A HIGH SPEED SCANNING SYSTEM FOR VISION BASED NAVIGATION/CONTROL OF MOBILE ROBOTS

VASILY RUBTSOV

A thesis submitted in partial fulfilment
of the requirements of De Montfort University
for the degree of Doctor of Philosophy

November 1998

De Montfort University

ABSTRACT

One of the main problems in the design of mobile robots is the development of creating smart integrated information systems. These systems may include different type of sensors. Usually a CCD vision system is a necessary part for these systems. This thesis considers the design of a fast mechanical scanning system for a CCD vision system and the synthesis of the optimal control for this system.

The mathematical model of the transport subsystem for mobile robots subjected to an external disturbance is created. The correctness of this model is proved on the base of simulation and experimental results. Regression coefficients are calculated and an estimate of model accuracy is carried out.

The approach of power calculation for the actuators for the fast mechanical scanning system is considered. Implementation of this approach extends the use of a general mathematical model of the transport subsystem of the mobile robot and as such considerably reduces the design time.

The main features for estimation of external disturbances are determined. Limitations of implementation of some technical solutions for mobile robot sensors are defined according to an analysis conducted of different factors for external disturbances.

Different kinematic schemes of the scanning systems have been analysed in this thesis. Practical recommendations of the kinematic scheme used in scanning systems are given. The essential features of a kinematic scheme for the fast mechanical scanning system have been developed and verified.

A method for the solution of the inverse kinematics of a 3 degree of freedom scanning system in terms of velocities and in accelerations is presented. This method is utilised for formulating optimal control for the fast mechanical scanning system.

The algorithms of fast scanning have been produced for the different types of the sensors. The limitations for the practical realisations for these algorithms are considered in this thesis.

The optimal control algorithms for the developed scanning system are produced. This control minimises the sum of instant powers of the scanning system actuators. A practical algorithm has been derived utilising the control scheme structure developed theoretically. The capability of this control algorithm has been proved by experimental study. Advantages of this developed control algorithm for 3 degree scanning system has been proven by experimentation.

ACKNOWLEDGEMENTS

The author would like to thank Professor J. A. J. Knight, supervisor and Head of Department of Mechanical and Manufacturing Engineering and Professor N. A. Lakota supervisor and Head of Department of Control Systems and Robotics, for their advice and support during the course of this work.

The support of Professor V. S. Medvedev and my various colleagues at Moscow Bauman Technical University during the course of this work is gratefully acknowledged.

Finally, I would like to thank my wife Irina for her encouragement, patience and support during the course of this work.

LIST OF CONTENTS

LIST OF VARIABLES AND NOMENCLATURE

A	Amplitude, Regression Coefficient
A_1	Energy Expenditure
A_i^j	Matrix of Elementary Transformation
a	Displacement of Object
ACS	Absolute Coordinate System
AFC	Amplitude Frequency Characteristics
α	Angle of Disturbance
BFA	Block Forming of Angles
B	Auxiliary Matrix for Calculating the Matrix Derivative
BFV	Block Forming of Velocities
b	Displacement of Object
β	Angle of Disturbance
C	Auxiliary Matrix for Calculating the Matrix Derivative
Cq_i	Cosine of the q_i Angle
c	Displacement of Object
χ	Angle of Object
D	Distance between Scanning System and Goal Point
d	Reduced Moment of Inertia
γ	Angle of Object
ε	Error of Calculation or Measurement
e	Basis Vector of Rotation Axis
F	Functional of Quality
f_v	Vector of Component for $\dot{T}$ Matrix
Φ	Angle of Rotation
φ	Angle of Rotation
g_i	Linear Function
H_p	Linear Size of Scanning system
h_i	Linear Function

I	Integrator, Intensity of Action
J	Inertia Moment, Jacobi Matrix
J_{ef}	Effective Inertia Moment
j_i	Vector- Column of Jacobi Matrix
K	Criterium of Quality
k	Size of Scanning Sector
k_i	Weight Coefficient
L_i	Length
L_{MRS}	Length of the MRS Base
L_p	Linear Size of Scanning system
l	Counter of Cycle
M_A	Matrix component for Jacobian
M_v	Matrix component for Jacobian
m	Regression Coefficient, Size of Scanning Sector
MRS	Mobile Robotic System
n	Size of Scanning Sector
$O_iX_iY_iZ_i$	Cartesian Coordinate System
P	Power
P_i	Power
P_{ef}	Required Power
p_i^j	Radius Vector
Q	Vector of Generalised Coordinates
Q_i	Projection Marx
q_c	Vector of Real Generalised Coordinates
q_i	Generalised Coordinate
q_f	Vector of Desired Generalised Coordinates
$\dot{q}_i$	First Derivative of Generalised Coordinate
$\ddot{q}_i$	Second Derivative of Generalised Coordinate
R	Radius Vector
RLS	Radio Locator Station

r_i^j	Radius Vector
S	Displacement, Motion Route, Spectral Density
SCS	System of Coordinate Connected with Base of Scanning system
Sq_i	Sine of the q_i Angle
S_h	Spectral Density
s_i	AuxiliaryCoefficient
S_v	Spectral Density
T	Kinetic Power
T_{ij}	Transfer Matrix
t	Time
V	Velocity
V^O	Vector of Velocities
V_{MRS}	Velocity of Mobile Robotic System
W	Output Function of System
w	Frequency, Rotary Velocity
x	Coordinate, Vector
y	Coordinate, Vector
z	Coordinate, Vector

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

1.1 APPLICATION OF MOBILE ROBOTS

It is widely believed that the further development and technical progress in industry will be hampered without the wide application of robots. At the present time, robots of the first and the second generations are widely spread in industrially developed countries.

The technical level of robots is continuously growing. The main achievements of these directions are increasing of accuracy, speed of acting and the rise of functional abilities.

The main directions in robotics area include creating adaptive autonomous robots, which can substitute man not only in conducting monotonous hard work, and where the presence of one is impossible. For example, these robots can be applied to work in conditions dangerous for man's life. In this connection, the questions of creating and designing an autonomous Mobile Robotic Systems (MRS), which can function at variable, undetermined conditions of a real environment (for example out door applications), are special particularly at present time.

Currently a significant amount of work connected with the development of MRS is reported typically such as research by *Platonov A. et al* [1996], *Stepanov U. et al* [1996]. Within these groups, different types of MRS can be discerned. The concepts of "developed sensor systems" and "developed mobile systems "are considered in many MRS projects. Typically a major goal being to organise the group action of the robots for difficult task solving across the whole breadth of the applied functional environment (the large area of land, air, water environment, big building etc).

Two ground breaking projects in this area are:

1) The development of a mobile system for extinguishing forest fires (*Kobayashi A.* [1996]) and

2) The development of a mobile system for work in areas struck by natural catastrophes (*Umertani Y. and Hirano S.* [1991]).

The mobile systems for extinguishing of forest fires reported by Kobayashi A. was designed to receive, analyse, interpret and act upon data from several sources including

- artificial satellites;
- controlled helicopters;
- fire brigades related to the MRS;
- transport related to the transfer of the fire brigades assigned to work with to MRS;
- the mobile control centre;
- the transport to move robots and people;
- the centre of communication and the calculation centre.

The developed mobile systems for work in areas affected by a catastrophe reported by Umertani and Hirano [1991] is currently a paper study.

The realisation is scheduled for early project will be realised in the 21-st century. The general scheme of the proposed mobile system will include accommodating following:

- a transport MRS with drives based on turbo propellers;
- a flying MRS and pediculators and manipulators;
- an MRS for air reconnaissance;
- an MRS for extinguishing of fires;
- an MRS for difficult works.

It is planned to use machines and mechanisms with a construction engineering capability in this project.

1.2 Structure and Schema of a Mobile Robotic System

For such an implementation as described in 1.1 above – it is desirable for such an MRS to have the capability to scan the adjacent environment intelligently, to recognise and process information about the environment, to respond to changes in the environment adequately. Hence, an MRS must have system of technical vision and artificial intelligence capabilities.

A generalised structure and schema of an MRS is represented in Figure 1.1.

1.3 INFORMATION PROBLEMS OF MOBILE ROBOTIC SYSTEM

Interaction of the MRS and an environment is achieved on the base of the active information moving actions, such as purposeful receiving of information and proceeding of one the moving MRS. There are the following information problems for MRS:

- to scan of environment in the nearest review zone (to find obstacles);
- to scan of environment in the far zone for planning further actions;
- to scan of environment and to select the landmarks for definition of own location;
- to move along the given area for getting the insufficient information.

Using the high accuracy system of a satellite the navigation system resolves some problems in the area of MRS navigation, but the satellite navigation can not effectively applied for estimation of area permeability, for accurate local calculation of route, relative location of MRS to moving objects and so on. Generally, there is a class of problems, where the determination of the coordinates for some environmental object relatively to MRS location is more important than high resolved measurement of absolute coordinates for an MRS.

Structure and schema of a mobile robotic system

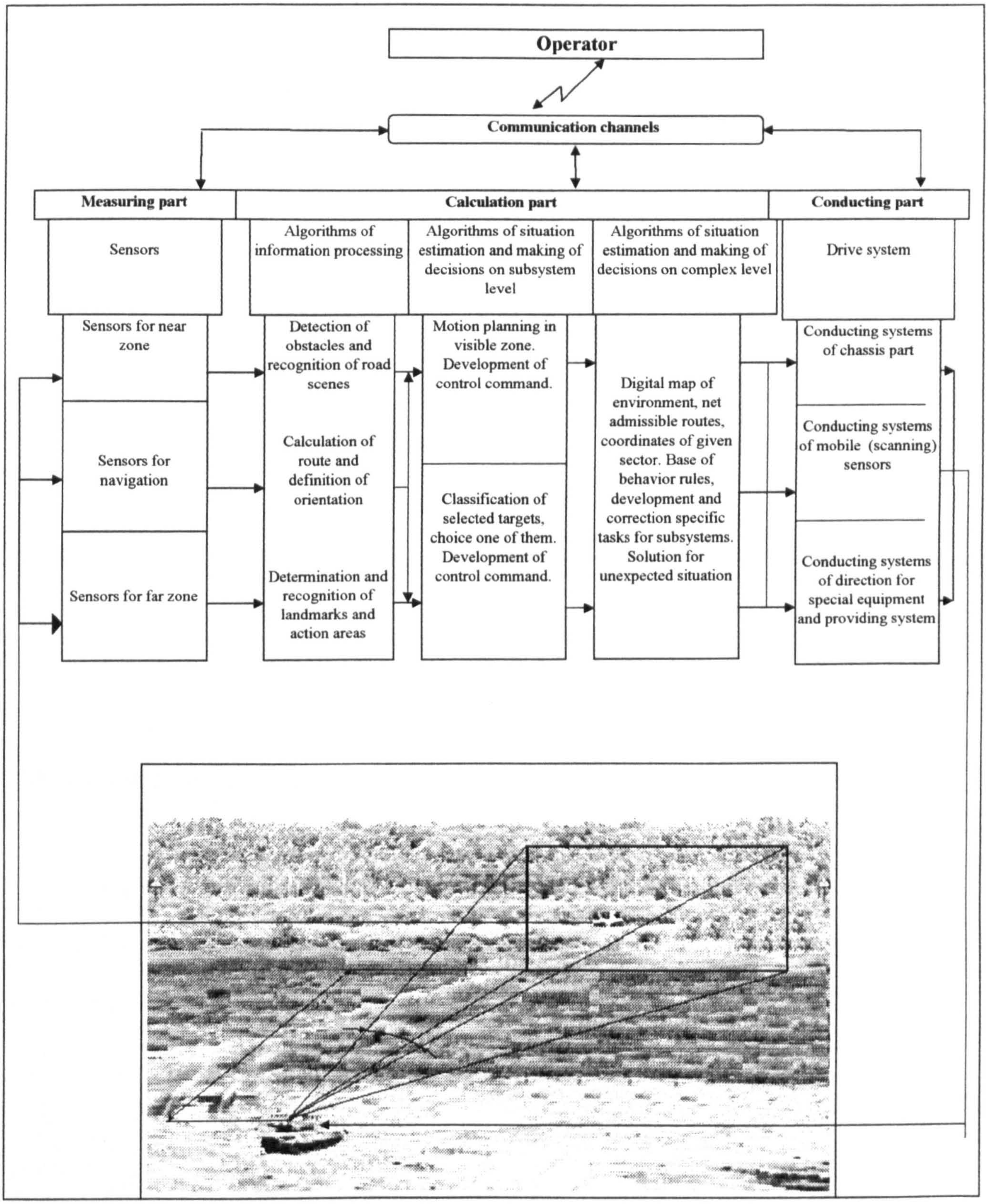


Figure 1.1 Structure and Schema of a Mobile Robotic System

Preferably, a navigation system must provide the following:

- automatic analysis of information from a CCD vision system to find the significant elements within an image (i.e. landmarks) to determine the desirable route to follow;
- to calculate the displacement of the CCD vision system in relation to the observed objects (two linear displacement in the plane of the CCD vision system view and estimation of the distance from the MRS to the object);
- to measure the linear velocity of MRS in relation to observed objects.

1.4 SENSOR PROBLEMS OF A MOBILE ROBOTIC SYSTEM

An MRS can use different sensors such as a CCD vision system, laser, radiolocator etc. Sensors which can determine information about the adjacent environment. The effectiveness of such sensors when implemented for out door application can affect the action of an MRS. For instance during motion on rough roads, the sensor may be subject to violent disturbance. In this case, using of the stabilisation devices is necessary.

Different methods and tools are used to reduce and to eliminate the influence of disturbance (such as rocking) to sensors. Typically, it may be applied as a stabilisation device for sensors alternatively as a stabilisation device for the MRS at the chassis level.

1.5 Stabilisation of Sensors for a Mobile Robotic System

The task of stabilisation devices is to provide a desired position of the sensors (line of sight of the optical device, symmetry axis of directional diagram and so on). This task must be solved with conditions of sensor base disturbance.

Although the question of sensor stabilisation where a sensor base is subject to disturbance, has been the subject of much investigation, there remain a number problems in this area after *Smirnov G.* [1990], *Kochergin V.* [1988]. These problems are characteristics of object rocking, factors forcing stabilisation of the sensors, geometry, kinematics and dynamics of the stabilisation system. The task of optimal search of required landmarks under condition of base rocking is considered in this work.

1.6 Aims and Objectives

The over aim of this research is to increase the effectiveness of the search and to provide for a more wider area of survey of the environment. In achieving, it is necessary to develop control algorithm and to choose the required kinematics for such scanning systems. Typical scheme of a scanning system is shown in Figure 1.2. To achieve this aim the following objectives have to be met.

- To determine an external disturbance to the base of sensor when the MRS is moving;
- to choose and to verify the optimal kinematic scheme for the scanning system;
- to synthesis optimal trajectories for scanning chosen environment sector;
- to synthesis optimal control for actuators of the scanning system.

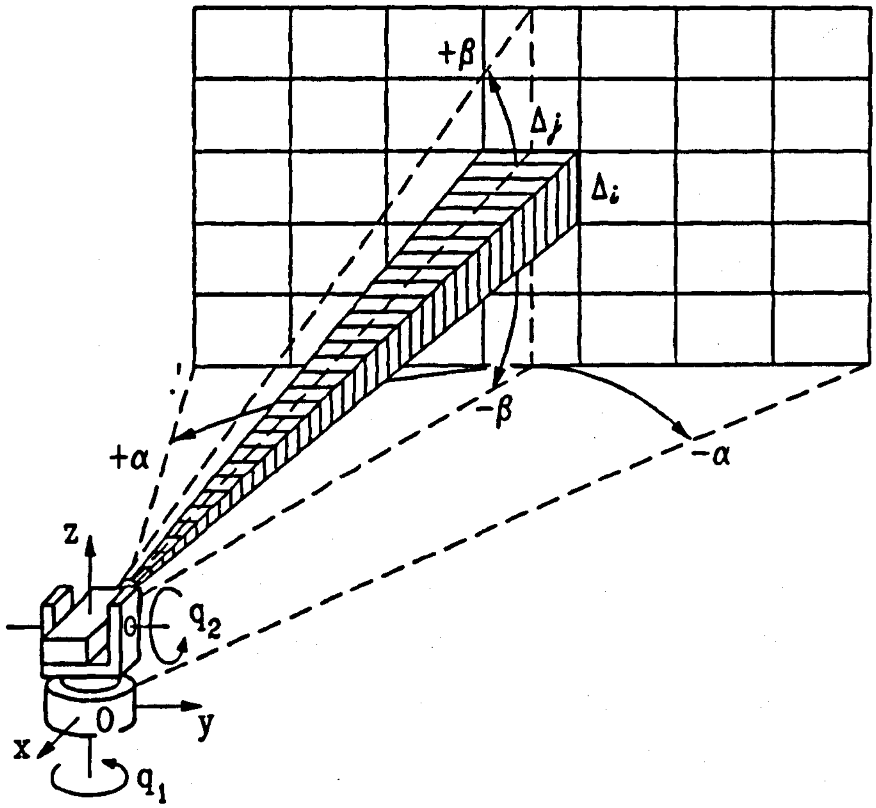


Figure 1.2 Typical Scanning System

1.7 STRUCTURE OF THESIS

The thesis has the following structure.

Second chapter includes consideration of external disturbances subjected to the sensor base when the MRS is moving. The mathematical model of the transport subsystem for the MRS when subjected an external disturbance is created during this research. The approach of initial data forming for the power calculation stage is developed according to analysis of this model. This approach allows estimating the necessary power of scanning system actuators. Carried out analysis of the created model shows the main singularities of ground mobile systems with taking into consideration external disturbance.

