find the positioning or orientational error distribution functions of a six degrees-of-freedom robot manipulator, only five supplementary output random parameters are needed and this is still under the computational limit of the presented model.

In investigating the output positioning and orientational error distributions of the end-effector of a robot manipulator for the specific cases discussed in chapters 4, 5 and 6, the robot workspace is divided into some independent smaller sub-domains such that the joint probability function of the random input variables can be fully expanded as a product of probability density functions for each input variables. Then the output positioning or orientational error distributions are computed by the marginal distribution method given in chapter 3. Error propagation programs based on the marginal distribution method were developed to plot the distribution functions of the positioning or orientational errors of the robot manipulators. Practically, the output positioning or orientational errors due to the 5N link parameter errors of a robot manipulator depend not only on the magnitude of the 5N link parameter errors but also on the robot configuration. All the simulation results presented in chapters 4, 5, and 6 show the importance of developing this general stochastic error propagation model to analyze the mixed error propagation phenomena existing in the structure of a robot manipulator.

In summary, this dissertation has developed a rigorous mathematical method for analyzing the positioning or orientational error distributions of the end-effector resulting from the combined effects of the mixed error distributions existing in the link parameter errors of a robot manipulator.

First, in chapter 2 a general error propagation model was developed. It describes the positioning and orientational errors of the end-effector with respect to the base

frame as a non-linear function of the five link parameter errors for each link.

To evaluate the positioning or orientational error distribution functions of the end-effector of a robot manipulator, in chapter 3 a general stochastic model was developed. The proposed model is based on the joint probabilities and marginal distribution calculation. The model is mathematically rigorous and general enough to analyze any kind of mixed error propagations for any robot manipulators.

In order to illustrate the important characteristic features of the presented stochastic model, in chapter 4 detailed error propagation and stochastic models for the positioning and orientational errors of a one degree-of-freedom robot were presented. The simulation results show that it is very important to consider the mixed error propagation phenomena existing in a one degree-of-freedom robot manipulator.

In chapter 5, a further illustration of the presented general stochastic approach was given for a two degrees-of-freedom vertical planar robot manipulator. The simulation results show the same important information as chapter 4. Namely, it is very important to use the presented model to investigate the possible mixed error propagation phase of a robot manipulator.

In order to prove that the presented stochastic model is able to analyze the error propagation for any degrees-of-freedom spatial robot manipulator, in chapter 6 detailed error propagation and stochastic models were derived. The simulation based on the derived models were thoroughly investigated. The results show the same features as shown in chapters 4 and 5. Consequently, it is very important and more realistic to apply the presented general stochastic model to analyze the arbitrarily mixed error propagations of a robot manipulator rather than other models based on the propagation of normally distributed errors.

As mentioned in the comments of chapter 6, three degrees-of-freedom was the maximum extent to which the simulation in this thesis were performed. From the presented results, we completely justify that the presented model can be applied to real world applications to improve the positioning and orientational accuracy of a robot manipulator.

Main contributions were made in the following area :

- A new and powerful general stochastic error propagation model of arbitrary structural robot manipulators was developed. The resulting general stochastic model use transcendental functions and is exact in that it involves no approximation beyond those made by the governing probability models.
- The joint probability techniques and decomposition to smaller sub-domains might be extended to any kind of robot manipulators.
- Quantative models of error propagation analysis may be further applied to the error distribution analysis along the desired trajectory.

In the field of robot trajectory control, this thesis has done little more than introduce a basic idea. Many extensions are possible, and I intend to pursue this aspect myself. In particular, one might develop an optimal trajectory based on the error distribution function obtained in this dissertation.

References

1. Farhad Azadivar, *The Effect of Joint Position Errors of Industrial Robots on Their Performance in Manufacturing Operations*, IEEE J. of Robotics And Automation, Vol. Ra-3, No.2, April 1987.
2. Shaheen Ahmad, *Analysis of Robot Drive Train Errors, Their Static Effects, and Their Compensations*, IEEE J. of Robotics and Automation, Vol. 4, No. 2, April 1988.
3. B. Benhabib, R. G. Fenton and A. A. Goldenberg, *Computer-Aided Joint Error Analysis of Robots*, IEEE J. of Robotics and Automation, Vol. Ra-4, No. 4, August 1987.
4. Jigien Chen and Lih-Ming Chao, *Positioning Error Analysis for Robot Manipulators with All Rotary Joints*, IEEE J. of Robotics and Automation, Vol. Ra-3, No. 6, December 1987.
5. J. Chen, C. B. Wang and J. C. S. Yang, *Robot Positioning Accuracy Improvement Through Kinematic Parameter Identification*, in Proc. Third Canadian CAD/CAM and Robotics Conf., Tronto, Canada, June 1984.
6. J. Denavit and R. S. Hartenberg, *em A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices*, ASME J. of Applied Mechanics, pp. 215-221, June 1955.
7. S. G. Dhande and J. Chakraborty, *Analysis and Synthesis of Mechanical Error in Linkages - A Stochastic Approach*, J. of Engineering for Industry, Transaction of ASME, Vol. 95, pp. 672-676, August 1973.
8. S. Dubowsky, J. Maatuk and N. D. Perreira, *A Parameter Identification Study of Kinematic Errors in Planar Mechanisms*, J. of Engineering for Industry,

pp. 635-642, May 1975.