The third chapter includes questions of choice for an expedient configuration for the kinematic scheme of the scanning system. Analysis of different kinematic schemes for the scanning system is carried out in this chapter. The limits of implementation for the different kinematic scheme are shown in this part of thesis. Solutions of the inverse kinematic task for velocities and for accelerations for a kinematically redundant scanning system are offered in this part of thesis.

The fourth chapter is devoted to the development of algorithms for fast scanning. Different algorithms are offered for different types of sensors during investigation. The limits of practical realisation for these algorithms are considered in this chapter.

In the fifth chapter, the optimal control algorithms for the developed scanning system are synthesised during this research. The algorithms for the technical realisation of the developed control are offered.

The sixth chapter includes experimental research. In this chapter the mathematical model of the transport subsystem of an MRS subject to external disturbance is proved. The bounds for the developed control algorithm are established and proved. A comparative analysis between a scanning system with two degrees of mobility and a scanning system with three degrees of mobility is carried out.

The seventh chapter consists of discussion about the main results obtained in this thesis.

In the eighth chapter the conclusion of the thesis is formed.

The ninth chapter includes a plan for future work.

This thesis includes two appendixes. In the first appendix, the characteristics of a typical road are shown. The second appendix includes experimental results. Diagrams of time dependencies of rotary positions, speeds and accelerations under MRS motion with different linear velocities are shown.

CHAPTER 2

A MATHEMATICAL MODEL OF A TRANSPORT SUBSYSTEM FOR A MOBILE ROBOT SUBJECT TO EXTERNAL DISTURBANCE

2. A MATHEMATICAL MODEL OF A TRANSPORT SUBSYSTEM FOR A MOBILE ROBOT SUBJECT TO EXTERNAL DISTURBANCE

In determining the advisable kinematic scheme and conducting drives for a fast scanning system with an extended zone of survey it is necessary to estimate the external disturbance to the base of the scanning system when the MRS is in motion. This problem is considered in this chapter. The mathematical model for a transport subsystem MRS is described in this chapter, the limits of its application are verified. The main peculiarities of ground mobile systems creating a disturbance to the base of a scanning system are shown using analysis of the considered model. The approach offered calculates the external disturbance at the stage of the power calculation for the scanning system actuators.

Methods of digital harmonic analysis theory, and the theory of mathematical model identification are used in this chapter.

2.1 Kinematic Model of Motion of Mobile Robotic System

A mobile robotic system belongs to a certain class of ground transport vehicles. To describe an MRS, it is necessary to create the mathematical model of its motion. Results of this are considered as the initial data for the scanning system design.

Figure 2.1 is considered to state the research task, where the MRS is shown when moving along a section of a given route. The movement of the MRS is of a complicated character, therefore in the common case it may be described, in general, as the linear movement of the MRS centre of mass (O_{-1} point) and rotation relative to the MRS centre of mass (α angle). Here, only the rotary component of movement will be taken into consideration, as according to *Rogozinnikov A.* [1996] the speed of the change of angle of bank determines the so called "effect of blur" for a video image in a TV system. That works at the medium and long viewable zones (on the contrary the visualisation of the near zone is a function of the velocity of change of the linear displacement of the MRS centre of mass). The right stationary Cartesian coordinate system $O_{-1}X_{-1}Y_{-1}Z_{-1}$ is rigidly fixed at the MRS centre of mass. It is an absolute coordinate system (ACS). Rotation of the MRS in the ACS is set at the right Cartesian coordinate system $O_0X_0Y_0Z_0$. It is rigidly fixed at the base of coordinator and is taken as the Coordinate System of the Scanator (CSS).

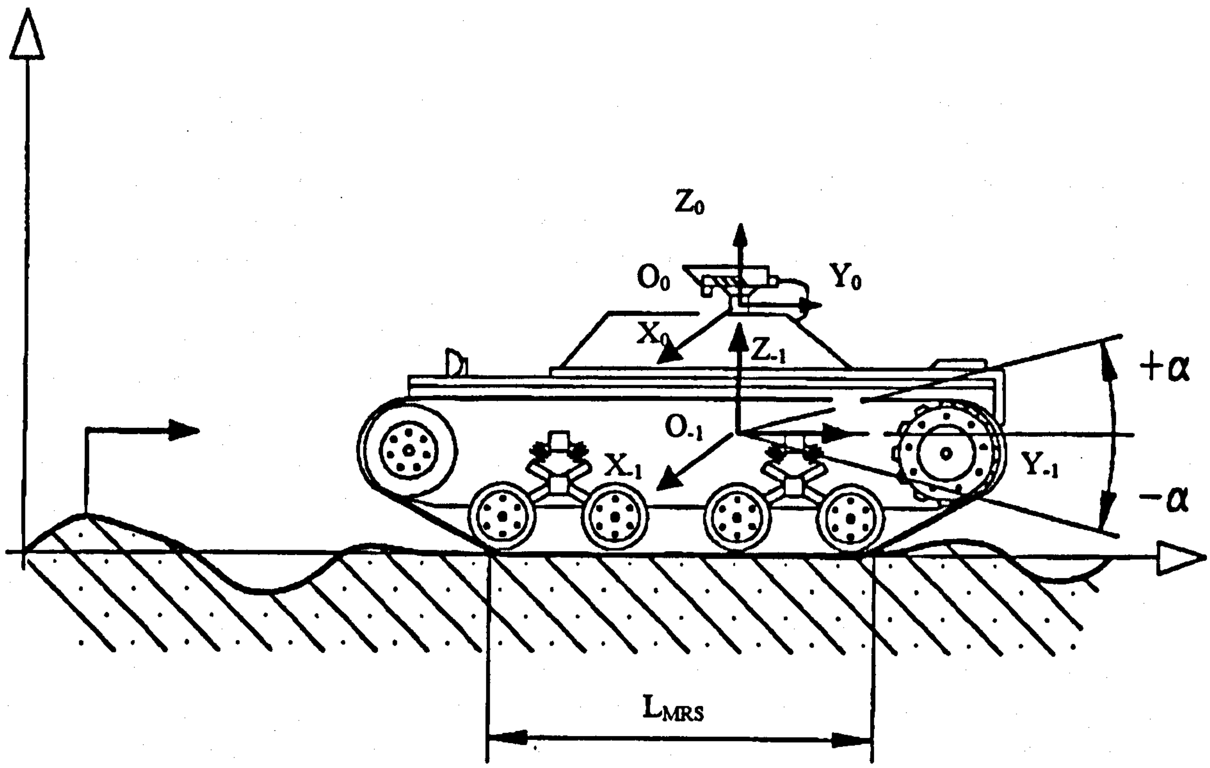


Figure 2.1 The Typical Mobile Robotic System

The rotary movement is described by the angles of pitch bank- α and roll bank (not shown). The profile of the route taken is given at the $O_tX_tY_tZ_t$ coordinate system. The pitch profile is $y_t\,(x_t)$; the roll profile is $y_t\,(z_t)$. When driving the MRS is subjected to external kinematic disturbances from the route profile. Therefore, in this model, the MRS velocity (V) along the route is considered as velocity V of route relative to the ACS. The MRS motion is considered as rectilinear and uniform (V= constant) one. The same approach has been used by *Smirnov* G. [1990], *Rogozinnikov* A. [1996]. The base length L_{MRS} for a track type MRS is shown in Figure. 2.1.

In this statement of the problem, the task is to determine of the change rules for the MRS pitch and roll banks as functions of the route profile. The results of such will be taken as the basic data for the power calculation of the scanning system and for the analysis of power characteristics of different variants for the proposed scanning systems design. For this reason, the maximum of the required parameters of disturbance is of interest. In this work, the case of driving a tracked MRS along a road is considered. Given the above statement of the task, a roll bank can be neglected according to *Wong J. Y.* [1982].

Taking into account these assumptions the set task can be reduced to the following mathematical formulation. It is necessary to determine the rule of change:

$$\alpha(t) = f\{y_t(x_t), V, L_{MRC}, t\}, - \qquad \textbf{(2.1)}$$

with limitations:

$$\dot{\alpha}_{max} = \dot{\alpha}_{max}^{real}$$

$$\ddot{\alpha}_{max} = \ddot{\alpha}_{max}^{real}$$

where:

$y_t(x_t)$ -is the pitch profile of the road for the MRS;

V –is the movement velocity of MRS driving;

$\alpha, \dot{\alpha}, \ddot{\alpha}$ are the calculated values of angles for the pitch, its velocity and accelerations;

$\{\dot{\alpha}, \ddot{\alpha}\}_{max}^{real}$ are the real values of the angle velocity and accelerations of angle for pitch.

2.2 Some Data from the Theory of Movement for Ground Transport Vehicle

Vehicle motion along irregular roads induces vibration to robot elements. Irregular surfaces can be classified by forms, by sizes and by character of the periodicity of the irregularities at the profile (that is cross-section of relief in the direction of the vehicle motion). The form of irregularities can be divided by sinusoidal, parabolic, rectangular and so on. Sizes of irregularities are determined with respect to length and height.

In a given length the irregularities can be characterised by impulse, ruts, slopes and pot-holes, according to *Smirnov G.* [1990]. Slopes are smooth irregularities, each of them has a length of more than 25 meters. It characterises the macroprofile of the surface. Irregularities can be placed across or at an acute angle to the movement direction. If the size of irregularity is more than that of the vehicle then it is named and considered a slope too.

Irregularities with lengths less then 25 meters compose a microprofile. Irregularities with a height less 1 cm and length less then 0.3 meters are called roughness. In general, they do not contribute noticeable disturbance to vehicle rocking.

The following conformities to alternation of irregularities of the road along the vehicle route are classified according to *Smirnov G.* [1990].

1. Single irregularities, which are located at a long distance. Single irregularities are pits, ditches and so on.
2. Periodic alternating irregularities with a similar size and form. This situation is probable when driving along improved roads. For example, the roads include plates of equal length. Their joints are the sources of impulse type oscillations. The road profile can be transformed to the smooth one due to gradually smoothing of joints. A smooth wavy profile can be formed due to sequential pressure to the road with constant frequency.
3. Random microprofile consists of nonregular alternating irregularities of different forms and sizes. Practically, transport vehicles mostly drive along roads with random microprofile.

Characteristics of irregularities size are represented in appendix A 1. These materials are received from *Smirnov G.* [1990].

Random change of the height of irregularity along the road can be characterised in a statistical way. It is not usual to take into consideration the characteristics of macroprofile as a function of random magnitude in the theory of transport vehicles.

According to the analysis of collected experimental research data for road microprofiles, carried out by different authors and research institutes (*Smirnov G.* [1990]), the correlation coefficients and dispersion are varied across a wide range. The transverse microprofile has smaller and shorter irregularities relative to the pitch microprofile.

At the same time, according to *Wong J.Y.* [1982] it is possible to represent a microprofile analytically. This theory explains the nature of periodic roughness on the roads. In accordance with the theory, after moving through any roughness a vehicle starts to oscillate with its own frequency and makes contact with the road over a certain interval. This interval distance depends on the oscillating frequency and velocity of the vehicle. (As the differences as the own frequency and as the exploitation velocity for different vehicles in determined conditions is small, increasing and reducing act over one and the same parts of the road). Moreover, road covering can be described by a periodic function in general, where only alternation and parameters of roughness are random.

2.3 Types of Oscillation of the Ground Transport Vehicles

It is usual to select vibrations with frequency values less than 22 Hz from the total vibration spectrum as characteristic of vehicles in motion. Such vibrations are named oscillations. Such division is conventional, but it has sound reason. It is considered that a man perceives the oscillations separately, but vibrations are perceived together (vibrations with the frequency values from 20 Hz to 20 000 Hz are perceived as audible noise). Besides, when a vehicle drives over roughness the vibrations subjected to the vehicle's occupants arise with frequencies less than 20 Hz. Vibrations with larger frequency value arise due to internal factors (as a rule, due to work of different mechanisms).

In reality where transport vehicles travel over roughness movement is generated with linear vertical, rotary pitch and rotary roll oscillations, mainly as a result of springing. The main part of the spring system of a vehicle is typically symmetric relative to the pitch axis. In this case the rotary pitch oscillations are independent from rotary roll oscillations and thus can be described separately according to *Wong J.Y.* [1982].

Oscillations can be divided into free and forced oscillations. Forced oscillations arise due to action of external forces. Free oscillations arise in the case of the absence of external forces, but where the equilibrium of the oscillating system is broken. If the disturbing force is periodic then the oscillations are steady, in other cases the oscillations are unsteady *Timoshenko S.* [1959]. During vehicle motion on a road with an irregular surface the unsteady oscillations are more widespread, free oscillations arise only when driving over a separate single irregularity due to *Smirnov G.* [1990].

In the common case, the road profile of the right track is not the same as the road profile of left track, and there is no correlation between them due to measurements of the road surface. Such a situation requires that consideration of the vehicle oscillations in the pitch and the roll vertical planes are made separately.

In practice, a real transport vehicle is a system with a large number of degrees of freedom (DOF), which can vibrate at the dominant (or subharmonics) frequency only. Values of displacements, velocities and accelerations of sprung masses depend on parameters of a vehicle and the spring system (primary, secondary or active). The influence of the oscillating system to the oscillation parameters when subjected to a periodic disturbance can be objectively estimated by means of Amplitude-Frequency Characteristics (AFC). They also serve as basis for system calculation with random action.

A typical AFC of a ground transport vehicle is shown in Figure 2.2, due to *Smirnov G.* [1990], where ν - is frequency of external action; z/q_0 is the relation ratio of the output value to irregular amplitude.

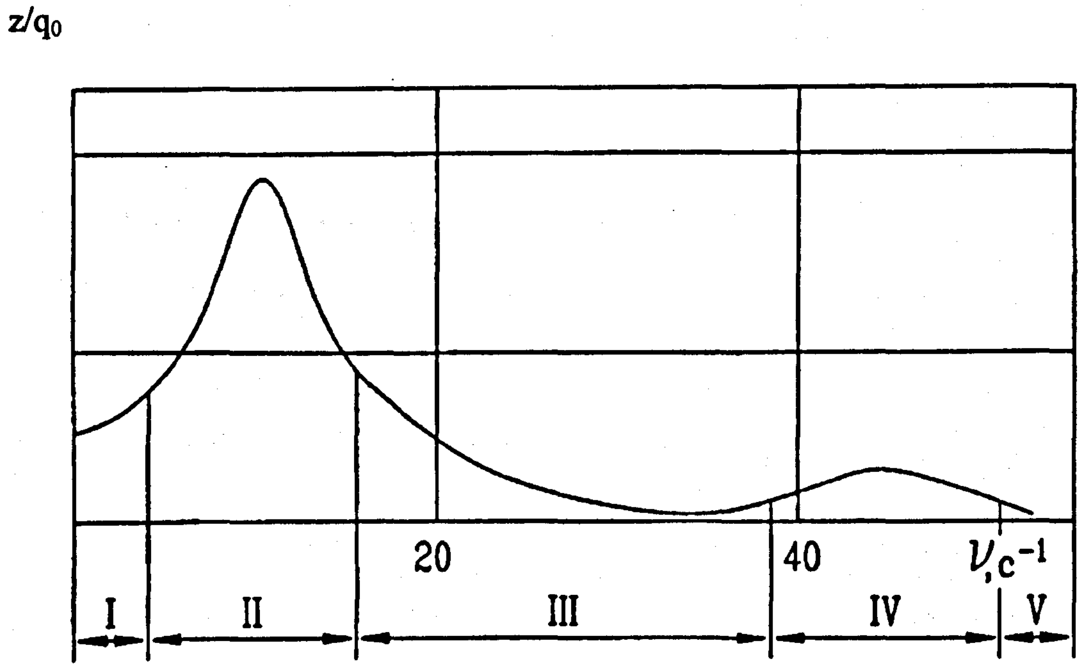


Figure 2.2 Typical Amplitude Frequency Characteristics of Ground Transport Vehicle

The shown AFC can be divided into the following parts: I-before resonance, II- low frequency resonance, III- between resonances, IV- high frequency resonance, V- after resonance.

The "before resonance part" corresponds to small velocity of movement and to a large length of roughness. In this case the vehicle body practically copies road profile, the difference between the value of active forces and static values is small.

A considerable increase of the output value relative to the irregularity height is a characteristic for the low frequency part. The amplitude value of the sprung mass oscillation increases and acceleration increases too.

During the "high frequency resonance" part, the sprung mass has very small oscillations but the accelerations are considerably high.

At the "between resonances" and "after resonance" parts the relative displacement and acceleration reduce together. The after resonance oscillations transform into the vibrations and acoustic oscillations.

2.4 Size Classification of Mobile Robotic System with Video Sensors

If an MRS is equipped with an onboard TV control system then the flow of useful information of the environment increases. Great attention is devoted to the quality of video information. It is a measure of the task execution success for an MRS during operation.

Rogozinnikov A. [1996] searched the questions of driving a telecontrolled MRS in detail. He investigated the problem of excluding the "blur effect" for a video image received from the onboard TV camera. He has offered a description of flutter (instead of oscillations) in the spring systems of a ground transport vehicle. According to this size classification of for telecontrolled vehicle looks as follows,

- large size, type of "DOROZHNIK" (weight more then 10000 kg),
- middle size, type of "STR" VNIITRANSMASH (weight more then 1000 kg, but less then 10000 kg),
- small size, type of "RAZVEDCHIK" CNIIRTK (weight more then 100 kg, but less then 1000 kg),
- diminutive size, type of "OMAR" SRISM MSTU (weight less then 100 kg).

This division is considerable, and some additional analysis is necessary for a detailed study of vehicles, but this classification opens the principal question of the possibility of application of video sensor systems when using a chassis within a determined group.

Rogozinnikov A. [1996] determined that movement of a large and middle size MRS is described by one-signal model, which takes into account oscillations of vehicle's body as a function of road parameters. The MRS drives within the after flutter zone with the typical velocity (in a range 1-8 m/s). Practical realisation of the CCD vision system for such systems is possible only with the presence of technical tools for TV camera stabilisation.

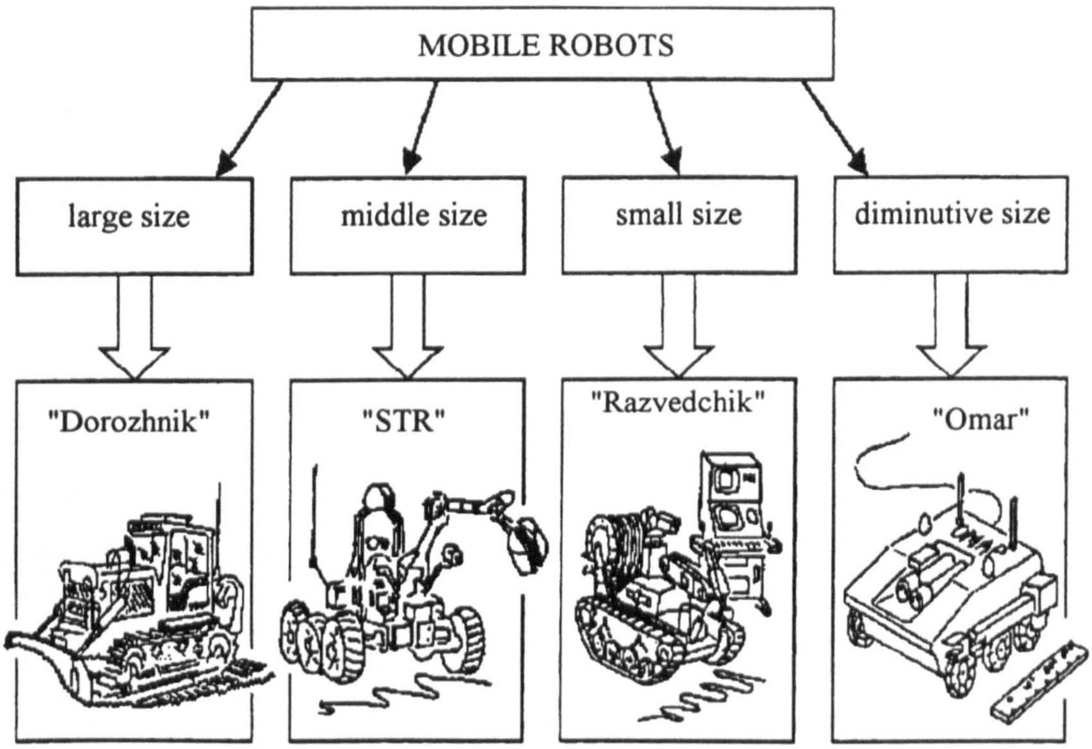


Figure 2.3 The Size Classification of Mobile Robotic Systems

On the contrary, according to *Rogozinnikov A.* [1996], to describe the movement of small and diminutive MRS, the classical one-signal model is not suitable. Consequently, it is necessary to form a mathematical model based on the methods developed by *Popov E.* [1982]. The reason being that the small and diminutive size of an MRS when in motion it is acted on by two frequent spectrums simultaneously - the road surface spectrum (as in the case of large and middle MRS) and the drives spectrum. These frequency spectrums have similar characteristics: practically the same amplitudes of peaks with small difference of the resonance frequencies. In these conditions, under determined velocities of the MRS motion (for "RAZVEDCHIK" MRS it is the range of velocity - 1-10 m/s) unsteady oscillations, pulsations (frequency domain of flutter) arise; due to *Timoshenko S.* [1959]. These oscillations are shown as a jump of supporting wheels and the rattle of a vehicle's body.

Rogozinnikov A. [1996] writes that stabilisation of a video sensor unsteady oscillations, (especially with the application of TV cameras with high resolution) in the flutter zone, for a small and diminutive MRS is a very difficult technical task. The reason being that the frequency domain of the flutter zone for an MRS is most wide, for example the "RAZVEDCHIK" type MRS has the domain rated from 30 to 300 Hz when the speed range is 1-10 m/s (see Figures 2.1, 2.3).

2.5 Kinematic Model of Mobile Robotic System

The complexity of analytical description (creation of the mathematical model) for an MRS when driving on a road is explained by two factors. The first, in the common case, a road profile is a random function. The second, an MRS is a multimass poly-resonance dynamic system with many degrees of freedom.

Traditionally, according to *Wong J.Y.* [1982] and *Smirnov G.* [1990] a transport vehicle is described by a simplified model with a limited number of frequencies. A model of the two mass system widely spreads for the transport vehicles description. Each mass of the pitch and roll banks of an MRS is described separately. The road profile can be represented by two means: a piecewise-analytical function or by a correlation function and spectral density.

An algebraic relation can be determined to calculate the forced oscillations of an MRS using the analytically derived disturbances. As a rule, it is described with a standard transfer function, *Popov E.* [1978] and *Popov E.* [1979].

To represent the external actions from the surface of the road as a correlation function and a spectral density (that is random magnitudes) the methods of statistical dynamics are used. It can be proved in the statistical dynamics theory that the application of transfer functions permits the calculation of the statistical characteristics of reaction for an oscillating system, i.e. to determine the characteristics of the MRS motion as that of a random process.

The degree of complexity of the developed mathematical model (or degree of description detail) for the physical process is determined with the requirements to the task set; according to *Chuyan R.* [1988]. The following assumptions are well accepted in relation to the task statement as described by equation 2.1; according to ground transport vehicles movement theory data analysis.

- The random microprofile of the road can be represented by a finite sum of the Fourier-series expansion components with sufficient accuracy (*Romanovsky P.* [1980]);

- Where an MRS is a telecontrolled vehicle of middle size (see Figure 2.3). Then the MRS motion is realised in the after flutter zone of frequency and the one-signal model may be used to describe it;
- The MRS velocity is considered to be small or medium (V=1-8 m/s). Such velocities are typical for this class of vehicles.

Besides, additional assumptions are required to be introduced, that are not obvious at this stage of research.

- It is admissible to use the one-signal polyharmonic model of the equivalent harmonic action at the stage of qualitative analysis of the ground MRS or at stage of the forming the external disturbance for the power calculation for scanning system actuators.
- The coefficients of equivalent harmonic action are fully determined by the MRS length of base, velocity and integral characteristics of microprofile for the given group of roads.

The model of MRS movement on an arbitrary road, at the before resonance zone of frequencies (see Figure 2.1 and part 1 at Figure 2.2) is considered. The initial random profile of the road $y_t(x_t)$ at the given interval is represented with the series:

$$y_t(x_t) = \sum_{i=1}^{n} A_i \sin(w_i x_t), - \qquad (2.2)$$

then the i-th component of pitch bank angle for MRS during disturbance from a side of i-th harmonic of road profile is equal to:

$$\alpha_i(t) = \arctan(A_i w_i \cos(w_i x_t)). \qquad (2.3)$$

The correctness of equation 2.3 can be easy proved. Really, when the streamline flow (copying) of smooth road profile the current value of pitch bank angle of the MRS is equal to the angle of tangent inclination to a road profile in the current point of road (see Figure 2.1). The tangent of the inclination angle is the partial derivative of the road profile with respect to x_t in the current point.

$$\tan(\alpha(x_t)) = \frac{\partial y_t(x_t)}{\partial x_t} = \sum_{i=1}^{n} A_i w_i \cos(w_i x_t) \qquad (2.4)$$

It is the required result (compare equation 2.3 with 2.4).

It is possible to simplify the equation 2.3 essentially for the MRS. The known expansion to power series of inverse trigonometric function arctan(γ) on interval $|\gamma| \le 1$ may be used.

$$\arctan(\gamma) = \gamma - \frac{\gamma^3}{3} + \frac{\gamma^5}{5} - \frac{\gamma^7}{7} + \ldots \qquad (2.5)$$

The first component of expansion has been taken into account. Thus it the equation 2.3 can be transformed into the following form:

$$\alpha_i(t) \approx A_i w_i \cos(w_i x_t), - \qquad (2.6)$$

The approximate equality sign may be omitted to simplify the description if it has no influence on the accuracy.

The limits of the applicability equation 2.6 are shown below. Values of relative error for the used approximation are illustrated with the next table.

γ	± 0.17	± 0.40	± 0.56	± 0.71	± 0.83	± 0.94	± 1.05
$1 - \frac{\gamma}{\arctan(\gamma)} \times 100\%$	-1	-5	-10	-15	-20	-25	-30

Table 2.1 Relative Error for the Used Approximation

According to Table 2.1, equation 2.6 gives overstated values of angle α component in relation to the real values. The error degree of the calculated data in relation to the actual data is determined with the absolute value of the relative error value and in practice for the i-th harmonic of input signal it does not exceed 10-15% (it corresponds 27 -31^0 of oscillations amplitude of the bank angles for MRS).

Taking into consideration that an MRS moves on a given road uniformly, linearly with V speed (see Figure 2.1), the equation 2.6 can be transformed into the time domain as shown in equation 2.7.

$$\alpha_i = A_i w_i \cos(w_i V t),- \qquad \textbf{(2.7)}$$

where V is linear velocity of the MRS motion;

t is current time.

The equation 2.7 may be used to estimate the contribution of i-th harmonic of the road profile into the required power of the MRS scanning system drives. According to *Podobry G., et al*

[1969] an effective power of the scanning system separate drive is estimated by the following equation:

$$P_{ef_i} = J_{ef}\frac{\partial\alpha}{\partial t}\frac{\partial^2\alpha}{\partial t^2},- \tag{2.8}$$

where J_{ef} is the effective inertia moment of the moving masses of the stabiliser in relation to the output shift of the stabilisation drive.

Taking into consideration the equation 2.7, the value of the effective power component to compensate the i-th harmonic of input signal is determined with the follow expression:

$$P_{ef_i} = 0.5J_{ef}A_i^2V^3w_i^5\sin(2w_iVt),- \tag{2.9}$$

It is obvious that the maximum value of the effective power component for stabilisation is equal to:

$$P_{ef_i}^{max} = 0.5J_{ef}A_i^2V^3w_i^5,- \tag{2.10}$$

The received equation 2.10 shows an important peculiarity of ground transport vehicles. It is the presence of a strong dependence for the required power of the onboard stabilisation system as a function of the vehicle linear velocity. The equation 2.10 shows cubic dependence of the required power for stabilisation in relation to velocity of an MRS.

In addition, further analysis of the equation 2.10 permits one to define another important parameter. The required stabilisation power depends quadratically on the spectrum harmonic amplitude of the road profile and on the fifth degree from cyclic frequency of this harmonic. This means that frequency dependence is considerably more strong. At the same time, amplitudes of the higher frequency harmonics are always smaller (see *Smirnov G.* [1990] and Application A). As the maximum values of disturbance parameters are of interest for this research (see the statement of task), hence, this singularity of MRS movement presents the possibility to exclude from consideration the low and medium frequency spectrum domain of the road microprofile.

The following consideration is made. The "before resonance" frequency domain of the AFC was considered above. Without loss of generality of the derived results, it is possible to extend the considered frequency domain. Consider some "equivalent" road represented according to the full mathematical dynamic model or from the results of field experiments of the MRS chassis movement. Such a "equivalent" road can provide ground conditions considered as an initial road. Further, such "equivalent" road has been considered in this work.

The digital method of restoring of an "equivalent" road is developed below. Consider the experimental data of the movement of an MRS travelling with constant velocity – V, on some random microprofile $y_t(x_t)$ as a function of time:

$$\alpha_{exp}(t),\ \dot{\alpha}_{exp}(t),\ \ddot{\alpha}_{exp}(t),\ \Delta t,\ 2\Delta t,... \qquad \textbf{(2.11)}$$

With supposition of angle smallness $\tan\alpha$ is equal to α and the following algorithm for creating an "equivalent" road may be implemented.

$$\left.\begin{aligned}&\Delta S = V\Delta t,\\&x(t+\Delta t) = x(t) + \Delta S,\\&y(t+\Delta t) = y(t) + \dot{\alpha}_{\exp} \times \Delta S + \frac{\dot{\alpha}_{\exp}(\Delta S)^2}{2V} +\\&\frac{(\Delta S)^3}{3V^2}(\ddot{\alpha}_{\exp} + 2\alpha_{\exp}(\dot{\alpha}_{\exp})^2).\end{aligned}\right\} \quad (2.12)$$

The "equivalent" road can be decomposed in series by analogy with that of the real road (see equation 2.12). That is why all previous received results are correct.

The transport subsystem of an MRS is a resonance system, which intensifies oscillations of one or several harmonics of the input signal spectrum.

As a result of the performed analysis, it is possible to write out the expression for calculation of the effective power during MRS movement in the following form:

$$P_{ef} \approx \sum_{j=1}^{k} P_{ef\,j}\,, - \quad (2.13)$$

where $k \langle\langle\ n$ (compare with equation 2.2). In practice, k is equal to 2. It follows from consideration of Figure 2.2 and taking into consideration that $P_{ef\,j}$, for engineering computations the following expression can be accepted.

$$P_{ef} \approx 0.5 J_{ef} A_i^2 V^3 w^5 \sin(2wVt), - \quad (2.14)$$

The value of spectrum harmonic expresses the frequency of input action, which corresponds to the highest resonance for the "equivalent" road for the MRS. In this case, it is not correct to consider a value of resonance harmonic as an amplitude in equation 2.14, as the expansion of the frequency domain does not permit such. The approach for calculation for "A" parameter in equation 2.14 is shown below.

For clarity of received results the following substitution is used:

$$w = \frac{2\pi}{mL_{MRS}}. \qquad \textbf{(2.15)}$$

This substitution expresses the normalisation of the resonance harmonic for an "equivalent" road in relation to the length of an MRS base.

The mathematical model of the MRS motion is received using the transformation of the equation 2.14 to the form of equation 2.7, taking into account equation 2.15.

$$\alpha(t) = \frac{2\pi}{mL_{MRS}} A\cos(\frac{2\pi}{mL_{MRS}})Vt. \qquad \textbf{(2.16)}$$

The advantage of the received kinematic model of the MRS is its simplicity. It determines the approach of external disturbance on the scanning system from the MRS, when the MRS is moving on a road with a random profile. A maximum energetic power can be determined, it is necessary on the power calculation stage. The real dynamics of MRS is actually taken into account with restoring the "equivalent" road.

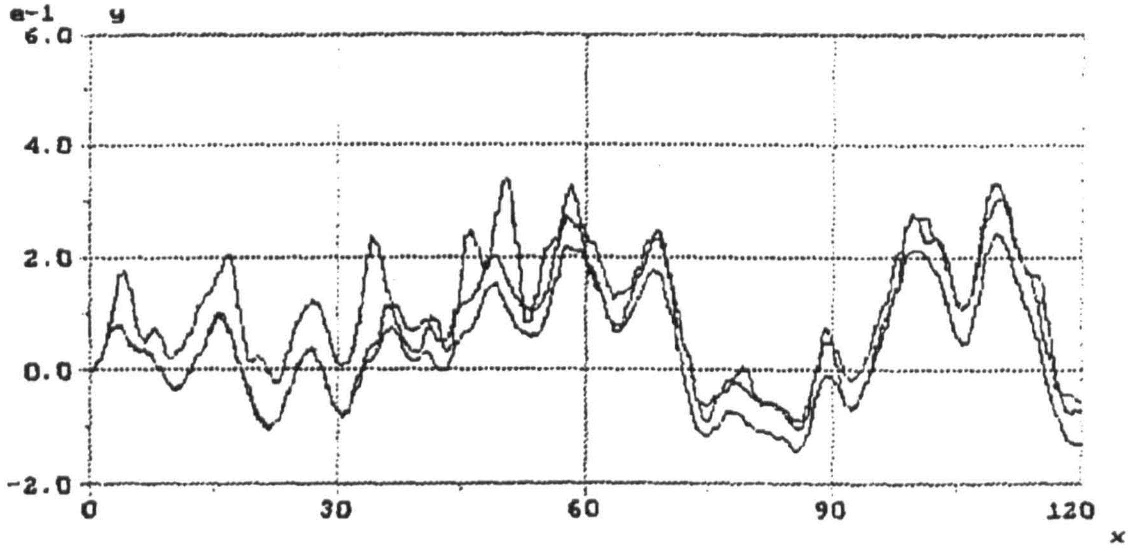


Figure 2.4 Restored Road

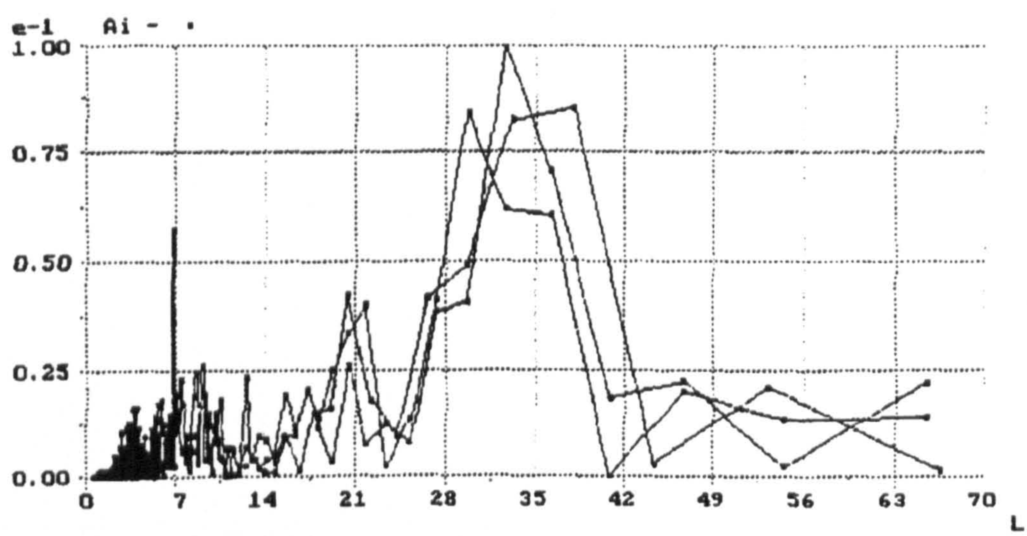


Figure 2.5 Parts of Spectrum for the Restored "Equivalent" Roads for Harmonics with L Lengths from 0 to 70 Meters of Wave

The model is not complete at this stage. It is necessary to define a method for calculation of the "A" parameter in equation 2.16.