9. Laurent P. Foulloy and Robert B. Kelley, *Improving the Precision of a Robot*, International Conf. on Robotics, Atlanta, Georgia, USA, March 13-15, 1984.
10. R. E. Garret and Allen S. Hall, Jr., *Effect of Tolerance and Clearance in Linkage Design*, J. of Engineering for Industry, Transaction of ASME, Series B, Vol. 91, No. 1, February 1969, pp. 198-202.
11. G. Hadley, *Nonlinear and Dynamic Programming*, Addison-Wesley, Reading, 1964, pp. 350-413.
12. Samad A. Hayati, *Robot Arm Geometric Link Parameter Estimation*, in Proc. 22nd IEEE Conf. Decision and Control, Dec. 1983, pp. 1477-1483.
13. S. Hayati and M. Mirmirani, *Improving the Absolute Positioning Accuracy of Robot Manipulators*, J. of Robotics Systems, 1985, pp. 397-413.
14. C. O. Huey and Joseph Anand, *Mapping of Positioning Error of Robot Manipulators*, Proceedings of An International Conference, Charlotte, North Carolina, USA, December 1984.
15. A. Kumar, *Positioning Accuracy of Manipulators with Encoder Equipped Joints*, The Winter Annual Meeting of ASME, Phoenix, Arizona, November 14-19, 1982.
16. A. Kumar and K. J. Waldron, *Numerical Plotting of Surfaces of Positioning Accuracy of Manipulators*, Mechanism Mach. Theory, Vol. 16, No. 4, pp. 361-368, 1981.
17. G. A. Korn and T. M. Korn, *Mathematical Handbook for Scientists and Engineers*, Mcgraw Hill, New York, 1968.

18. C. S. G. Lee and M. Ziegler, *A Geometric Approach in Solving the Inverse Kinematics of Puma Robots*, Robot System Division, U. of Michigan, Dec. 1983.
19. Paul L. Meyer, *Introduction Probability And Statistical Applications*, Addison-Wesley, Reading.
20. B. W. Mooring, *The Effect of Joint Axis Misalignment on Robot Positioning Accuracy*, The Proceedings of ASME Conf. on Computers in Engineering, Vol. 2, pp. 151-155, August 7-11, 1983.
21. D. S. Neculescu, A. Fahim and M. Fortin, *Robot Positioning and Orientation Accuracy*, Optimization Dnys, Montreal, May 13-15, 1987, pp. 60-61.
22. D. S. Neculescu, *Failure Diagnosis, Fail Safe Operation and Reliability of Robots*, Proc. DREP/RRMC Military Robotic Applications Workshop, Victoria, B. C., Aug., 1987, pp. 136-142.
23. D. S. Neculescu, A. Fahim and C. Lu, *Robot Accuracy Evaluation for Arbitrarily Distributed Random Errors*, Submitted for publication in IEEE Trans. on Reliability.
24. Ingram Olkin, Leon J. Gleser and Cyrus Derman, *Probability Models and Applications*,
25. Emanuel Parzen, *Modern Probability Theory and Its Applications*, Wiley, 1960.
26. R. P. Paul, *Robot Manipulators : Mathematics, Programming and Control*, Cambridge, MA: MIT Press, 1982.

27. H. W. Stone, A. C. Sanderson and C. P. Neuman, *Arm Signature Identification*, in Proc. 1986 IEEE Int. Conf. Robotics and Automation, San Francisco, CA., April 1986.
28. J. S. Shamma and D. E. Whitney, *A Method for Inverse Robot Calibration*, Tran. ASME, J. of Dyn. Syst. Meas. and Control, Vol. 109, pp. 36-43, March 1987.
29. W. K. Veitschegger and Chi-Haur Wu, *Robot Accuracy Analysis Based on Kinematics*, IEEE J. of Robotics and Automation, Vol. Ra-2, No. 3, September 1986.
30. K. J. Waldron, *Positioning Accuracy of Manipulators*, Paper delivered at the National Science Foundation Workshop on the Impact on the Academic Community of Required Research Activity for Generalized Robotic Manipulators., U. of Florida, Gainesville, Fla., 1978.
31. D. E. Whitney, *The Mathematics of Prosthetic Arms and Manipulators*, Tran. ASME, J. of Dyn. Syst. Meas. and Control. May 1972, pp. 447-453.
32. D. E. Whitney, C. A. Lozinski and J. M. Rourke, *Industrial Robot Calibration Method And Results*, in Proc. ASME Conf. Computers and Engineering, Las Vegas, November 1984.
33. Chi-Haur Wu, *The Kinematic Error Model for The Design of Robot Manipulator*, Proceedings of the 1983 American Control Conference, San Francisco, CA, June 1983.
34. Chi-Haur Wu, *A Kinematic CAD Tool for The Design and Control of a Robot Manipulator*, The International J. of Robotics Research, Vol. 3, No. 1, Spring 1984.

Appendix A

Modified A Transformation Matrix

For a N degrees-of-freedom robot manipulator, it is necessary that the manipulator has at least N joints. If we have assigned coordinate frames to all links according to the D-H conventions, we can establish the relationship between successive frames $n - 1$ and n by the following rotations and translations :

- rotate about Z_{n-1} , an angle θ_n .
- translate along Z_{n-1} , a distance d_n .
- translate along rotated $X_{n-1} = X_n$, a length a_n .
- rotate about X_n , the twist angle α_n .