2.6 IDENTIFICATION OF MODEL FOR MOVEMENT OF MOBILE ROBOTIC SYSTEM

The task of including the experimental data into the model arises during the process of mathematical simulation system design. Results of digital experiments represent a set of magnitudes that are sufficiently representative of experimental data from the point of view of further utilisation. The more effective mean for representation of the experimental results and digital calculation in mathematical simulation systems is the creation of the identification model.

The task of identification is the creation of an object model according to the search results of the object reaction to disturbance of the environment.

The mathematical task of identification is formulated in the following form. There are r experiments, for each of them vectors of the input parameters "x_r" is determined and W_r the output function of the system is known (is calculated or is measured). It is required to create the dependence (model), which describes the characteristics of the system to be investigated.

$$W=F(x). \qquad \textbf{(2.17)}$$

The equation 2.17 is an equation of regression. It can be considered that to have experimental data will be enough to create a mathematical description of an object (*Chuyan R.* [1988].; *Astapov U., Medvedev V.* [1982]).

Due to the supposition mentioned above, the mathematical model of the MRS movement may be described by the equation 2.16, this is the given equation of regression and the task of identification is reduced to the determination of the coefficients for this equation. But the form of mathematical model must be proved.

To prove the regularity and to identify the chosen mathematical model of the MRS motion a set of data in the form of expression 2.11 is used. The results of experiments with the mathematical dynamic model of the MRS transport subsystem when the MRS drives on a bumpy road with a random microprofile are discussed below. [Main characteristics of the MRS chassis are: mass 3500 kg, moment of inertia in relatively the centre of mass- 5950 kg $\times$ m^2, L_{MRS}-3.4 m. The experiment was carried out with speeds of MRS- V= {2, 4, 8} m/s.]

The restored part of the "equivalent" roads within the interval 0, 120 m, and the MRS velocities V= {2, 4, 8} m/s, are shown in Figure 2.4. These parts of road are regenerated according to algorithm described in 2.12 with the use of experimental data. The full length of road for the digital simulation was equal to 245 m. Analysis of Figure 2.4 shows that the coincidence of restored roads is acceptable at the chosen linear velocities of MRS. This permits one to conclude that the selected structure of the model (equation of regression) is correct. Actually, the offered method is used to take into consideration the vehicle velocity with regeneration of the road according to the algorithm described in 2.12. Obviously if it was not correct then the correlation of the derived results would not be so high.

Parts of spectrums for the restored "equivalent" roads are shown at Figure 2.5 for the harmonics with lengths of wave from 0 to 70 m. The Fourier series expansion for odd functions is applied for creating the spectrums. It is set on interval 0, 245 m. The used digital algorithm for interpolation of the function with Fourier series is shown.

Each interpolated road y(x) is set with the number of discrete points (y_i, x_i) where $x_i = \{0,\ \Delta x,\ \Delta 2x,\ ...,\ 245\}$. In this statement of the task, the algorithm of the Fourier series expansion with all points of the road is represented in the next form:

$$\left.\begin{aligned} y_i^N(x) &= \sum_{i=1}^{N-1} A_i \sin(\frac{2\pi}{L_i}x), \\ A_i &= \frac{2\Delta x}{iL_i}\sum_{j=1}^{N-1} y(x_j)\sin(\frac{2\pi}{L_i}x_j) + y(x_{j+1})\sin(\frac{2\pi}{L_i}x_{j+1}), \\ L_i &= \frac{2\times 245}{i}. \end{aligned}\right\} \qquad (2.18)$$

The relative error of offered approximation does not exceed 0.8%. The formula for calculation (on method of least squares) is shown below:

$$\varepsilon = \frac{100}{N-1}\left[\sum_{i=2}^{N}(1-\frac{y_i^N(x_i)}{y(x_i)})^2\right]^{1/2} [\%] \qquad (2.19)$$

The received results are analysed. High correlation of frequency spectrums proves the correctness for the offered equation of regression. The form of spectrums shows the presence of two resonance in the MRS transport subsystem. The main result is that the location of higher resonance (narrow peak on L_p= 6.8 m) does not depend on the MRS velocity. It allows one to identify the "m" parameter in the MRS motion model (see equations 2.15 and 2.16):

$$m = \frac{L_p}{L_{MRS}} = 2. \qquad \textbf{(2.20)}$$

Thus, the MRS amplifies irregularities of a random road if its length is equal to the twice the length of an MRS base. It does not depend on the amplitude of irregularities or speed of movement in the considered range.

The data presented in Figure 2.5 justifies the possibility not to take into consideration the polyharmonic model. Actually, using values of the parameters from Figure 2.5, it is possible to create the spectrogram of contribution for each harmonic, in effect the power of stabilisation for the scanning system drive (see equation 2.10) in units of abscissa axis. This spectrum is not shown here as actually it consists of one ordinate for harmonic with wave length L_p and another component: $P_{ef_i}^{max}(L_i \neq L_p) \to 0$ are negligibly small.

Actually, the received result shows the wave nature of oscillations the ground transport vehicles. When (a vehicle moves) and the harmonics of the road profile are equal to a wave length twice the size of the vehicle in this direction, then the contribution will be greatest. Namely these harmonics force maximum oscillations of the sprung mass. The harmonics with shorter wave length are extinguished by the spring system. This fact is known from the theory of wheel machines (*Smirnov G.* [1990]). Roughness with the wave length less than the maximum deflecting of the tyre has no influence on the oscillations of vehicle because of the absorbing ability of tyre. Moreover, it follows from this fact that vehicles with short wheel base oscillate strongly and wheel based MRS have essentially higher dynamics relative to track based MRS.

The data of Figure 2.4 and 2.5 justify the correctness of the received model of the MRS and to determine one of the regression coefficients – "m". To calculate another coefficient- "A" the identification methods for the over determination systems are applied.

On the base of a set of "equivalent" roads some average $y_{ave}(x)$ is defined.

$$y_{ave}(x) = \frac{1}{3}\sum_{i=1}^{3} y_i(x). \tag{2.21}$$

The received road may be represented by Fourier series for odd functions at the interval [0, 245] m. For this aim the (2.18) algorithm is used.

$$y_{ave}(x) \rightarrow y_1^N(x). \tag{2.22}$$

The reduction procedure (above and below) of the received series is performed.

$$y_1^N(x) \rightarrow y_M^k(x). \tag{2.23}$$

The relative error is estimated with the following formula (compare with (2.19)).

$$\varepsilon = \frac{100}{N-1}\left[\sum_{i=2}^{N}\left(1 - \frac{y_M^K(x_i)}{y_1^N(x_i)}\right)^2\right]^{1/2}, [\%]. \tag{2.24}$$

Where $y_M^k(x_i)$ is the value of the road ordinate with approximation of the reduced Fourier series: M>1, K<N. The combined analysis of the speed change and value of the relative approximation error can be taken as criteria for reducing (*Chuyan R.* [1988]).

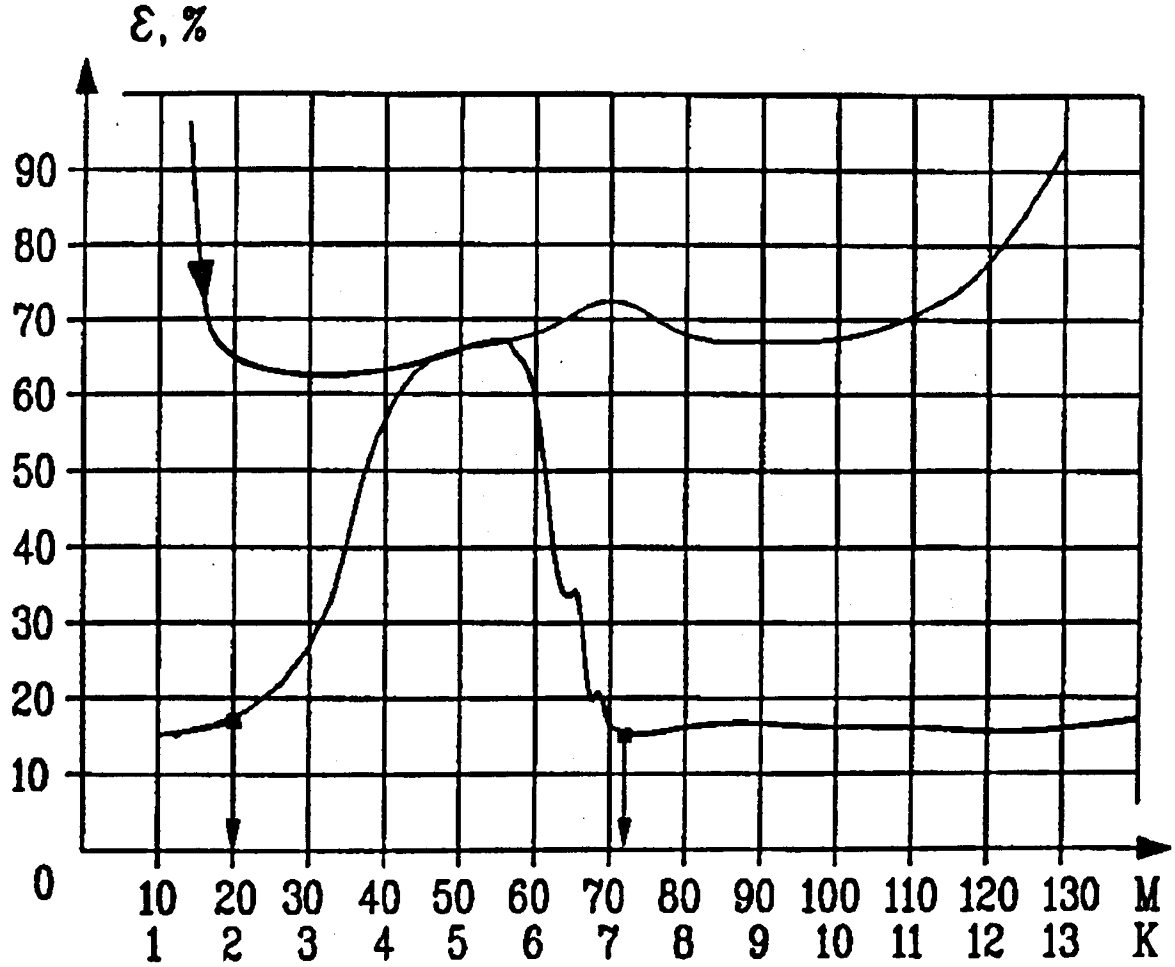


Figure 2.6 Graphs of Change $\varepsilon = \varepsilon(M)$ and $\varepsilon = \varepsilon(K)$

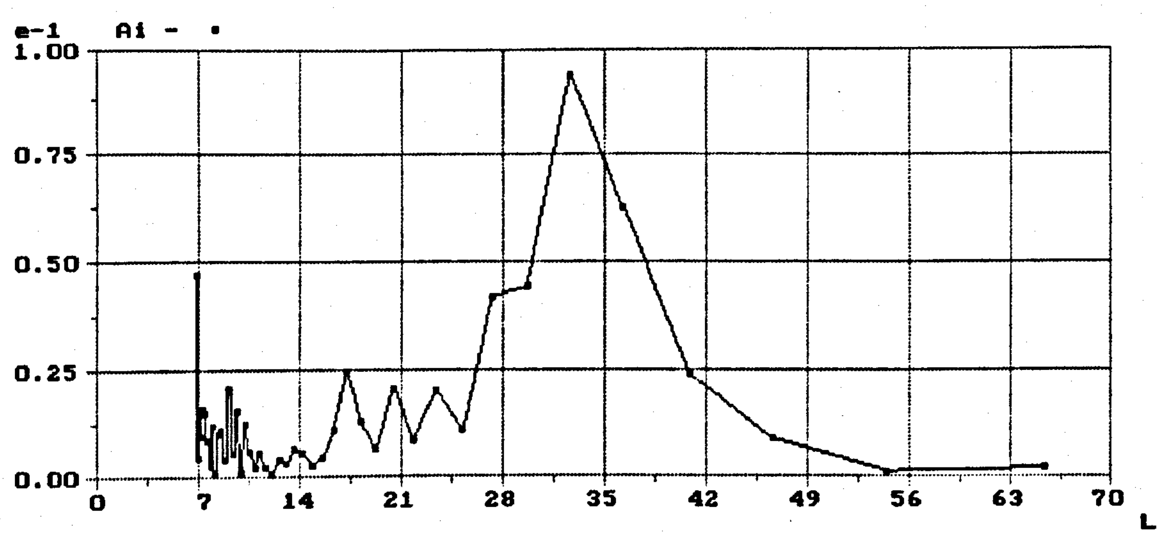


Figure 2.7 Spectrum of Resulting Road

The graphs of change $\varepsilon = \varepsilon(M)$ and $\varepsilon = \varepsilon(K)$ calculated by means of equation 2.24 are shown on Figure 2.6. Due to the received data it is clear that cutting off the high harmonics of the Fourier series up to K=72 has negligible influence on the approximation accuracy. The relative error increases to 14-16 % during reduction of the initial series. This value of error is admissible for engineering calculations. But further reduction of the approximation series increases the resulting error to 60-70 % (presence of speed extremum). It is shown at Figure 2.6. Thus the initial Fourier series (with the number of expansion set at N=1225) is limited by the 72-nd expansion set from the right with the approximation error ε(K=72)=14%. According to the third equation of 2.18 the 72-nd harmonic of the initial series is the harmonic of the higher resonance peak (L_p=6.8 m). The graph of change for the relative error proves the correctness of the offered model and the accuracy of the identification parameter m. Without taking into consideration some lower harmonics of the initial series the same situation is observed (see Figure 2.6). But, it is impossible to reduce the number of the expansion set without considerable loss of accuracy. In this case, the number of lower harmonics are reduced and the resulting relative error is ε(k=72, M=2)=16%.

For identification of the regression coefficient "A" in the equation 2.16 the following is offered. On the base of the received spectrum for the resulting road (see Figure 2.7) the parameter "A" is calculated with summation of the harmonic amplitude modules across the whole width of the spectrum.

$$A= \sum_{i=M}^{K} A_i = 0.63 \text{ (m)}. \qquad (2.25)$$

The offered method is based on the following fact. Research shows that the initial Fourier series $y_1^N(x)$ consists of a large number of mutually compensating harmonics and harmonics with the wave length shorter than that of the high resonance harmonics. These harmonics have insignificant influence on the resulting energetic of the MRS. Actually, the harmonics of the approximation series considered in equation 2.25 are located between resonance domain at the middle spectrum of the considered "equivalent" road. Thus, in the offered method of identification for the "A" regression coefficient, only that harmonics of the road microprofile are considered. Which cause oscillations of the sprung masses of the MRS during movement.

It is important that in practice parameter "A" can be calculated using another method instead of utilisation results of simulation and the mathematical dynamic model of the MRS chassis. The value of this regression coefficient can be determined by means of catalogue data or direct measurement for values of action intensity of microprofile (intensity of irregularities) for a determined class of road with taking into account the MRS base length.

The calculation accuracy of maximum power characteristics of the MRS oscillations is estimated when the MRS is driving on roads with random characteristics of microprofile. The data of digital simulation of the MRS transport subsystem is used for this aim.

The initial set of experimental data can be divided into two sequences the teaching and testing ones according to the known test criteria of the identification model accuracy *Chuyan R.* [1988]. The teaching sequence serves to determine the regression coefficients of the model. The testing sequence serves to verify the model.

The maximum values of speed modules for the MRS pitch angle obtained by means of experiment with use the mathematical dynamic model of the MRS chassis and calculated according to the offered method for different speed values of MRS movement on the road with random microprofile have been compared. The results of the estimation are shown on Table 2.2.

V, [m/s]	2	4	8
$\dot{\alpha}_{max}$, [1/s]	1.08	2.15	4.30
$\dot{\alpha}_{max}^{exp}$, [1/s]	0.95	1.96	3.68
ε , %	13	10	17

Table 2.2 Comparison of Maximum Rotary Velocities Received by Different Means

The offered method gives the overstated values of energetic limit relative to real values. The reason is the offered approximation equation 2.6. It is possible to increase accuracy considerably with the use of a larger number within the expansion set. But in practice the received accuracy is tolerable enough. As is shown in Table 2.2 the relative error of calculation for speed is 10-17 %. It is important that calculated values are greater than real ones always. The same situation is with the calculation of accelerations.

According to the mutual analysis of Table 2.1 and Table 2.2, it is possible to determine the expected accuracy of the calculated data using the following method. It is possible to define amplitude coefficients in the model expressed in 2.16 knowing the regression coefficients. After that due to the received results and data from Table2.1, it is possible to estimate the expected error of calculations. It is easy to show that this method gives high accuracy for small and middle velocities of MRS movement. The real relative error of calculation for large velocities of MRS exceeds calculated error.

2.7 Analysis of the Movement Model for a Mobile Robotic System. The Peculiarities of Ground Mobile Systems in Effecting Disturbance on Scanning Systems of CCD Vision System.

The developed mathematical model of MRS movement permits one not only to form the initial data for the stage of power calculation of the scanning system drives. Moreover, by means of this method it is possible to carry out the analysis of behaviour of a telecontrolled mobile system to estimate the effects of external disturbances on the mechanism of the video image stabilisation system.

To eliminate the "blur effect" of a TV image during an MRS movement it is necessary to compensate the forced oscillations of the transport vehicle body. The maximum effective power of the direct stabilisation drive, which is necessary for such compensation, is determined by the following expression:

$$P_{max}^{ef} = 0.5 J_{ef} A^2 V^3 \left[\frac{2\pi}{m L_{MRS}} \right], \quad (2.26)$$

where

J_{ef} is the reduced to output axis of stabilisation drive the summary inertia moment of the movable parts of mechanism, gearbox and motor;

A, m are the regression coefficients, constant of the current road conditions and the MRS chassis;

V is the linear velocity of MRS movement;

L_{MRS} is the length of supporting surface for a track MRS or length of the tyre impress for a wheel vehicle.

According to consideration of equation 2.26, ground transport vehicles can be considered as objects with high dynamic parameters. Also it indicates strong requirement for the drives of the stabilisation system. The stabilisation power is dependent on the velocity of the vehicle motion in the third degree. The influence of the amplitude irregularities (coefficient "A") is smaller than dependence upon the vehicle velocity (only quadratic dependence), especially when taking into consideration that in practice (due to *Smirnov G.* [1990], application 1) the maximum value of typical microprofiles differs insignificantly for a determined type of road. The road irregularities and wave length, with respect to the length of the vehicle supporting ground area in the direction movement have a greater influence on the required characteristics of the drive for the oscillation stabilisation of the transport vehicle chassis. According to equation 2.26, it influences in the fifth degree the required drive power.

Moreover, in practice the length of supporting ground area exactly determines the required property of frequency for the stabilisation drives. It follows from the equation 2.16, which describes the mathematical model of an MRS. In the same way, as it follows from equation 2.16 a wheeled vehicle has considerably higher frequency spectrum relative to a track vehicle with the same mass-inertia characteristics. A direct drive stabilisation system requires compensation of external disturbances in the frequency domain up to 50 Hz for middle and big MRS according to the offered classification of MRS shown in Figure 2.3.

Higher requirements are suggested for the stabilisation drive for an MRS of small and diminutive carrying TV systems vehicles In this case, the highest border of required frequency domain is of the order of 300 Hz even for the track vehicles: after *Rogozinnikov A.* [1996].

Analysis carried out indicates that traditional self-aligning systems with constant current motors are hardly effective for application an MRS with CCD vision system scanning systems; according to *Kochergin V.* [1988]. They can not provide the required band width for stabilisation of the vehicle oscillations, especially in the case of application of the wheel vehicles and with short wheel base.

A secondary spring system for the video sensor or scanning system of an MRS CCD vision system is hardly effective in the high frequency domain of action (*Smirnov G.* [1990]). It can not provide reduction of oscillations of a vehicle body in the area of second (high frequency) resonance.

The implementation of the gyrostabilisation platforms for stabilisation has been shown not to be good. The mass-inertia and power characteristics of a vehicle are degraded according *Chelyshev V.* [1995].

Accordingly the two directions of the scanning system systems design for a telecontrolled MRS may be considered:

- providing the CCD vision system scanning systems with an active spring system in the scanning system base or the inside stabilisation system for an CCD vision system;

-providing the scanning system drive with a piezoelectric rotating motors. These motors can realise the controlled rotation with frequency of up to 300-400 Hz. It is enough for the orientation of TV camera in space and for stabilisation of video image with MRS motion.

2.8 SUMMARY

The following results are presented in this chapter.

1. The mathematical model of the transport subsystem for the telecontrolled MRS has been developed. On the base of experimental result validation of the model and identification of the regression coefficients are proved, the model accuracy is estimated.

2. An approach is offered for the initial data extracting for the power calculation stage of a scanning system drive using the developed model of MRS motion. Application of the offered approach with research of scanning systems power characteristics allows one to avoid creating a general mathematical dynamic model of the MRS transport subsystem. This considerably reduces the design time.

3. The analysis of the MRS mathematical model has been carried out. The main features of the ground mobile systems are determined together with action on the mechanisms of the CCD vision system scanning systems. The obtained estimation of different external disturbing factors allows one to define the limits of applicability and a technical solutions for realisation of the MRS CCD vision systems.

CHAPTER 3

CHOICE AND VERIFICATION OF THE KINEMATIC SCHEME FOR PROPOSED FAST SCANING SYSTEM

3. CHOICE AND VERIFICATION OF THE KINEMATIC SCHEME FOR PROPOSED FAST SCANING SYSTEM

In this chapter, the kinematic analysis of the coordination mechanism for an MRS scanning system is undertaken. The choice of kinematic scheme is justified and the initial data for the control algorithm synthesis is obtained.

The matrix approach of the kinematic analysis for open circuit is used with applying the matrix apparatus 4×4.

3.1 Methods of Kinematic Analysis for Joint Mechanisms

3.1.1 The Statement of Problem

MRS scanning systems belong to the mechanisms of space orientation for view-finder systems, which pertain to the group of joint space mechanisms with close kinematic circuit with lower kinematic pairs of the fifth class *Frolov K.* [1988]. Many mechanisms from different areas of technology belong to this group. Great attention is paid to these systems in the theory of mechanism by *Vorobiev E.* and *Dimentberg F.* [1991], *Artobolevsky I.* [1988].

The kinematic analysis task traditionally appears with the beginning of a new systems design. The research of operation and functioning conditions, displacements, velocities and accelerations to the goal point are studied. Requirements to the conducting mechanism of the system are set.

To formulate the requirements for the drives system, the kinematic and geometric requirements to the goal point motion, in an absolute coordinate system, are transformed into the requirements for displacements, velocities and accelerations of the kinematic pairs. To solve such a problem the inverse tasks for positioning, velocities and accelerations have to be solved *Frolov K.* [1988], *Fu K. S.* [1989]:

- the inverse task for position consists of the determination of the relative coordinates of the mechanism links with respect to the given position for a goal point (in number of cases whilst taking into consideration the orientation of the last link);

- the inverse task for velocities consists in definition of the required generalised velocities of the kinematic pairs with respect to the given velocities of the output link;

- in practice, the required generalised accelerations for the kinematic pairs are not often determined. Instead, the task of determination of the necessary drive moments is resolved to realise the required motion. It is the so called first task of dynamics. (It is obvious that these calculations can be performed with the knowledge of the geometric and inertial parameters of the mechanism only. In the first stage of design these parameters are unknown and are chosen on the basis of the experience of an engineer who has significant experience in the design of such devices.

The results of the analysis of the inverse tasks for positions and velocities are the basis for the synthesis of the kinematic control algorithms: the positioning control algorithms, and control algorithms on the basis of the velocity vector *Frolov K.* [1988]. The positioning control algorithms are synthesised to move a point to the given position. These algorithms work on the basis of the mechanism kinematic scheme. Control algorithms for the velocity vector are synthesised to give a goal point a given linear (rotary) velocity. These algorithms also work on the basis of the mechanism's kinematic scheme.

The direct kinematic task appears at the stage of the functioning test of the designed the system variant, when mathematical simulation is carried out. The resulting movement of a goal point is calculated from the known values of the mechanism generalised coordinates, velocities and accelerations.

The methods of analytical and spherical trigonometry and geometric method are traditionally based on theory of a small rotation of a rigid body, and applied for the analysis of view-finder systems. *Ishlinsky A.* [1963], *Kudrevich B.* [1963], *Rivkin S.* [1978], *Ovakimov A.* [1971/b] describe these methods. In practice, all these methods lead to bulky transformations, require complex geometrical structure and experience to chose the "spherical triangles" at auxiliary drawing and to determine the desired the expressions. Accordingly these methods are seldom applied to the considered group of mechanism design (*Vorobiev E.* and *Dimentberg F.* [1991]).

The methods based on the theory of limited rotations of rigid bodies which use Rodriga-Hamilton or Kelly-Klein parameters are promising for this area. Implementation of these parameters leads to the application of quaternions, which are expressed in the form of a hypercomplex number by means of the Rodriga-Hamilton parameters. For example, a sum of two finite rotations is represented in the form of two corresponding quaternion product.

Ishlinsky A. [1976] used the theory of finite rotations of rigid bodies using quaternions to solve task of geometry for the Cardano suspension. Quaternions and the Rodriga-Hamilton parameters are used in the applied gyroscope theory by *Zeldovich S.* and *Okon I.* [1974] and also for certain geometrical and kinematic tasks for inertial navigation consideration, according to *Storozhenko V.* and *Temchenko M.* [1971]. This method develops analytical calculations, that do not require any geometrical construction. The advantage of this method is the fact that these kinematic parameters do not degenerate with any configuration of the rigid body. Use of these parameters leads to a description of the rigid body movement by means of linear equations. It simplifies the solving of these tasks. But the final formulas for the coordinate transformation using Rodriga-Hamilton parameters are also bulky. The question of use of these formulas for the calculations needs in special consideration in each case according to *Rivkin S.* [1978].

The complexity of the calculations for the considered group of multilink joint mechanisms has required much development of computer based methods. At the present time there are several methods for the analytical decision of the stated task. Three of them are widely used. These methods are the vector method, the method of screws and the method of matrix. Moreover, there are program methods, *Peisach E.* [1981], which allow one to determine a solution for some frequent cases. But in such cases, the task of kinematic analysis cannot be solved without determination singularities, these are determined by means of the solution of the system analytical equations.

3.1.2 The Vector Method

The vector method is widely used in the classical theory of mechanisms. *Lebedev P.* [1982] and *Ovakimov A.* [1971/a] have used this method for open kinematic circuits. The known Rodriga-Hamilton formula for the finite rotation of rigid body in vector form is based on this method.

$$r_1 = r\cos(\varphi) + (1-\cos(\varphi))(e\,r)\,e + e \times r\sin(\varphi), \quad (3.1)$$

where r and r_1 are vectors connected with the rigid bodies before and after rotation; e is the basis vector of rotation axis; φ is the angle of rotation.

In practice, the application of the vector method has reduced the necessary number of calculations. But the development of fast computer technology diminished this advantage unimportant at the present time. For this reason, the classical vector method has been somewhat by superseded to the more universal matrix methods of kinematic analysis.

3.1.3 Method of Screws

The classical theory of screws has been created by Ball. R. *Dimentberg F.* [1978] has written the screw calculus has initially determined by Kotelnikov A. and Study E. He has developed this theory further.

According to *Vorobiev E.* and *Dimentberg F.* [1991], the screw calculus is a form to create the screw theory. It can describe the kinematics of a rigid body when subjected to space movement. In the screw calculus, screws are represented in the form of special complex (dual) vectors with the use of Clifford symbols, which are represented by means of three complex (dual) coordinates. Consequently and according to the Kotelnikov-Study principle of displacement, the screws algebra becomes analogous to the algebra of free vectors. A full analogy between the kinematics of a body with one stationary point and the kinematics of a free body follows from it. For example, the dual Rodriga formula has the following form (compare with equation 3.1):

$$R_1 = R\cos(\varphi) + (E\,R)E(1-\cos\Phi) + E\times R\sin(\Phi)\,. \tag{3.2}$$

By mean of the displacement principle it is possible to solve the task of movement for a rigid body system. The relative displacements of which are subordinate to the conditions of geometric constraints. Consequently, the possibility of special simplification and clearness arises during the solution of the movement of a space joint and other mechanisms. Accordingly, the screw calculus is widely used in the kinematics of space mechanisms (*Kulakov F.* [1980] and *Dimentberg F.* [1978]).

But this method has an essential disadvantage. The attempt to use the dual vector algebra in dynamics does not lead to the adequate simple relations in kinematics (statics). It is connected with the following fact. The dual operator in the screw dynamics equation of a rigid body, that links the kinematic and force screws is the so called binor, which can not be derived out of the real operator by means of substitution of the real values with the dual ones. Due to this reason, the common case, the displacement principle can not applied to dynamics, and the dynamic tasks have to be solved by means of the ordinary theory of screws.

3.1.4 Method of Matrix

At present time, the matrix method of the kinematics calculation for 4×4 matrix with real elements is regarded as "method of matrix" in application to the considering group of space mechanisms.

In practice there are many other matrix methods: such as the matrix method with dual elements, the vector-matrix method that uses 3×3 matrix with real elements and 6×6 screw affinor in the elementary theory of screws. Currently the method of 4×4 matrix is more wide spread in comparison with these methods. The method of 4×4 matrix being regarded as the "method of matrix".

An approach for the position definition for space mechanisms using a matrix with real elements in the common case is offered by *Moroshkin U.* [1952]. After that, *Denevit J., Hartenberg R. S.* [1955] and other scientists develop this approach effectively.

The main idea of this method is to determine the relative positions of the mechanism link by means of a bypass of the mechanism circuit by the coordinate trihedron. The main element of this bypass is the transformation of coordinate system, which realises the transformation from one link to the neighbouring link.

The idea of mapping three-dimensional Euclidean space to the four-dimensional space of homogeneous coordinates is the basis of this method. It is achieved by means of the substitution of two sequential transformations of the coordinate systems: (rotation and displacement) - with one linear transformation. It permits the description many geometric, kinematic and dynamic relations for complex space mechanisms, compactly *Popov E. et al* [1978]:

$$p_0 = T_{01}\, p_1 ,- \tag{3.3}$$

where p_0 and p_1 are the radius-vectors of some point in the "zero" and in the "first" coordinate systems respectively, in homogeneous coordinates; T_{01} is the 4×4 transfer matrix from "zero" to "first" coordinate system.

The radius vector of a point in homogeneous coordinates is determined out of the radius vector in Cartesian coordinates by means of elementary extension.

$$p = [r^T, 1]^T = [[x, y, z, 1]]^T \; . \tag{3.4}$$

The method of matrix is a universal method, which can be applied for calculation of any space mechanism, including the of orientation mechanisms for view-finder systems.

The important advantages of this method are the follow:

- simplicity in use and the geometric interpretation provides: directly - ease of the mathematical model creation and indirectly- justification of the correctness of the received results;
- the developed mathematical apparatus of the 4×4 dimension matrix permits the avoidance of direct differentiation, in procedures of calculation;
- the presence of a number of good structured algorithms, for kinematic and dynamic tasks (direct and inverse), that provide simplicity of programming.

Recently the disadvantage of this method was considered, some calculation redundancy, in particular, the necessity to operate with a great number of zero elements of matrix has identified.

But, at the present time, this fact is unimportant due to the ever enhancing performance of computers. Moreover, it has become clear that application of the matrix method, combined with program complexes of analytical transformation, such as ALCOR and REDUCE, provides compact analytical solutions suitable to the programming of difficult control algorithms on the onboard computer of an MRS according to (*Klimov D.* and *Rudenko V.* [1985], *Nesterov V.* [1991]).

3.2 Solution of Inverse Kinematics for Mobile Robotic System Scanning System

It is necessary to know the solution of the inverse kinematic task to analyse the traditional kinematic schemes of scanning systems with kinematic redundancy.

The solution method of the inverse kinematic task for a redundant scanning system is demonstrated in the following example. The kinematic model of a scanning system with three degrees of freedom (DOF) is shown in Figure 3.1. The scanning system has three rotary degrees with coordinates (angles of relative turn of the mechanism links): q_1, q_2, q_3, - where q_1 - is the angle of rotation, q_2 - is the angle of pin inclination, q3 - is the angle of position. A sensor device with view-find beam L is rigidly fixed at the joint of q_3 angle (it is not shown in the Figure 3.1). The absolute value of L is equal to dimension D. It is the distance up to the goal point O_4. H_p and L_p are actual sizes of the scanning system structure. $O_0X_0Y_0Z_0$ is a right Cartesian coordinate system. It is rigidly fixed at the scanning system base. The current position of the goal point O_4 in $O_0X_0Y_0Z_0$ is the set with the radius-vector - $r_4^0=[x_4^0, y_4^0, z_4^0]^T$.

Traditionally the inverse kinematic task is formulated in the following form: to determine $Q=[q_1, q_2, q_3, D]^T$ - a vector of coordinate for the scanning system using $r_4^0=[x_4^0, y_4^0, z_4^0]^T$ - a known vector of the Cartesian coordinate of the goal point at the current moment of time. The traditional interpretation of the problem has been extended. The following equation system is considered as a solution of the inverse kinematic task.

$$\left.\begin{aligned} Q &= f_q(r^0,...), \\ \dot{Q} &= f_{\dot{q}}(V^0,...), \\ \ddot{Q} &= f_{\ddot{q}}(A^0,...). \end{aligned}\right\}. \qquad (3.5)$$

where $V^0 = \dot{r}_4^0$ - is a linear velocity of the goal point in $O_0X_0Y_0Z_0$,

$A^0 = \ddot{r}_4^0$ - is a linear acceleration of the goal point in $O_0X_0Y_0Z_0$.

Besides, the analytical solution of (3.5) system must have a compact form. It is dictated by the requirement for practical realisation for the control algorithms in real time using the onboard computers.

The definition of generalised coordinates for the scanning system drive mechanism using the given position of the goal point is a difficult task. It requires one to solve a system of non-linear equations 3.5. There are a several methods (*Popov E. et el* [1978], *Artobolevsky I.* [1988], *Paul R.* [1972], *Paul R.* [1979], *Urevich E.* [1984], *Frolov K.* [1988]) available to solve this difficult task. All of them do not give automated algorithms and require an individual approach for every concrete kinematic scheme (*Popov E. et el* [1978]).