This may be expressed as the product of four homogeneous transformations relating the coordinate frame of link n to the coordinate frame of link $n - 1$ [26], [6]. This relationship is called as A matrix which can be expressed as

$$A_n = Rot(Z_{n-1}, \theta_n) \cdot Trans(Z_n, d_n) \cdot Trans(a_n, 0, 0) \cdot Rot(X_n, \alpha_n) \quad (A.1)$$

which is equal to

$$A_n = \begin{bmatrix} C\theta_n & -S\theta_n C\alpha_n & S\theta_n S\alpha_n & a_n C\theta_n \\ S\theta_n & C\theta_n C\alpha_n & -C\theta_n S\alpha_n & a_n S\theta_n \\ 0 & S\alpha_n & C\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (A.2)$$

For the analysis presented in this dissertation, an extra rotation matrix is multiplied to compensate for the errors in parallel or near parallel consecutive joints. Even when the consecutive joints are not parallel or near parallel, inclusion of this extra rotation term is convenient for the error propagation analysis described in the following chapters.

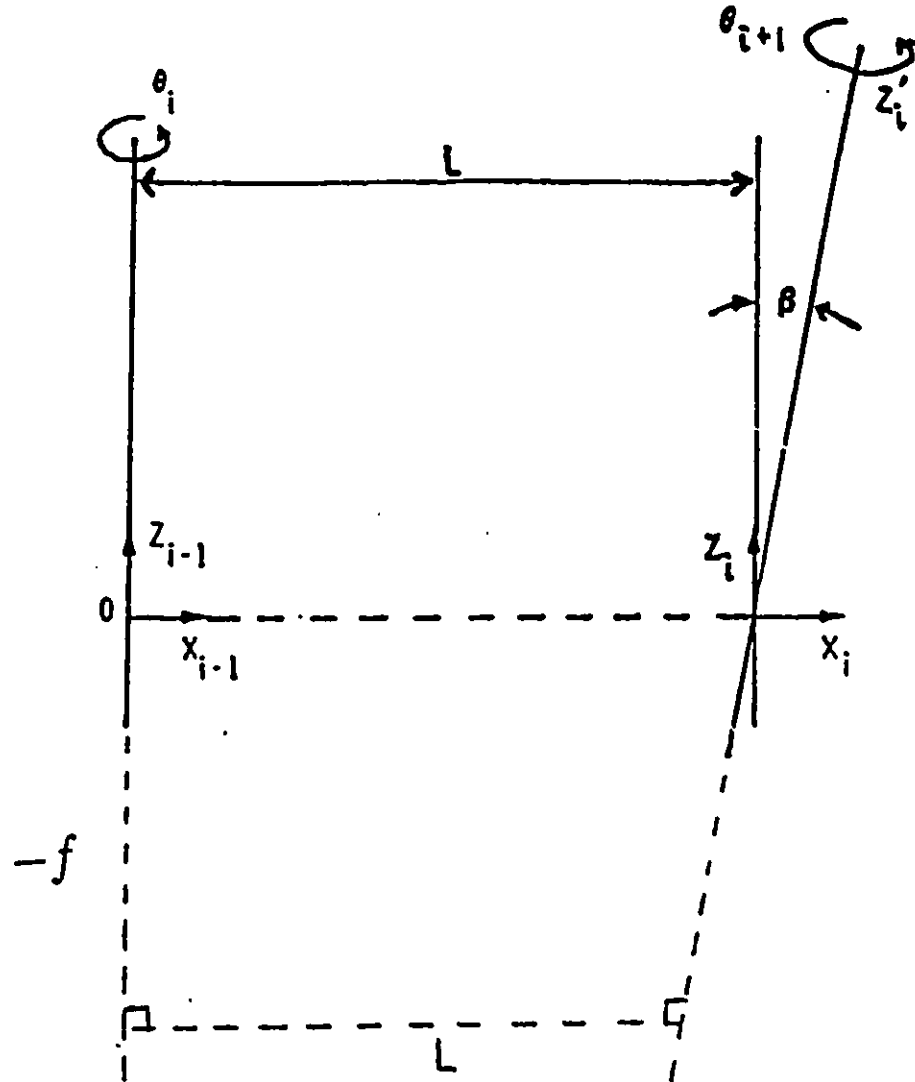


Figure A.1: drawing of two consecutive axes with near parallel joint axes

Appendix B

Program for Predicting The Positioning and Orientational Error of a End-Effector

This chapter includes four programs which can predict the positioning and orientational error distribution functions for one degree-of-freedom, two degrees-of-freedom and three degrees-of-freedom robot manipulators. The first program can predict the positioning and orientational errors from the propagation of the $5N$ link parameter errors of a N degrees-of-freedom robot manipulator. The second program is for predicting the distribution function of Cartesian positioning error DY of a one degree-of-freedom revolute robot manipulator, resulting from the joint variable error in the defined work volume. The third program is for predicting the distribution function of Cartesian positioning error DY of a two degrees-of-freedom revolute robot manipulator, resulting from the mixed error propagation of the joint variable errors in the defined work volume. The fourth program is used for predicting the distribution function of Cartesian positioning error DZ , resulting from the mixed error propagation of the joint variable errors in the defined work volume.

B.1 Error Propagation Program for Fixed 5N Link Parameter Errors

```
C *****
C *
C *          ROBOT ERROR PROPAGATION
C *
C *          Algorithms by
C *
C *          D. Necsulescu
C *          A. Fahim
C *          Chun Lu
C *
C *****
C *      Work is based on "Robot Accuracy Analysis Based on Kinematics"
C *      W.K. Veitschegger and Chi-Haur Wu, IEEE Journal of Robotics,
C *      Vol. RA-2, No.3, Sept. 1986
C *      *****
C *
C *          Nomenclature:
C *
C *          I = NUMBER OF LINKS
C *          L = Rows of the rotation matrix
C *          M = Columns of the rotation matrix = 3
C *          K = ERROR PARAMETERS = 5
C *
C *
C *
C *****
C
C
PROGRAM ERPRPG
CALL WS
CALL ERROR
```

CALL OUTPUT

STOP

END

C

C

C

BLOCK DATA

REAL R(3,3),P(3,1),M1(3,1),M2(3,1),M3(3,1),M4(3,1),M5(3,1),

W1(3,6),W2(3,6),W3(3,6),W4(3,6),W5(3,6),W6(3,6)

REAL THE(6),ALPS(6),OFS(6),LEN(6),DTHE(6,1),DOFS(6,1),DLEN(6,1),

DALP(6,1),DBET(6,1),D1(3,1),DEL1(3,1),DP(3,1)

COMMON /AAA/R,P,M1,M2,M3,M4,M5,W1,W2,W3,W4,W5,W6

COMMON/BBB/THE,ALPS,OFS,LEN,DTHE,DOFS,DLEN,DALP,DBET,D1,DEL1,DP,

BET

DATA THE/6*0.5236/

DATA OFS/0,0,0.14986,0.43307,0,0/

DATA LEN/0,0.4318,-0.02032,0,0,0/

DATA ALPS,BET/-1.5708,-3.1415927,1.5708,-1.5708,1.5708,0,0/

DATA DTHE,DALP,DBET,DLEN,DOFS/18*0.017453,12*0.0001/

END

C

C

C

SUBROUTINE ROT(THE,BET,ALP,R)

DIMENSION R(3,3)

BET=0

R(1,1) = COS(THE)*COS(BET)-SIN(THE)*SIN(ALP)*SIN(BET)

R(1,2) = -SIN(THE)*COS(ALP)

R(1,3) = COS(THE)*SIN(BET)+SIN(THE)*SIN(ALP)*COS(BET)

R(2,1) = SIN(THE)*COS(BET)+COS(THE)*SIN(ALP)*SIN(BET)

R(2,2) = COS(THE)*COS(ALP)

R(2,3) = SIN(THE)*SIN(BET)-COS(THE)*SIN(ALP)*COS(BET)

R(3,1) = -COS(ALP)*SIN(BET)

R(3,2) = SIN(ALP)

R(3,3) = COS(ALP)*COS(BET)

```

RETURN
END
C
C
C
SUBROUTINE TRA(LEN,THE,OFS,P)
REAL LEN,P(3,1)
P(1,1) = LEN*COS(THE)
P(2,1) = LEN*SIN(THE)
P(3,1) = OFS
RETURN
END
C
C
C
SUBROUTINE MS(THE,ALP,OFS,LEN,M1,M2,M3,M4,M5)
REAL LEN, M1(3,1),M2(3,1),M3(3,1),M4(3,1),M5(3,1)
M1(1,1) = 0
M1(2,1) = 0
M1(3,1) = 1
M2(1,1) = COS(THE)
M2(2,1) = SIN(THE)
M2(3,1) = 0
M3(1,1) = -OFS*SIN(THE)
M3(2,1) = OFS*COS(THE)
M3(3,1) = 0
M4(1,1) = LEN*SIN(THE)*SIN(ALP)-OFS*COS(THE)*COS(ALP)
M4(2,1) = -LEN*COS(THE)*SIN(ALP)-OFS*SIN(THE)*COS(ALP)
M4(3,1) = LEN*COS(ALP)
M5(1,1) = -SIN(THE)*COS(ALP)
M5(2,1) = COS(THE)*COS(ALP)
M5(3,1) = SIN(ALP)
RETURN
END
C
C

```

C

SUBROUTINE WS

```
REAL R(3,3),P(3,1),M1(3,1),M2(3,1),M3(3,1),M4(3,1),M5(3,1),W1(3,6)
,W2(3,6),W3(3,6),W4(3,6),W5(3,6),W6(3,6),TEMP(3,1),TEMP2(3,1)
REAL THE(6),ALPS(6),OFS(6),LEN(6),DTHE(6,1),DOFS(6,1),DLEN(6,1),
DALP(6,1),DBET(6,1),D1(3,1),DEL1(3,1),DP(3,1)
COMMON /AAA/R,P,M1,M2,M3,M4,M5,W1,W2,W3,W4,W5,W6
COMMON /BBB/THE,ALPS,OFS,LEN,DTHE,DOFS,DLEN,DALP,DBET,D1,DEL1,DP,
```