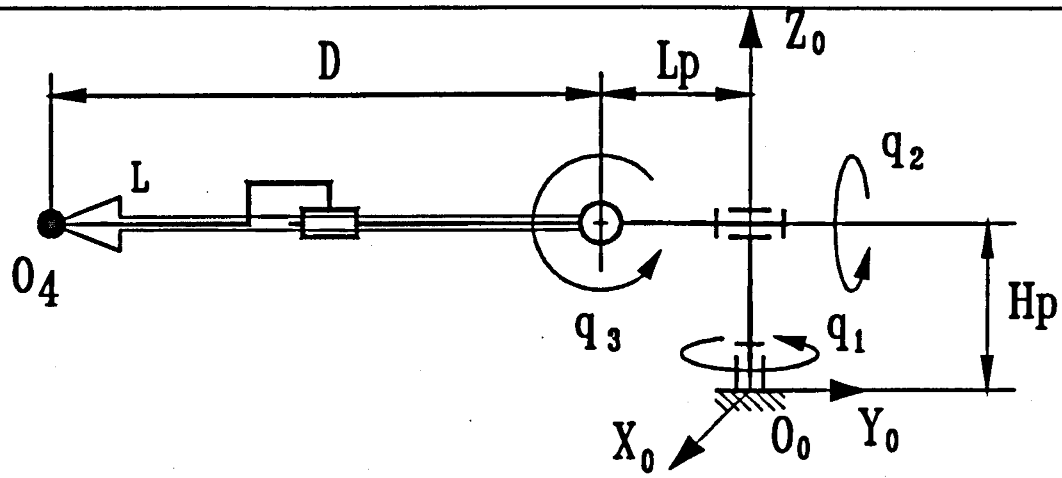


Figure 3.1 Kinematic Scheme of Scanning system

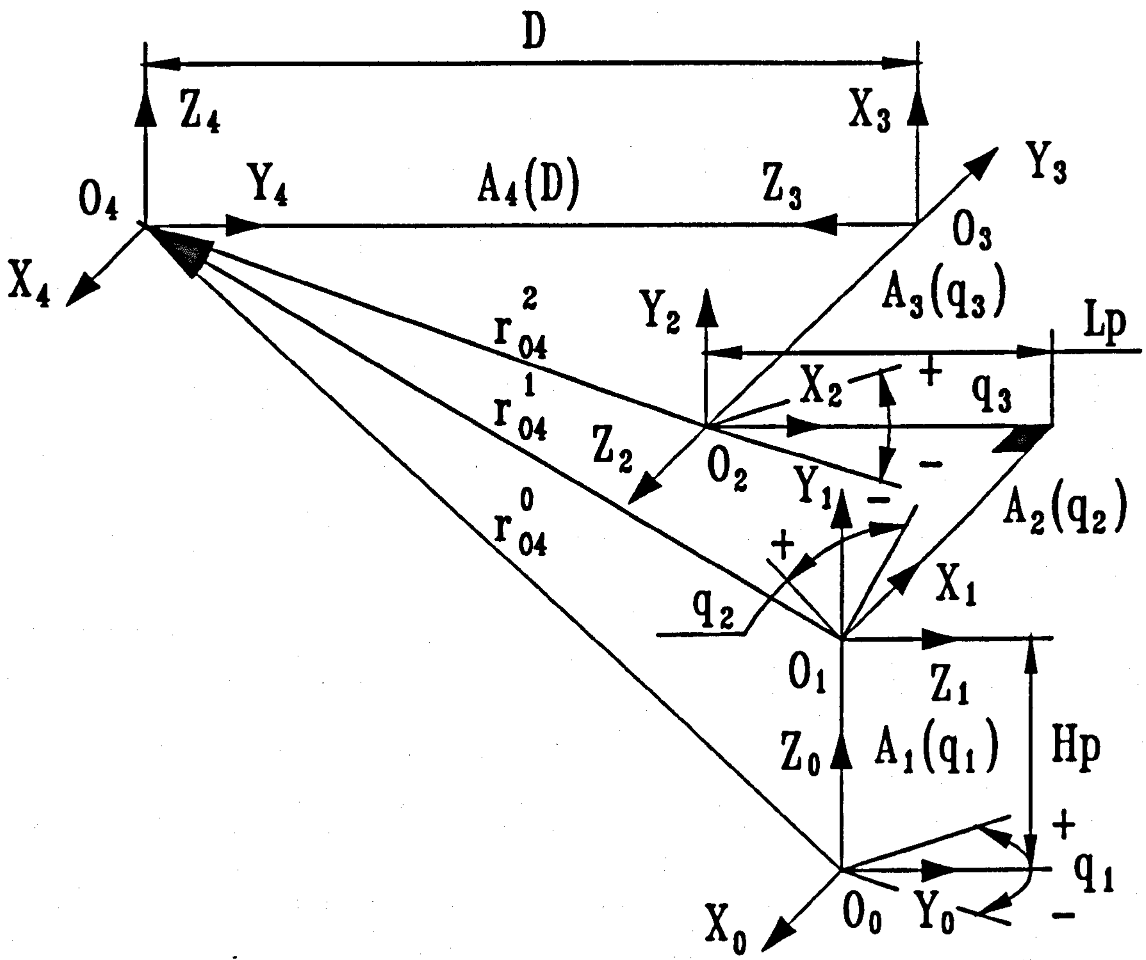


Figure3.2 Calculating Scheme of Scanning system

At the same time, according to *Kochergin V.* [1988] solutions of the inverse kinematic tasks obtained up to present time, for different kinematic schemes of scanning systems, do not include the class of kinematic redundant scanning systems where the number of simultaneously working drives is more than two. The resulting equations are solved practically only relative to values of the generalised coordinates and, besides, it is linearised according to supposition of the smallness of angles. A direct differentiation of these equations leads to an extraordinary bulky form for writing the solutions of inverse kinematic task in derivatives. In addition, accuracy of the solutions is reduced, as the linearised dependencies are considered to be the initial ones.

Another approach is shown in the papers *by Nesterov V.* [1991] and *Nesterov V.* [1994]. The author offers the approach to the automated deduction of the analytical solutions for the inverse kinematic tasks for positions, velocities and accelerations of the open kinematic loop of mechanisms for view-finder systems. The theory of 4×4 matrix, the method of indirect differentiation for the matrix equations with application of projection matrix, the idea of linear expansion for the Jacobian relative to the kinematically necessary and kinematically redundant generalised coordinates is used in this approach. It is shown, that the resulting equations are a function of the generalised coordinates and their derivatives with respective to time only. That is why it has a compact form and can be applied at the control algorithms level on of the onboard computers. It is important, also, that this approach does not limit the number of degrees of freedom. Therefore this method is suitable for the analytical description of systems with kinematic redundancy.

To solve the stated task an approach is used as described by *Nesterov V.* [1991] and *Nesterov V.* [1994]. It provides fulfilment of all requirements:

a) kinematically redundant schemes of the scanning system (see Figure 3.1);

b) analytical solutions in the form (3.5);

c) a compact form of writing of the resulting equations.

The bounded coordinate systems with the scanning system links are shown at Figure 3.2. $O_iX_iY_iZ_i$ (i=1, 2, 3, 4) are a right Cartesian coordinate system. The next step is to form the transfer matrix between the bounded coordinate systems of the mechanism' links.

Pantushin S. et al [1986] have shown that the composition of transfer matrixes for complicated space mechanisms with use of Hartenberg's parameters, for arbitrary circuits, is not convenient, and in some cases is impossible. Consequently, another method was applied in this thesis.

Six matrixes of elementary transformation of a right Coordinate System (CS) in terms of 4×4 matrix is introduced according to *Frolov K.* [1988].The transformation from CS $O_iX_iY_iZ_i$ to CS $O_{i+1}X_{i+1}Y_{i+1}Z_{i+1}$ is realised by means of the following transformations:

1. rotation on α relative to axis O_iX_i;

2. rotation on β relative to axis O_iY_i;

3. rotation on χ relative to axis O_iZ_i;

4. a shift in the value of "a" along axis O_iX_i;

5. a shift in the value of "b" along axis O_iY_i;

6. a shift in the value of "c" along axis O_iZ_i;

The corresponding 4×4 matrix of elementary transformation are designated as: $A_x^r(\alpha)$, $A_y^r(\beta)$, $A_z^r(\chi)$, $A_x^t(a)$, $A_y^t(b)$, $A_z^t(c)$.

The following designations for reduction of notation are introduced: E- identity matrix $C\alpha$ - cosine of α angle, $S\alpha$ - sine of α angle, 0 is null matrix.

$$A_x^r(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha & -S\alpha & 0 \\ 0 & S\alpha & C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A_x^t(\alpha) = \begin{bmatrix} & & & a \\ & E & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_y^r(\beta) = \begin{bmatrix} C\beta & 0 & S\beta & 0 \\ 0 & 1 & 0 & 0 \\ -S\beta & 0 & C\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A_y^t(\alpha) = \begin{bmatrix} & & & 0 \\ & E & & b \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \textbf{(3.6)}$$

$$A_x^r(\alpha) = \begin{bmatrix} C\chi & -S\chi & 0 & 0 \\ S\chi & C\chi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A_z^t(\alpha) = \begin{bmatrix} & & & 0 \\ & E & & 0 \\ & & & c \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The grouping rule has been formulated. A transfer matrix between two linked coordinate systems of a complicated spatial mechanism, is a product of any number of cofactors, each of them is one of the six matrices of elementary transformation equation 3.6. Thus a parameter of the first matrix of elementary transformation should be a generalised coordinate of mechanism (for rotary joints). The validity of this rule provides an arbitrary choice of linked coordinate systems of mechanism and it saves the possibility having to apply the approach of indirect differentiation for 4×4 matrix.

After these notations, the transfer matrixes of the scanning system can be written in the following form:

$$\left.\begin{aligned} A_1(q_1) &= A_z^r(q_1 + \pi/2)A_z^t(H_p)A_x^r(\pi/2)A_y^r(\pi/2), \\ A_2(q_2) &= A_z^r(q_2)A_y^r(-\pi/2)A_y^t(-L_p), \\ A_3(q_3) &= A_z^r(q_3 + \pi/2)A_x^r(-\pi/2), \\ A_4(D) &= A_z^r(-\pi/2)A_z^t(D)A_x^r(-\pi/2). \end{aligned}\right\} \qquad \textbf{(3.7)}$$

The following designations for reduction of notation are introduced: Cq_1 is the cosine of angle q_1, Sq_2 is the sine of angle q_2, Cq_2 is the cosine of angle q_2, Sq_3 is the sine of angle q_3, Cq_3 is the cosine of angle q_3, Sq_3 is the sine of angle q_3.

The following equations are obtained after product and substitution of known formulas (*Gantmacher F.* [1988], *Bronshtein I.* and *Semendyaev K.* [1988]).

$$\left.\begin{aligned}
A_1 &= \begin{vmatrix} -Cq_1 & 0 & -Sq_1 & 0 \\ -Sq_1 & 0 & Cq_1 & 0 \\ 0 & 1 & 0 & H_p \\ 0 & 0 & 0 & 1 \end{vmatrix}, \\
A_2 &= \begin{vmatrix} 0 & -Sq_2 & -Cq_2 & 0 \\ 0 & Cq_2 & -Sq_2 & 0 \\ 1 & 0 & 0 & -L_p \\ 0 & 0 & 0 & 1 \end{vmatrix}, \\
A_3 &= \begin{vmatrix} -Sq_3 & 0 & -Cq_3 & 0 \\ Cq_3 & 0 & -Sq_3 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}, \\
A_4 &= \begin{vmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & D \\ 0 & 0 & 0 & 1 \end{vmatrix}.
\end{aligned}\right\} \qquad \mathbf{(3.8)}$$

To solve the inverse kinematic task in position (the first expression in equation 3.5) the following approach described by *Popov E. et al* [1978] is applied. The main idea based on the following fact. The required expressions can be derived with consecutive transfer from the start link of the mechanism to the finish link of the mechanism by operation with transfer matrixes.

The following relation is considered as an initial premise (see Figure 3.2). Their expressions are known from the matrix theory of *Popov E. et al* [1978].

$$p_4^0 = A_1(q_1)A_2(q_2)A_3(q_3)A_4(D)p_4^4. \tag{3.9}$$

It is obviously, that p_4^4 and p_4^0 are known. The task is to define the unknown q_1, q_2, and q_3 and D. Both sides of equations 3.9 can be divided by $[A_1, A_2]^{-1}$.

$$[A_1]^{-1} [A_2]^{-1} p_4^0 = A_3 A_4 p_4^4. \tag{3.10}$$

Notation: the inversion of transfer matrixes 4 × 4 is realised according to the next rule.

$$\text{if } A=\begin{bmatrix} & [L] & & \vdots & r \\ \cdots\cdots & \cdots & \cdots & \vdots\cdots & \cdots \\ 0 & 0 & 0 & \vdots & 1 \end{bmatrix}, \text{ then } [A]^{-1}=\begin{bmatrix} & [L]^T & & \vdots & -[L]^T r \\ \cdots\cdots & \cdots & \cdots & \vdots\cdots & \cdots \\ 0 & 0 & 0 & \vdots & 1 \end{bmatrix}. \tag{3.11}$$

It is known from the theory of 4 × 4 matrix that the left and right expressions from equation 3.10 that equality is the radius vector of the goal point in the $O_2X_2Y_2Z_2$ coordinate system (see Figure 3.2) in homogeneous coordinates, that is:

$$p_4^2 = [A_2]^{-1}[A_1]^{-1}p_4^0 = A_3A_4p_4^4. \tag{3.12}$$

And it follows analogously:

$$p_4^1 = [A_1]^{-1} p_4^0 = A_2 A_3 A_4 p_4^4. \qquad (3.13)$$

The next expression is obtained with the uncovering of the right part of equation 3.13 at equality:

$$p_4^2 = [x_4^2, y_4^2, z_4^2, 1]^T = [-DCq_3, -DSq_3, 0, 1]^T. \qquad (3.14)$$

Here and below the geometric sizes of coordinates may be neglected to simplify calculations, that is $H_p=L_p=0$ (see Figure3.1 and Figure 3.2). In practice such an assumption is generally accepted, almost always, at the stage of the kinematic analysis for a view-finder systems (*Kochergin V.* [1988]).

The next equation has been obtained from the equation 3.14.

$$\left.\begin{aligned} \tan(q_3) &= \frac{y_4^2}{x_4^2}, \\ [D]^2 &= [x_4^2]^2 + [y_4^2]^2. \end{aligned}\right\} \qquad (3.15)$$

And analogously the following equation is determined from 3.13.

$$p_4^1 = [x_4^1, y_4^1, z_4^1, 1]^T = [DSq_2Sq_3, -DCq_2Sq_3, -DCq_3, 1]^T. \qquad (3.16)$$

And further:

$$\tan(q_2) = -\frac{y_4^1}{x_4^1}. \qquad (3.17)$$

The received expressions 3.15, 3.17 are the exact relations, which must be determined, but at this stage they are expressed by the components of the radius vector of the goal point in the linked coordinate systems $O_1X_1Y_1Z_1$ and $O_2X_2Y_2Z_2$. It is necessary to define the dependencies of the generalised coordinates, for the scanning system from the goal point radius vector components in the coordinate system linked with the base of scanning system (see Figure 3.1 and Figure 3.2).

The considered variant has kinematic redundancy, that is why only three generalised coordinates from q_1, q_2, q_3, D can be independent. In addition, distance D from the scanning system up to the goal point has to be one of them. This condition is determined by the possibility to close the kinematic loop (see Figure 3.2). The final solution of the inverse kinematic task has the following form:

$$\left.\begin{aligned} q_2 &= q_2(r_4^0, q_1), \\ q_3 &= q_3(r_4^0, q_1), \\ D &= D(r_4^0, q_1). \end{aligned}\right\}. \qquad (3.18)$$

Deficient expressions for 3.15 and 3.17 are deduced from the left parts of 3.8, 3.12 and 3.13, taking into account 3.11.

$$\left.\begin{aligned}&p_4^1 = [x_4^1, y_4^1, z_4^1, 1]^T = [A_1]^{-1} p_4^0 = \\ &[-x_4^0 Cq_1 - y_4^0 Sq_1, z_4^0, -x_4^0 Sq_1 + y_4^0 Cq_1, 1]^T, \\ &p_4^2 = [x_4^2, y_4^2, z_4^2, 1]^T = [A_2]^{-1}[A_1]^{-1} p_4^0 = [-x_4^0 Sq_1 + y_4^0 Cq_1, \\ &Sq_2(x_4^0 Cq_1 + y_4^0 Sq_1) + z_4^0 Cq_2, Cq_2(x_4^0 Cq_1 + y_4^0 Sq_1) - z_4^0 Sq_2, 1]^T.\end{aligned}\right\} \qquad (3.19)$$

The determination of the inverse kinematic task in positions for the analysing variant of the scanning system kinematic scheme is achieved by substituting the relation expressed in 3.19 in equations 3.17 and 3.15.

$$\left.\begin{aligned}&q_2 = \operatorname{arctg}\left(\frac{x_4^0 Cq_1 + y_4^0 Sq_1}{z_4^0}\right), \\ &q_3 = \operatorname{arctg}\left(\frac{Sq_2(x_4^0 Cq_1 + y_4^0 Sq_1) + z_4^0 Cq_2}{-x_4^0 Sq_1 + y_4^0 Cq_1}\right), \\ &D = \sqrt{[x_4^0]^2 + [y_4^0]^2 + [z_4^0]^2}.\end{aligned}\right\} \qquad (3.20)$$

To solve the inverse kinematic task for velocity and acceleration the approach offered by *Nesterov V.* [1991] and *Nesterov V.* [1994] has been used. The determination of the direct kinematic task in terms of four dimension matrix theory is the initial premise. According to this theory the following relation (decision of direct kinematic task) is correct for a scanning system kinematic scheme (see Figure 3.1) in the transfer from the three dimension Euclidean space, with respect to $O_0X_0Y_0Z_0$ axis, to four dimension space of homogeneous coordinates (see Figure 3.2).

$$\left.\begin{aligned} p_4^0 &= T_4 p_4^4, \\ \dot{p}_4^0 &= \dot{T}_4\, p_4^4, \\ \ddot{p}_4^0 &= \ddot{T}_4\, p_4^4. \end{aligned}\right\} \qquad (3.21)$$

where

$T_4=A_1A_2A_3A_4$ - is the matrix of position 4×4 for O_4 point in $O_0X_0Y_0Z_0$;

$\dot{T}_4$ and $\ddot{T}_4$ - are the respectively the first and the second derivations with respect to time of T_4 matrix;

$p_4^0=[r_4^0,\ 1]^T$ - is the radius vector in homogeneous coordinates of O_4 point in $O_0X_0Y_0Z_0$; $p_4^4=[r_4^4,\ 1]^T=[0, 0, 0, 1]^T$ - is the radius vector in homogeneous coordinates of O_4 point in $O_4X_4Y_4Z_4$.