BET

```
DIMENSION WW1(3,1),WW2(3,1),WW3(3,1),WW4(3,1),WW5(3,1),WW6(3,1),
RR(3,3),PP(3,1),RTEMP(3,3)
```

```
DO 100 I=1,3
```

```
CALL MS(THE(I),ALPS(I),OFS(I),LEN(I),M1,M2,M3,M4,M5)
```

```
II=I-1
```

```
IF (II.GT.0) GOTO 20
```

```
W1(1,1) = 0
```

```
W1(2,1) = 0
```

```
W1(3,1) = 0
```

```
W2(1,1) = 0
```

```
W2(2,1) = 0
```

```
W2(3,1) = 1
```

```
W3(1,1) = M2(1,1)
```

```
W3(2,1) = M2(2,1)
```

```
W3(3,1) = 0
```

```
W4(1,1) = M3(1,1)
```

```
W4(2,1) = M3(2,1)
```

```
W4(3,1) = 0
```

```
W5(1,1) = M5(1,1)
```

```
W5(2,1) = M5(2,1)
```

```
W5(3,1) = M5(3,1)
```

```
W6(1,1) = M4(1,1)
```

```
W6(2,1) = M4(2,1)
```

```
W6(3,1) = M4(3,1)
```

```
GO TO 100
```

```
20 CONTINUE
```

```

CALL TRA(LEN(II),THE(II),OFS(II),P)
CALL ROT(THE(II),BET,ALPS(II),R)
IF (II.EQ.1) GOTO 30
CALL DM(3,3,3,RR,R,RTEMP)
CALL DM(3,1,3,RR,P,TEMP2)
CALL SM(3,3,RTEMP,R,R)
CALL SM(3,1,PP,TEMP2,P)
30 CALL DM(3,1,3,R,M1,WW2)
CALL CM(P,WW2,WW1)
CALL DM(3,1,3,R,M2,WW3)
CALL CM(P,WW3,TEMP)
CALL DM(3,1,3,R,M3,TEMP2)
CALL SM(3,1,TEMP,TEMP2,WW4)
CALL DM(3,1,3,R,M5,WW5)
CALL DM(3,1,3,R,M4,TEMP2)
CALL CM(P,WW5,TEMP)
CALL SM(3,1,TEMP,TEMP2,WW6)
DO 50 M=1,3
W1(M,I) = WW1(M,1)
W2(M,I) = WW2(M,1)
W3(M,I) = WW3(M,1)
W4(M,I) = WW4(M,1)
W5(M,I) = WW5(M,1)
W6(M,I) = WW6(M,1)
DO 50 N=1,3
DO 50 NN=1,3
RR(N,NN)=R(N,NN)
PP(N,1)=P(N,1)
50 CONTINUE
100 CONTINUE
RETURN
END
C
C
C
SUBROUTINE ERROR

```

```

REAL R(3,3),P(3,1),M1(3,1),M2(3,1),M3(3,1),M4(3,1),M5(3,1),
W1(3,6),W2(3,6),W3(3,6),W4(3,6),W5(3,6),W6(3,6),TEMP(3,1)
REAL THE(6),ALPS(6),OFS(6),LEN(6),DTHE(6,1),DOFS(6,1),DLEN(6,1),
DALP(6,1),DBET(6,1),D1(3,1),DEL1(3,1),DP(3,1)
COMMON /AAA/R,P,M1,M2,M3,M4,M5,W1,W2,W3,W4,W5,W6
COMMON /BBB/THE,ALPS,OFS,LEN,DTHE,DOFS,DLEN,DALP,DBET,D1,DEL1,DP,

```

```

BET

```

```

DIMENSION RR(3,3),PP(3,1),TEMP1(3,1),TEMP2(3,1)

```

```

CALL DM(3,1,3,W1,DTHE,TEMP1)

```

```

CALL DM(3,1,3,W2,DOFS,TEMP)

```

```

CALL SM(3,1,TEMP1,TEMP,TEMP2)

```

```

CALL DM(3,1,3,W3,DLEN,TEMP)

```

```

CALL SM(3,1,TEMP2,TEMP,TEMP1)

```

```

CALL DM(3,1,3,W4,DALP,TEMP)

```

```

CALL SM(3,1,TEMP1,TEMP,TEMP2)

```

```

CALL DM(3,1,3,W6,DBET,TEMP)

```

```

CALL SM(3,1,TEMP2,TEMP,D1)

```

```

CALL DM(3,1,3,W2,DTHE,TEMP1)

```

```

CALL DM(3,1,3,W3,DALP,TEMP)

```

```

CALL SM(3,1,TEMP1,TEMP,DEL1)

```

```

CALL DM(3,1,3,W5,DBET,TEMP)

```

```

CALL SM(3,1,DEL1,TEMP1,TEMP1)

```

```

CALL SM(3,1,TEMP1,TEMP,DEL1)

```

```

CALL TRA(LEN(3),THE(3),OFS(3),PP)

```

```

CALL ROT(THE(3),BET,ALPS(3),RR)

```

```

CALL DM(3,1,3,R,PP,TEMP)

```

```

CALL SM(3,1,TEMP,P,TEMP1)

```

```

CALL CM(DEL1,TEMP1,TEMP2)

```

```

CALL SM(3,1,TEMP2,D1,DP)

```

```

RETURN

```

```

END

```

```

C

```

```

C

```

```

C

```

```

SUBROUTINE SM(I,J,A,B,C)

```

```

DIMENSION A(I,J),B(I,J),C(I,J)
DO 10 M=1,I
DO 10 L=1,J
C(M,L) = 0
C(M,L) = A(M,L) + B(M,L)
10 CONTINUE
RETURN
END
C
C
C
SUBROUTINE CM(U,V,W)
DIMENSION U(3,1),V(3,1),W(3,1)
W(1,1) = U(2,1)*V(3,1) - U(3,1)*V(2,1)
W(2,1) = U(3,1)*V(1,1) - U(1,1)*V(3,1)
W(3,1) = U(1,1)*V(2,1) - U(2,1)*V(1,1)
RETURN
END
C
C
C
SUBROUTINE DM(M,L,N,A,B,C)
DIMENSION A(M,N),B(N,L),C(M,L)
DO 10 I=1,M
DO 10 J=1,L
C(I,J) = 0
DO 10 K=1,N
C(I,J) = A(I,K)*B(K,J) + C(I,J)
10 CONTINUE
RETURN
END
C
C
C
SUBROUTINE OUTPUT

```

```

REAL R(3,3),P(3,1),M1(3,1),M2(3,1),M3(3,1),M4(3,1),M5(3,1),W1(3,6)
,W2(3,6),W3(3,6),W4(3,6),W5(3,6),W6(3,6)
REAL THE(6),ALPS(6),OFS(6),LEN(6),DTHE(6,1),DOFS(6,1),DLEN(6,1)
,DALP(6,1),DBET(6,1),D1(3,1),DEL1(3,1),DP(3,1)
COMMON /AAA/R,P,M1,M2,M3,M4,M5,W1,W2,W3,W4,W5,W6
COMMON /BBB/THE,ALPS,OFS,LEN,DTHE,DOFS,DLEN,DALP,DBET,D1,DEL1,DP,

```

```

BET

```

```

OPEN(UNIT=6,FILE='ERRPRPG DATA')

```

```

WRITE(6,20)

```

```

20 FORMAT(8X,'DX',8X,'DY',8X,'DZ',5X,'DEL1X',5X,'DEL1Y',5X,'DEL1Z'//)

```

```

WRITE(6,40) DP(1,1),DP(2,1),DP(3,1),DEL1(1,1),DEL1(2,1),DEL1(3,1)

```

```

40 FORMAT(6(1X,F9.7))

```

```

DO 80 I=1,3

```

```

DO 80 J=1,3

```

```

WRITE (6,60)W1(I,J),W2(I,J),W3(I,J),W4(I,J),W5(I,J),W6(I,J)

```

```

60 FORMAT (6(1X,F9.5))

```

```

80 CONTINUE

```

```

CLOSE(6)

```

```

RETURN

```

```

END

```

B.2 Program for Predicting The Distribution of The Cartesian Positioning Error DY of a 1 DOF Robot Manipulator

```

C*****
C*          Program for Predicting The Distribution of The Cartesian
C*          Positioning Error Dy of 1 DOF Robot Manipulator
C*****
PROGRAM DY1D
IMPLICIT DOUBLEPRECISION (A-Z)
C*****
C*          Input Data of Link Parameters and Link Parameter Errors
C*****
DATA A1,PI /1.,3.14159265358979323846/
SIGDT1=0.1*PI/180.
SS=0.5D0/(SIGDT1**2)
PP=1.D0/(SIGDT1*SQRT(2.D0*PI))
DT1MAX=5.*SIGDT1
DT1MIN=-DT1MAX
Y1MAX=A1*DT1MAX
Y1MIN=-Y1MAX
Y1=Y1MIN
DY1=(Y1MAX-Y1MIN)/100.
XI=1.
AA=0.
AN=0.
Y2MAX=PI/2.
Y2MIN=-Y2MAX
TT=DT1MAX-DT1MIN
DY2=PI/4000.
C*****
C*          Starting Loop for Calculating the CDF Value of DY

```

```

C*****
DO 180 XI=1,101
FY1=0.
FN1=0.
Y2=Y2MIN
C*****
C*           Starting Loop for Finding PDF of DY by the Marginal
C*           Distribution Method
C*****
DO 50 CI=1,4000
XJ=1./ABS(A1*COS(Y2))
IF (XJ.GT.1.D+6) GOTO 50
DT1=Y1/(A1*COS(Y2))
IF (ABS(DT1).GT.DT1MAX) GOTO 50
T2=Y2*180./PI
IF (T2.GE.(-15).AND.T2.LE.15) THEN
GOTO 5
ELSE
GOTO 10
ENDIF
5 X=(DT1-DT1MIN)/TT
IF (X.EQ.0) GOTO 50
PDT1=X*(1.D0-X)**9*110.D0
GOTO 45
10 IF (ABS(T2).GT.15.AND.ABS(T2).LT.75) THEN
GOTO 15
ELSE
GOTO 20
ENDIF
15 X=(DT1-DT1MIN)/TT
PDT1=20.*X*(1.-X)**3
GOTO 45
20 PDT1=1.
45 PNT1=EXP(-(DT1**2)*SS)
FY1=FY1+PDT1*XJ
FN1=FN1+PNT1*XJ

```

```
50 Y2=Y2MIN+CI*DY2
FY1=FY1/PI*DY2/TT
FN1=FN1/PI*DY2*PP
130 WRITE(5,150) XI,Y1,FY1,FN1
150 FORMAT(4(3X,E10.4))
AA=AA+FY1*DY1
AN=AN+FN1*DY1
WRITE(1,*) 'HELLO Y1=',XI
180 Y1=Y1MIN+DY1*XI
WRITE(5,200) AA,AN
200 FORMAT('AA=',E10.4,5X,'AN=',E10.4)
STOP
END
```