The T_4 matrix for the scanning system's kinematic scheme and the transfer matrix 3.8 can be written in the form:

$$.T_4 = \begin{bmatrix} Cq_1Cq_2 & Cq_1Sq_2Sq_3 - Sq_1Cq_3 & Cq_1Sq_2Cq_3 + Sq_1Sq_3 & L_pSq_1 - D(Cq_1Sq_2Sq_3 - Sq_1Cq_3) \\ Sq_1Cq_2 & Sq_1Sq_2Sq_3 + Cq_1Cq_3 & Sq_1Sq_2Cq_3 - Cq_1Sq_3 & -L_pSq_1 - D(Sq_1Sq_2Sq_3 + Cq_1Cq_3) \\ -Sq_2 & Cq_2Sq_3 & Cq_2Cq_3 & -DCq_2Sq_3 + H_p \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.22)$$

The first and the second derivations with respect to time of the T_4 matrix can be determined with two different approaches: 1) direct differentiation and 2) with use of the method of indirect differentiation from 4×4 matrix theory. In practice, the second approach is often used. The main reason is that at the present time differentiation (especially double differentiation) expression of type shown in 3.22 is carried out in program complexes for analytical transformation. These complexes whilst not allowing differentiation of complex transcendental expressions do it permits the production of all elementary matrix transformations.

The next expression has been written using the means of indirect differentiation for 4×4 matrix.

$$\left.\begin{aligned}
&T_i=T_{i-1}A_i,\\
&\dot{T}_i=\dot{T}_{i-1}A_i+B_i\dot{q}_i,\\
&\ddot{T}_i=\ddot{T}_i A_i+B_i\ddot{q}_i+C_i,\\
&B_i=T_{i-1}Q_jA_i,\\
&C_i=2\dot{T}_{i-1}Q_jA_i\dot{q}_i+T_{i-1}[Q_i]^2A_i(\dot{q}_i)^2,\\
&T_0=E,\\
&\dot{T}_0=\ddot{T}_0=0,\\
&i=1,2,\ldots N+1,\ j=1,2,\ldots 6
\end{aligned}\right\}\qquad(3.23)$$

In addition, Q_i -are so called projection matrixes. In this algorithm for the calculation of the T_4 matrix with respect to time, the view of Q_i is dependent on the structure of the corresponding transfer matrix in 3.7. Q_j is equal to Q^1 in the case of rotation of angle α relative to the O_iX_i axis. Q_j is equal to Q^2 in the case of rotation of angle β relative to the O_iY_i axis. Q_j is equal to Q^3 in the case of rotation on angle χ relative to the O_iZ_i axis. Q_j is equal to Q^4 in case of transfer on "a" value along the O_iX_i axis. Q_j is equal to Q^5 in case of transfer on "b" value along the O_iY_i axis. Q_j is equal to Q^6 in case of transfer on "c" value along the O_iZ_i axis.

$$Q^1=\begin{bmatrix}0&0&0&0\\0&0&-1&0\\0&1&0&0\\0&0&0&0\end{bmatrix},\ Q^2=\begin{bmatrix}0&0&1&0\\0&0&0&0\\-1&0&0&0\\0&0&0&0\end{bmatrix},\ Q^3=\begin{bmatrix}0&-1&0&0\\1&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix},$$

$$Q^4=\begin{bmatrix}0&0&0&1\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix},\ Q^5=\begin{bmatrix}0&0&0&0\\0&0&0&1\\0&0&0&0\\0&0&0&0\end{bmatrix},\ Q^6=\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&1\\0&0&0&0\end{bmatrix}. \quad \textbf{(3.24)}$$

According to equation 3.7, the projection matrices for considering the kinematic scheme of the scanning system are equal to:

$$Q_1=Q_2=Q_3=Q^3,\ Q_4=Q^6. \quad \textbf{(3.25)}$$

The second equation of the system 3.21 has been considered to receive the analytical decision of inverse kinematic task in velocity. It is obviously that the equation for velocity can be rewritten, taking into account that $p_4^4=[0, 0, 0, 1]^T$, in the following form:

$$V^0=f_v[\dot{q}_1,\dot{q}_2,\dot{q}_3,\dot{D},\ldots], \quad \textbf{(3.26)}$$

where

$V^0=[V_x^0, V_y^0, V_z^0]=\dot{r}_4^0$ - is the vector of linear speed for the goal point (see Figure 3.1);

$f_v=[\dot{T}_4(1,4),\dot{T}_4(2,4),\dot{T}_4(3,4)]^T$ - is the vector of component for the right column of the $\dot{T}_4$ matrix.

equation 3.26 has been transformed to the standard form:

$$V^0=J[\dot{q}_1,\dot{q}_2,\dot{q}_3,\dot{D}]^T ,- \qquad (3.27)$$

J is Jacobi matrix 3×4 (Jacobian) of system with kinematic redundancy. The following designation has been introduced.

$$J=[j_1, j_2, j_3, j_4] ,- \qquad (3.28)$$

where j_i is i-th vector- column of Jacobi matrix, i= 1, 2, 3 ,4.

The linear expansion of the Jacobian in relative to both kinematic redundant vectors of velocity and kinetically necessary. Equation 3.27 can be rewritten in the following form:

$$V^0=M_v[\dot{q}_2,\dot{q}_3,\dot{D}] +Pv \qquad (3.29)$$

Where $M_v=[j_2, j_3, j_4]$; $Pv= j_1 \dot{q}_1$.

In equation 3.29 the generalised coordinate - $\dot{D}$ has to be chosen as a necessary coordinate in vector's component of speeds. This coordinate characterises the change velocity of current distance. In another case, M_v -3×3 matrix will be degenerate. It physically follows from the possibility the model chaining kinematically closed.

For the analysing variant of the scanning system the matrices - component for 3.29 has the following form:

$$M_v = \begin{bmatrix} -Cq_1Cq_2Sq_3D & -(Sq_1Sq_3 + Cq_1Sq_2Cq_3)D & Sq_1Cq_3 - Cq_1Sq_2Sq_3 \\ -Sq_1Cq_2Sq_3D & (Cq_1Sq_3 - Sq_1Sq_2Cq_3)D & -Cq_1Cq_3 - Sq_1Sq_2Sq_3 \\ Sq_2Sq_3D & -Cq_2Cq_3D & -Cq_2Sq_3 \end{bmatrix} \quad 3.30$$

$$P_v = \begin{bmatrix} D\dot{q}_1(Cq_1Cq_3 + Sq_1Sq_2Sq_3) \\ D\dot{q}_1(Sq_1Cq_3 - Cq_1Sq_2Sq_3) \\ 0 \end{bmatrix} \quad (3.31)$$

The desired analytical dependencies can be easy obtained from equation 3.29.

$$[\dot{q}_2, \dot{q}_3, \dot{D}]^T = [M_v]^{-1} (V_0 - P_v). \quad (3.32)$$

The 3 × 3 matrix M_v (see equation 3.30) has been converted:

$$M_v^{-1} = \begin{bmatrix} -\frac{Cq_1Cq_2}{Sq_3D} & -\frac{Sq_1Cq_2}{Sq_3D} & -\frac{Sq_2}{Sq_3D} \\ \frac{Sq_1Sq_3 + Cq_1Sq_2Sq_3}{D} & \frac{Cq_1Sq_3 - Sq_1Sq_2Cq_3}{D} & -\frac{Cq_2Cq_3}{Sq_3D} \\ Sq_1Cq_3 - Cq_1Sq_2Sq_3 & -Cq_1Cq_3 - Sq_1Sq_2Sq_3 & -Cq_2Sq_3 \end{bmatrix}, \quad (3.33)$$

The inverse kinematic task for velocities has been obtained from 3.32 with reduction of obvious relations like $[Cq_i]^2+[Sq_i]^2$, i=1, 2, 3.

$$\left.\begin{array}{l}\dot{q}_2 = \dfrac{-Cq_1Cq_2V_x^0 - Sq_1Cq_2V_y^0 + Sq_2V_z^0 + Cq_2Cq_3D\dot{q}_1}{DSq_3},\\ \dot{q}_3 = \dfrac{(-Sq_1Sq_3 - Cq_1Sq_2Cq_3)V_x^0 - Cq_2Cq_3V_z^0}{D} + \\ \dfrac{Sq_2D\dot{q} + (Cq_1Sq_3 - Sq_1Sq_2Cq_3)V_y^0}{D},\\ \dot{D} = (-Sq_1Cq_3 - Cq_1Sq_2Cq_3)V_x^0 - Cq_2Sq_3V_z^0 - \\ (Cq_1Cq_3 - Sq_1Sq_2Sq_3)V_y^0.\end{array}\right\} \quad (3.34)$$

The third equation of system 3.21 has been considered to obtain the result of the inverse kinematic task for accelerations. The equation for accelerations can be rewritten, with taking into account that p_4^4=[0, 0, 0, 1]^T, in the follow form:

$$A^0 = f_A[\ddot{q}_1, \ddot{q}_2, \ddot{q}_3, \ddot{D}, \ldots], - \quad (3.35)$$

where

$A^0=[A_x^0, A_y^0, A_z^0]=\ddot{r}_4^0$ - is a vector of linear acceleration for the goal point (see Figure 3.1);

$f_A=[\ddot{T}_4(1,4), \ddot{T}_4(2,4), \ddot{T}_4(3,4)]^T$ - is a vector of the component for the right column of the $\dot{T}_4$ matrix.

It is shown above that 3.35 can be rewritten in the following form:

$$A^0 = M_v[\ddot{q}_2, \ddot{q}_3, \ddot{D}] + P_A, - \quad (3.36)$$

where $P_A=P_A(\ddot{q}_1, \ldots, \ddot{q}_{N-2}, \dot{q}_1, \ldots, \dot{q}_{N-1}, \dot{D}, \ldots, q_N, D)$.

From 3.36, the desired analytical decision in matrix form is determined with the following equation, according to *Nesterov V.* [1991] and *Nesterov V.* [1994].

$$\ddot{Q}=[M_A]^{-1}(A^0-P_A),- \quad (3.37)$$

where $M_v=M_A$.

On this stage, the practical realisation of the approach offered by *Nesterov V.* [1991] and *Nesterov V.* [1994] shows that unacceptable errors, connected with shortage of memory, develops when solving equation 3.37 in the ALCOR system of version 1.1.

From equation 3.36, the value of P_A can been rewritten in the following form:

$$A^0=M_v[\ddot{q}_2,\ddot{q}_3,\ddot{D}]^T+\dot{M}_v[\dot{q}_2,\dot{q}_3,\dot{D}]^T+\dot{P}_v. \quad (3.38)$$

The desired solution in matrix form can be represented as:

$$[\ddot{q}_2,\ddot{q}_3,\ddot{D}]^T=M_v^{-1}(-(\dot{M}_v[\dot{q}_2,\dot{q}_3,\dot{D}]^T-\dot{P}_v)+A^0)=$$

$$=-M_v^{-1}(\dot{M}_v[\dot{q}_2,\dot{q}_3,\dot{D}]^T+\dot{P}_v)+M_v^{-1}A^0. \quad (3.39)$$

where 3.39 has been uncovered taking into account equations 3.33, 3.40 and 3.33. After reduction of obvious relations like $[Cq_i]^2+[Sq_i]^{2,}$ i=1, 2, 3, the result of the inverse kinematic task for accelerations has been received.

$$\left.\begin{aligned}
&\dot{P}_v(1,1) = D(\dot{q}_1)^2(Sq_1Cq_3 - Cq_1Sq_2Sq_3) + (Cq_1Sq_3 + Sq_1Sq_2Sq_3)D(\dot{q}_1 + \ddot{q}_1) + \\
&D\dot{q}_1\dot{q}_3(-Cq_1Sq_3 + Sq_1Sq_2Cq_3) + Sq_1Cq_2Sq_3D\dot{q}_1\dot{q}_2, \\
&\dot{P}_v(2,1) = D(\dot{q}_1)^2(Cq_1Cq_3 + Sq_1Sq_2Sq_3) + (Sq_1Cq_3 - Cq_1Sq_2Sq_3)D(\dot{q}_1 + \ddot{q}_1) - \\
&D\dot{q}_1\dot{q}_3(Sq_1Sq_3 + Cq_1Sq_2Cq_3) - Cq_1Cq_2Sq_3D\dot{q}_1\dot{q}_2, \\
&\dot{P}_v(3,1) = 0.
\end{aligned}\right\} \quad (3.40)$$

$$\left.\begin{aligned}
&\dot{M}_v(1,1) = \dot{D}Cq_1Cq_2Sq_3 - \dot{q}_3DCq_1Cq_2Cq_3 + \dot{q}_2DCq_1Sq_2Sq_3 + \\
&\dot{q}_1DSq_1Cq_2Sq_3, \\
&\dot{M}_v(1,2) = -\dot{D}(Sq_1Sq_3 + Cq_1Sq_2Cq_3) - \dot{q}_3D(Sq_1Cq_3 - Cq_1Sq_2Sq_3) - \\
&\dot{q}_2DCq_1Cq_2Cq_3 - \dot{q}_1(Cq_1Sq_3 - Sq_1Sq_2Cq_3), \\
&\dot{M}_v(1,3) = \dot{q}_1(Cq_1Cq_3 + Sq_1Sq_2Sq_3) - \dot{q}_3(Sq_1Sq_3 + Cq_1Sq_2Cq_3) - \\
&\dot{q}_2Cq_1Cq_2Sq_3, \\
&\dot{M}_v(2,1) = -\dot{D}Sq_1Cq_2Sq_3 - \dot{q}_3DSq_1Cq_2Cq_3 + \dot{q}_2DSq_1Sq_2Sq_3 - \\
&\dot{q}_1DCq_1Cq_2Sq_3, \\
&\dot{M}_v(2,2) = \dot{D}(Cq_1Sq_3 - Sq_1Sq_2Cq_3) + \dot{q}_3D(Cq_1Cq_3 + Sq_1Sq_2Sq_3) - \\
&\dot{q}_2DSq_1Cq_2Cq_3 - \dot{q}_1D(Sq_1Sq_3 - Cq_1Sq_2Cq_3), \\
&\dot{M}_v(2,3) = \dot{q}_1(Sq_1Cq_3 - Cq_1Sq_2Sq_3) + \dot{q}_3(Cq_1Sq_3 - Sq_1Sq_2Cq_3) - \\
&\dot{q}_2Sq_1Cq_2Sq_3, \\
&\dot{M}_v(3,1) = \dot{D}Sq_2Sq_3 + \dot{q}_2DCq_2Sq_3 + \dot{q}_3DSq_2Cq_3, \\
&\dot{M}_v(3,2) = -\dot{D}Cq_2Cq_3 + \dot{q}_2DSq_2Cq_3 + \dot{q}_3DCq_2Sq_3, \\
&\dot{M}_v(3,3) = \dot{q}_2Sq_2Sq_3 - \dot{q}_3Cq_2Cq_3.
\end{aligned}\right\} \quad (3.41)$$

$$\left.\begin{aligned}
&\ddot{q}_2 = \frac{1}{DSq_3}(-Cq_1Cq_2A_x^0 - Sq_1Cq_2A_y^0 + Sq_2A_z^0 + Cq_2Cq_3D\ddot{q}_1 + \\
&Cq_2Sq_2Sq_3D(\dot{q}_1)^2 + 2Cq_2Cq_3\dot{D}\dot{q}_1 - 2Cq_2Sq_3\dot{q}_3\dot{q}_1 D - \\
&2Sq_3\dot{D}\dot{q}_2 - 2Cq_3\dot{q}_3\dot{q}_2 D), \\
&\ddot{q}_3 = \frac{1}{D}((-Sq_1Sq_3 - Cq_1Sq_2Cq_3)A_x^0 + (Cq_1Sq_3 - Sq_1Sq_2Cq_3)A_y^0 - \\
&Cq_2Cq_3A_z^0 + Sq_2D\ddot{q}_1 - (Cq_2)^2Cq_3Sq_3D(\dot{q}_1)^2 + 2Sq_2\dot{D}\dot{q}_1 + \\
&2Cq_2(Sq_3)^2\dot{q}_2\dot{q}_1 D - 2\dot{D}\dot{q}_3 + 2Cq_3Sq_3D(\dot{q}_2)^2).
\end{aligned}\right\} \quad (3.42)$$

The following results have been received in this section:

1. The analytical decision of the inverse kinematic task for a mechanism with three axes (see Figure 3.1) with kinematic redundancy has been developed for:

a) «position» - (see (3.20)),

b) «velocity» - (see (3.34)),

c) «acceleration» - (see (3.42)).