B.3 Program for Predicting The Distribution Function of The Cartesian Positioning Error DY of a 2 DOF Robot Manipulator

```

C*****
C*           Program for Predicting the Error Distribution of
C*           The Cartesian Positioning Error DY of 2 DOF Robot
C*****
PROGRAM YH2D
IMPLICIT DOUBLEPRECISION (A-Z)
C*****
C*           Input Data of Link Parameters
C*           and Link Parameter Errors
C*****
DATA A1,A2,PI,BI /2*1.0,3.14159265358979323846,100/
DATA CI,DI /40.,40./
SIGDT=0.1*PI/180.
DTMAX=5.*SIGDT
DTMIN=-DTMAX
TT=(DTMAX-DTMIN)
Y1MAX=(A1+2.*A2)*DTMAX
Y1MIN=-Y1MAX
DY1=(Y1MAX-Y1MIN)/BI
DY2=PI/CI
DY3=DY2
DY4=(DTMAX-DTMIN)/DI
SY=0.5/(SIGDT**2)
SS=1.0/(PI**2)/(TT**2)
PP=0.5/(PI**3)/(SIGDT**2)
AA=0.
AN=0.
XI=1.
Y1=Y1MIN

```

```

C*****
C*           Starting Loop for Calculating CDF Value of DY
C*****
DO 1200 BB=1,BI+1
Y2=-PI/2.
FY1=0.
FN1=0.
C*****
C*           Starting Loop for Finding PDF Value of DY
C*           by the Marginal Distribution Method
C*****
DO 1100 C1=1,CI
CO2=COS(Y2)
S2=SIN(Y2)
YT2=Y2*180./PI
Y3=-PI/2.
FY2=0.
FN2=0.
DO 1000 C2=1,CI
CO23=COS(Y2+Y3)
S23=SIN(Y2+Y3)
YT23=(Y2+Y3)*180./PI
XJ=1./ABS(A2*CO23)
IF (XJ.GT.1.D6) GOTO 1000
Y4=DTMIN
FY3=0.
FN3=0.
DO 900 D1=1,DI
X4=(Y4-DTMIN)/TT
DT2=(Y1-(A1*CO2+A2*CO23)*Y4)/(A2*CO23)
IF (DT2.LT.DTMIN.OR.DT2.GT.DTMAX) GOTO 900
IF ((-15).LE.YT2.AND.YT2.LE.15) THEN
GOTO 50
ELSE
GOTO 100
ENDIF

```

```

50 PY4=X4*(1.-X4)**9*110.
GOTO 300
100 IF (15.LT.YT2.AND.YT2.LT.75) THEN
GOTO 150
ELSEIF ((-75).LT.YT2.AND.YT2.LT.(-15)) THEN
GOTO 150
ELSE
GoTO 200
ENDIF
150 PY4=20.*X4*(1.-X4)**3
GOTO 300
200 PY4=1.D0
300 CONTINUE
X5=(DT2-DTMIN)/TT
PN4=EXP(-Y4**2*SY)
IF ((-15).LE.YT23.AND.YT23.LE.15) THEN
GOTO 370
ELSEIF (165.LE.YT23.AND.YT23.LE.180) THEN
GOTO 360
ELSEIF ((-180).LE.YT23.AND.YT23.LE.(-165)) THEN
GOTO 360
ELSE
GOTO 400
ENDIF
360 X5=1.-X5
370 PDT2=X5*(1.-X5)**9*110.
GOTO 600
400 IF (15.LT.YT23.AND.YT23.LT.75) THEN
GOTO 450
ELSEIF ((-75).LT.YT23.AND.YT23.LT.(-15)) THEN
GOTO 450
ELSEIF (105.LT.YT23.AND.YT23.LT.165) THEN
GOTO 440
ELSEIF ((-165).LT.YT23.AND.YT23.LT.(-105)) THEN
GOTO 440
ELSE

```

```

GOTO 500
ENDIF
440 X5=1.-X5
450 PDT2=20.*X5*(1.-X5)**3
GOTO 600
500 PDT2=1.
600 CONTINUE
PNDT2=EXP(-DT2**2*SY)
FY3=FY3+PDT2*PY4*DY4
FN3=FN3+PNDT2*PN4*DY4
900 Y4=DTMIN+D1*DY4
FY2=FY2+FY3*DY3*XJ
FN2=FN2+FN3*DY3*XJ
1000 Y3=-PI/2.+C2*DY3
FY1=FY1+FY2*DY2
FN1=FN1+FN2*DY2
1100 Y2=-PI/2.+C1*DY2
FY1=FY1*SS
FN1=FN1*PP
WRITE(5,1150) XI,Y1,FY1,FN1
1150 FORMAT(4(3X,E10.4))
AA=AA+FY1*DY1
AN=AN+FN1*DY1
WRITE(1,*) 'HELLO !! Y1=',XI
XI=XI+1.
1200 Y1=Y1MIN+BB*DY1
WRITE(5,1300) AA,AN
1300 FORMAT('AA=',E10.4,5X,'AN=',E10.4)
STOP
END

```

B.4 Program for Predicting The Distribution of The Cartesian Positioning Error DZ of a 3 DOF Robot Manipulator

```

C*****
C*           Program for Predicting The Distribution of The Cartesian
C*           Positioning Error DZ of 3 DOF Robot
C*****
PROGRAM ZT3D
IMPLICIT DOUBLEPRECISION (A-Z)
C*****
C*           Input Data of Link Parameters and Link Parameter Errors
C*****
INTEGER BI,CI,DI,C1,C2,C3,C4,C5
DATA A2,A3,PI /0.4318D0,0.43307D0,3.14159265358979323846D0/
DATA BI,CI,DI,D2 /100,50,20,0.14909D0/
SIGDT=0.1D0*PI/180.D0
DTMAX=5.D0*SIGDT
DTMIN=-DTMAX
TT=DTMAX-DTMIN
Y1MAX=(A2+2*A3)*DTMAX
Y1MIN=-Y1MAX
DY1=(Y1MAX-Y1MIN)/DBLE(BI)
DY2=PI/DBLE(CI)
DY3=DY2
DY4=TT/DBLE(DI)
YMIN=-PI/2.D0
FAC=180.D0/PI
CON=1.D0/(PI**2)/(TT**2)*DY2*DY3*DY4
SS=0.5D0/(SIGDT**2)
PP=1.D0/(PI**2)*DY2*DY3*DY4/(SIGDT**2)/(2.D0*PI)
AA=0.
AN=0.

```

```

XI=1.D0
Y1=Y1MIN
C*****
C*           Starting Loop for Calculating CDF Value of DZ
C*****
DO 1000 BB=1,BI+1
Y2=YMIN
FY2=0.D0
FN2=0.
C*****
C*           Starting Loop for Finding PDF Value of DY by The
C*           Marginal Distribution Method
C*****
DO 900 C1=1,CI
DC2=DCOS(Y2)
YT2=Y2*FAC
Y3=YMIN
FY3=0.D0
FN3=0.
DO 800 C2=1,CI
DC23=DCOS(Y2+Y3)
XT=DABS(A3*DC23)
IF (XT.LT.1.D-5) GOTO 800
YT23=(Y2+Y3)*FAC
Y4=DTMIN
FY4=0.D0
FN4=0.
DO 700 C3=1,DI
DT3=(Y1-(-A3*DC23-A2*DC2)*Y4)/(-A3*DC23)
IF (DT3.LT.DTMIN.OR.DT3.GT.DTMAX) GOTO 700
IF ((-15.D0).LE.YT2.AND.YT2.LE.15.D0) THEN
GOTO 50
ELSE
GOTO 100
ENDIF
50 X4=(Y4-DTMIN)/TT

```

```

PY4=X4*(1.D0-X4)**9*110.D0
GOTO 300
100 IF (15.D0.LT.YT2.AND.YT2.LT.75.D0) THEN
GOTO 150
ELSEIF ((-75.D0).LT.YT2.AND.YT2.LT.(-15.D0)) THEN
GOTO 150
ELSE
GOTO 200
ENDIF
150 X4=(Y4-DTMIN)/TT
PY4=20.D0*X4*(1.D0-X4)**3
GOTO 300
200 PY4=1.D0
300 CONTINUE
PN4=EXP(-(Y4**2)*SS)
IF ((-15.D0).LE.YT23.AND.YT23.LE.15.D0) THEN
GOTO 350
ELSEIF (165.D0.LE.YT23.AND.YT23.LE.180.D0) THEN
GOTO 360
ELSEIF ((-180.D0).LE.YT23.AND.YT23.LE.(-165.D0)) THEN
GOTO 360
ELSE
GOTO 400
ENDIF
350 X=(DT3-DTMIN)/TT
GOTO 370
360 X=1.-(DT3-DTMIN)/TT
370 PDT3=X*(1.D0-X)**9*110.D0
GOTO 600
400 IF (75.D0.LE.YT23.AND.YT23.LE.105.D0) THEN
GOTO 450
ELSEIF ((-105.D0).LE.YT23.AND.YT23.LE.(-75.D0)) THEN
GOTO 450
ELSE
GOTO 500
ENDIF

```

```

450 PDT3=1.D0
GOTO 600
500 IF (YT23.GT.15.AND.YT23.LT.75) THEN
GOTO 510
ELSEIF (YT23.GT.(-75).AND.YT23.LT.(-15)) THEN
GOTO 510
ELSE
GOTO 520
ENDIF
510 X=(DT3-DTMIN)/TT
GOTO 530
520 X=1.-(DT3-DTMIN)/TT
530 PDT3=20.D0*X*(1.D0-X)**3
600 CONTINUE
PNT3=EXP(-(DT3**2)*SS)
FY4=FY4+PY4*PDT3
FN4=FN4+PN4*PNT3
700 Y4=DTMIN+DY4*DBLE(C3)
FY3=FY3+FY4/XT
FN3=FN3+FN4/XT
800 Y3=YMIN+DY3*DBLE(C2)
FY2=FY2+FY3
FN2=FN2+FN3
900 Y2=YMIN+DY2*DBLE(C1)
FY2=FY2*CON
FN2=FN2*PP
WRITE(5,950) XI,Y1,FY2,FN2
950 FORMAT(4(5X,E10.4))
IF (FY2.GT.1000) GOTO 960
AA=AA+FY2*DY1
AN=AN+FN2*DY1
960 WRITE(1,*) 'HELLO !!, Y1=',XI
XI=XI+1.D0
1000 Y1=Y1MIN+DY1*BB
WRITE(5,1050) AA,AN
1050 FORMAT(5X,'AA=',E10.4,5X,'AN=',E10.4)

```

STOP
END