This solution does not limit the possible movement of the goal point.

The derived equations serve for synthesis of the control algorithms, which can be realised in an onboard computer. Besides, these equations can be used for analysis of traditional schemes of scanning systems with two axes. Also it can be used in modelling to avoid procedures of digital integration, reducing calculation time and increasing the accuracy of the result.

2. Algorithm modification of the methods used has been carried out, this reduces the requirements for an onboard computer's memory essentially.

The reliability of the derived result is shown when realisation of all complex of produced calculation in ALCOR - system analytical transformations is achieved.

3.3 Analysis of Kinematic Schemes for Scanning System of Mobile Robotic System

To verify the choice of kinematic scheme for the view-finder scanning system system for an MRS it is necessary to carry out an analysis of advantages and disadvantages for the possible schemes. It is advisable to accept maximum values of velocities and accelerations in the scanning system joints that arise in process of work as a criteria for execution of the analysis. In practice, velocities and accelerations in the scanning system joints determine the attainable accuracy of stabilisation, maximum velocity of guidance, mass-inertia characteristics of the device and it's energy consumption. Moreover, together with the choice of kinematic scheme for a scanning system, it is necessary to estimate the complexity of the technical realisation of the computer control algorithms for the drives for DOF.

The peculiarities of the typical scanning system kinematic schemes should be considered. Typical schemes for scanning systems are shown at Figure 3.3. The problem is to choose the kinematic scheme that allows the most simple computer control algorithms to be obtained for mutually perpendicular axes of scanning system orientation, in the case of crossed axes of scanning system the solution of the inverse kinematic task becomes very difficult.

The kinematic scheme of a two axes scanning system with the azimuth-angle positioning axis is shown at Figure 3.3 a. This scheme is traditional for the design of the view-finder systems for different aims according to *Bairashevsky A.* and *Nichiporenko N.* [1982], *Belyansky P.* and *Sergeev B.* [1980]. The right Cartesian coordinate system- $O_0X_0Y_0Z_0$ is connected with the base of scanning system rigidly. q_1- is the rotation angle, q_3- is the position angle, L is the view-finder beam of sensors (it is not shown at the scheme). O_4 is a goal point, D is the distance- module of the view-finder beam, r_4^0 is the radius vector of the current coordinates for the goal point in the coordinate system of the scanning system base.

Due to *Zufrin A.* [1970] it is known that the kinematic scheme shown at Figure 3.3 a is widely used, due to its simplicity. But, it is noted that when applied with two-axes azimuth-angle position stabilisation, the velocities and accelerations considerably increase for the rotation angle especially with a large angle of position (typically, where the sum of real angle and maximum angle of vehicle inclination is more than 70^0). That is why two axes stabilisation of this type does not allow realisation of view-finding near the zenith *Rivkin S.* [1978].

The kinematic scheme of the two axes scanning system with two angles of position axes is shown at Figure 3.3 b. This scheme is a traditional one too (*Bairashevsky A.* and *Nichiporenko N.* [1982], *Belyansky P.* and *Sergeev* B. [1980]). The right Cartesian system- $O_0X_0Y_0Z_0$ is connected with the scanning system base rigidly. q_2, q_3 are the angles of position, L is the view-finder beam of sensoring (it is not shown at the scheme). O_4 is a goal point, D is the distance module of view-finder beam, r_4^0 is the radius vector of current coordinates for the goal point in the coordinate system of the scanning system base.

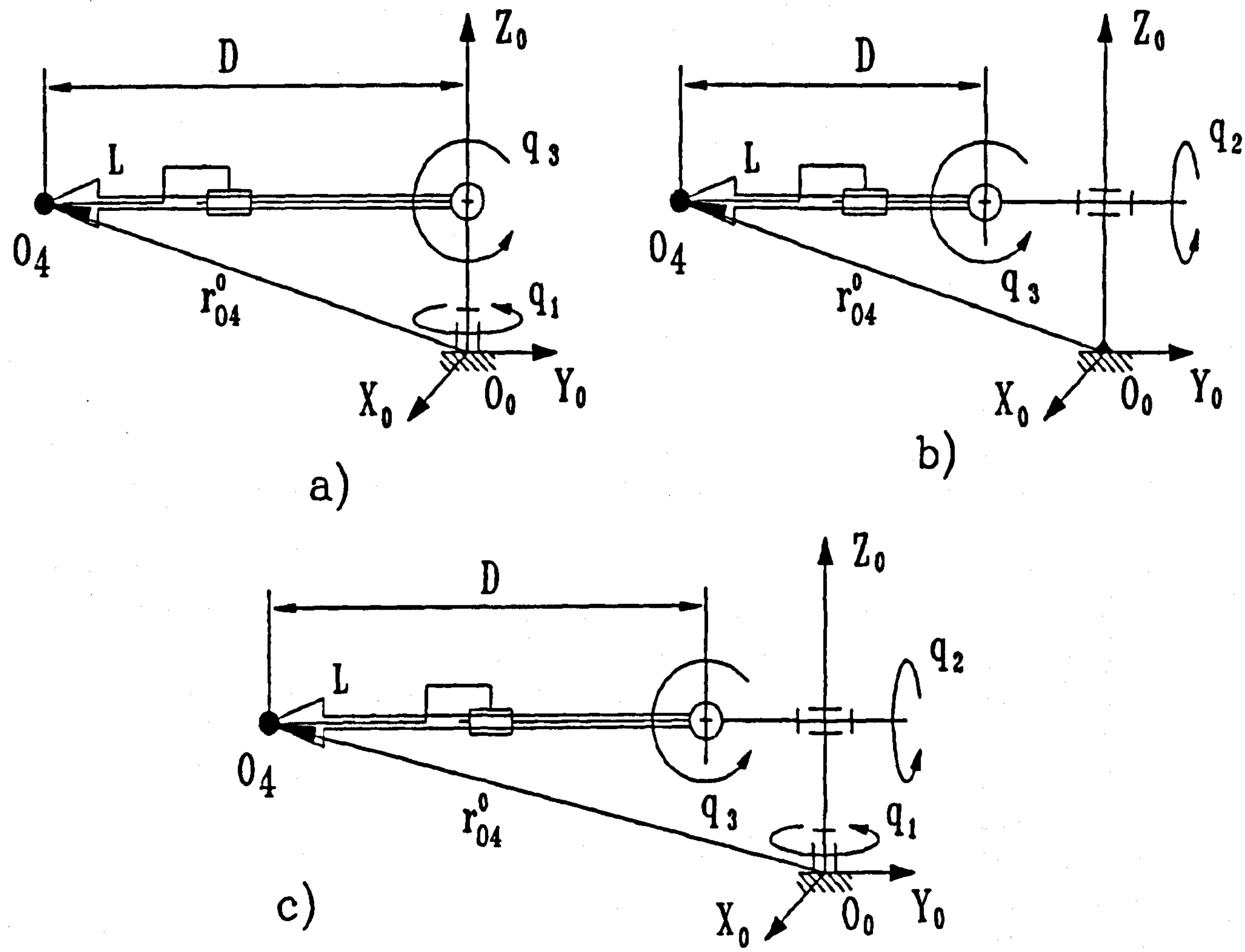


Figure 3.3 A Kinematic Scheme of Two Axes Scanning System with Azimuth-Angle of Position Axes

Figure 3.3 B Kinematic Scheme of Two Axes Scanning System with Two Angles of Position Axes

Figure 3.3 C Kinematic Scheme of Three Axes Scanning System with Mutually Perpendicular Axes

Here scanning systems with mutually perpendicular axes (two angles of position axes) are applied in systems for following high-altitude targets to provide constant self-aligning of targets near the zenith. Bringing in the azimuth axis allows avoidance ever increasing velocities and accelerations of the sensors when view-finding near the zenith *Hanevsky* [1951]. The disadvantage of such a scanning system is insufficient universality as it can not provide view-finding for small heights. That is why these scanning systems are rarely used.

Systems with three axes stabilisation are widely used in ship Radio Locator Station (RLS) with view finding of unmoveable objects where precise compensation for rocking is required *Bairashevsky A.* and *Nichiporenko N.* [1982]. A three axes scanning system with mutually perpendicular axes is shown at Figure 3.3 c. The scanning system has three rotation degrees of mobility with coordinates q_1, q_2, q_3. q_1 is the angle of rotation, q_2 is the angle of inclination for the pin axis, q_3 is the angle of position. A sensor (not shown in Figure 3.3 c) is connected in the joint of angle q_3 rigidly with L a view-finding beam, the module of which is equal to D in magnitude. D is the distance up to the O_4 goal point. A right Cartesian coordinate system-$O_0X_0Y_0Z_0$ is connected with the scanning system base rigidly. r_4^0 is the current position of the O_4 goal point. The advantage of this scanning system in relation to a two axes scanning system with mutually perpendicular axes (see Figure 3.3. b), is that it does not limit view-finding of top of a semisphere. But, in addition, the azimuth axis is added to this two axes system. Control of this scanning system is realised with minimum acceleration for all axes and without dependence on the position of the goal point in the top semisphere or from vehicle movement, according to *Popov E.* [1982]. The control algorithms for stabilisation modes are shown by *Rivkin S.* [1978]. The control algorithms for the following mode are developed by *Nesterov V.* [1994]. But the receiving data for arbitrary view-finding for this variant of scanning system is a difficult task.

It is noted in literature devoted to scanning system design that the implementation of two axes scanning systems in high precision systems can be justified for a small range of angles only, and in cases when the simplicity of construction and control is very important.

It is known that other configurations of scanning systems with three, four and more DOF in accordance with the number of stabilising components of the vehicle oscillations have been studied. But, algorithmically, all of them work in a two-axes scanning system mode. Thus multiaxes systems are controlled separately: two drives for the space orientation for the view-finding beam is controlled separately from other drives. These other drivers compensate for vehicle oscillations directly. Consequently, it is not meaningful from the point of view of control algorithms to select such, particularly as all of them have the main disadvantages of the two axes schemes (perhaps in to a smaller degree).

The main problem of two axes scanning systems is described below. For this aim the obtained analytical solution of the inverse kinematics for the three axes scheme shown at Figure 3.3 c (it is analogous to Figure 3.1) may be considered. The main interest is devoted to the solution equations of the inverse kinematic tasks in "velocities" and in "accelerations". The conditions of degeneration for velocities and accelerations of the generalised coordinates in scanning system mechanism joints are defined below.

According to the theory of limits:

$$\left.\begin{array}{l} \lim(\dot{q}_i,\ddot{q}_i)=\pm\infty, \\ UP(\dot{q}_i,\ddot{q}_i)\neq 0, \\ DOWN(\dot{q}_i,\ddot{q}_i)\rightarrow 0 i=1,..,N,- \end{array}\right\} \qquad (3.43)$$

where the following designations are introduced: N is a number of scanning system joints, $(\dot{q}_i,\ddot{q}_i)=\dfrac{UP(\dot{q}_i,\ddot{q}_i)}{DOWN(\dot{q}_i,\ddot{q}_i)}$.

$$DOWN(\dot{q}_i,\ddot{q}_i)\rightarrow 0,\ i=1,..,N,- \qquad (3.44)$$

This is the necessary condition of degeneration, with observance of it, the denominator of the equation for the corresponding derivation goes to zero limit, when keeping this condition. On other hand.

$$UP(\dot{q}_i,\ddot{q}_i)\neq 0,\ \ i=1,..,N,- \qquad (3.45)$$

This equation is a sufficient condition of degeneration.

The equations 3.34 and 3.42 include solutions of the inverse kinematic tasks in "derivatives" for kinematic schemes of two axes scanning systems shown at Figure 3.3 a and at Figure 3.3 b. Actually, the solution for scanning systems with two angles of position is obtained by the direct substitution $\{q_1,\dot{q}_1,\ddot{q}_1\}=0$.

$$\left.\begin{aligned}
\dot{q}_2 &= \frac{Sq_2V_z^0 - Cq_2V_x^0}{DSq_3},\\
\dot{q}_3 &= \frac{-Sq_2Cq_3V_x^0 + Sq_3V_y^0 - Cq_2Cq_3V_z^0}{D},\\
\dot{D} &= -Sq_2Sq_3V_x^0 - Cq_3V_y^0 - Cq_2Sq_3V_z^0,\\
\ddot{q}_2 &= \frac{-Cq_2A_x^0 + Sq_2A_z^0 - 2\dot{q}_2(Sq_3\dot{D} + 2DCq_3\dot{q}_3)}{D\times Sq_3},\\
\ddot{q}_3 &= \frac{-Sq_2Cq_3A_x^0 + Sq_3A_y^0 - Cq_2Cq_3A_z^0 + DCq_3Sq_3(\dot{q}_2)^2 - 2\dot{D}\dot{q}_3}{D}.
\end{aligned}\right\} \qquad (3.46)$$

The solution for a scanning system with the azimuth-angle of position axes is obtained with the substitution $\{q_2, \dot{q}_2, \ddot{q}_2\}=0$.

$$\left.\begin{aligned}
\dot{q}_1 &= \frac{Cq_1V_x^0 + Sq_1V_y^0}{DCq_3},\\
\dot{q}_3 &= \frac{-Sq_1Sq_3V_x^0 - Cq_3V_z^0 + Cq_1Sq_3V_y^0}{D},\\
\dot{D} &= Sq_1Cq_3V_x^0 - Sq_3V_z^0 - Cq_1Cq_3V_y^0,\\
\ddot{q}_1 &= \frac{Cq_1A_x^0 + Sq_1A_y^0 + 2\dot{q}_1(-Cq_3\dot{D} + 2DSq_3\dot{q}_3)}{D\times Cq_3},\\
\ddot{q}_3 &= \frac{-Sq_1Sq_3A_x^0 - Cq_3A_z^0 + Cq_1Sq_3A_y^0 - DCq_3Sq_3(\dot{q}_1)^2 - 2\dot{D}\dot{q}_3}{D}.
\end{aligned}\right\} \qquad (3.47)$$

The developed equations 3.46 and 3.47 have been analysed. Asymptotes of the degeneration areas are obtained from these equations directly using the theory of limits and the necessary conditions of degeneration.

Due to the equation 3.47 the necessary conditions of degeneration (in generalised coordinates) have the following form for the scheme of a two-axes scanning system with azimuth-angle of position.

$$\left.\begin{aligned} DCq_3 &\rightarrow 0, \\ D &\rightarrow 0. \end{aligned}\right\} \qquad (3.48)$$

It means that the degeneration areas for velocities and accelerations (areas of three dimension space), where possible sharp increase of velocities and accelerations of the scanning system joints are concentrated around the geometric objects given with the equations (in generalised coordinates):

$$\left.\begin{aligned} DCq_3 &= 0, \\ D &= 0. \end{aligned}\right\} \qquad (3.49)$$

The geometric interpretation of the necessary degeneration conditions for a two axes scanning system with azimuth-angle of position axes is considered. According to Figure 3.3 the first equation from 3.49 describes a straight line in Euclidean space $\Re^3$ with basis $O_0X_0Y_0Z_0$. This line coincides with O_0Z_0 axis. The second equation from 3.49 gives point the O_o. These statements are consistent with the known conclusions by *Rivkin S.* [1978], *Belyansky P.* and *Sergeev B.* [1980], that at first, the two axes scanning system with azimuth-angle of position axes does not work near the zenith area correctly, and secondly that any view-finding device has a degeneration area, which is concentrated near the connection point to vehicle.

Two important aspects for use in practice may be noted. The first, the degeneration area with asymptote $DCq_3=0$ moves inside some conic area with the presence of the base rocking. The second, the degeneration area is present even with full compensation of rocking (for example, with additional drives). There is no possibility to compensate it. The degeneration areas of two axes scanning system with azimuth-angle of position axes are shown in Figure 3.4a, with and without a rocking base.

Analogously, from 3.46, the necessary conditions of degeneration (in generalised coordinates) have the following form for the scheme of a two axes scanning system with two angles of position.

$$\left.\begin{aligned} DSq_3 &\to 0, \\ D &\to 0. \end{aligned}\right\} \qquad (3.50)$$

It means, that the degeneration area for velocities and accelerations is concentrated around the geometric objects, which are given with equations (in generalised coordinates):

$$\left.\begin{aligned} DSq_3 &= 0, \\ D &= 0. \end{aligned}\right\} \qquad (3.51)$$

Geometric interpretation of the necessary degeneration conditions for a two axes scanning system with two angles of position axes is considered. According to Figure 3.3 the first equation from equation 3.51 describes a straight line in Euclidean space $\Re^3$ with basis $O_0X_0Y_0Z_0$. This line coincides with O_0Y_0 axis. The second equation from equation 3.51 gives the point O_o. These statements are consistent with the known conclusions by *Rivkin S.* [1978], *Belyansky P.* and *Sergeev B.* [1980], that at first, a two axes scanning system with two angles of position axes is not effective for small heights, and secondly, that any view-finding device has a degeneration area, which is concentrated near the connection point to vehicle.

It is typical for these kinematic schemes, that at first, the degeneration area with the asymptote $DSq_3=0$ moves inside some conic area with the presence of base rocking. Secondly, the degeneration area is present even when full compensation of rocking (for example, with additional drives). There is no possibility to compensate it. Degeneration areas of a two an axes scanning system with azimuth-angle of position axes are shown in Figure 3.4 b with and without a rocking base.

According to the extension of obtained results, the following rule is formulated by induction for scanning systems with an arbitrary number of rotation axes, and controlled due to the principle for a system without kinematic redundancy.

The degeneration area axis of arbitrary variant for the scanning system design, in which the view-finding mode is realised with two DOF, coincides with the axis of the joint which takes part in the realisation of the view-finding mode and is the nearest to the vehicle.

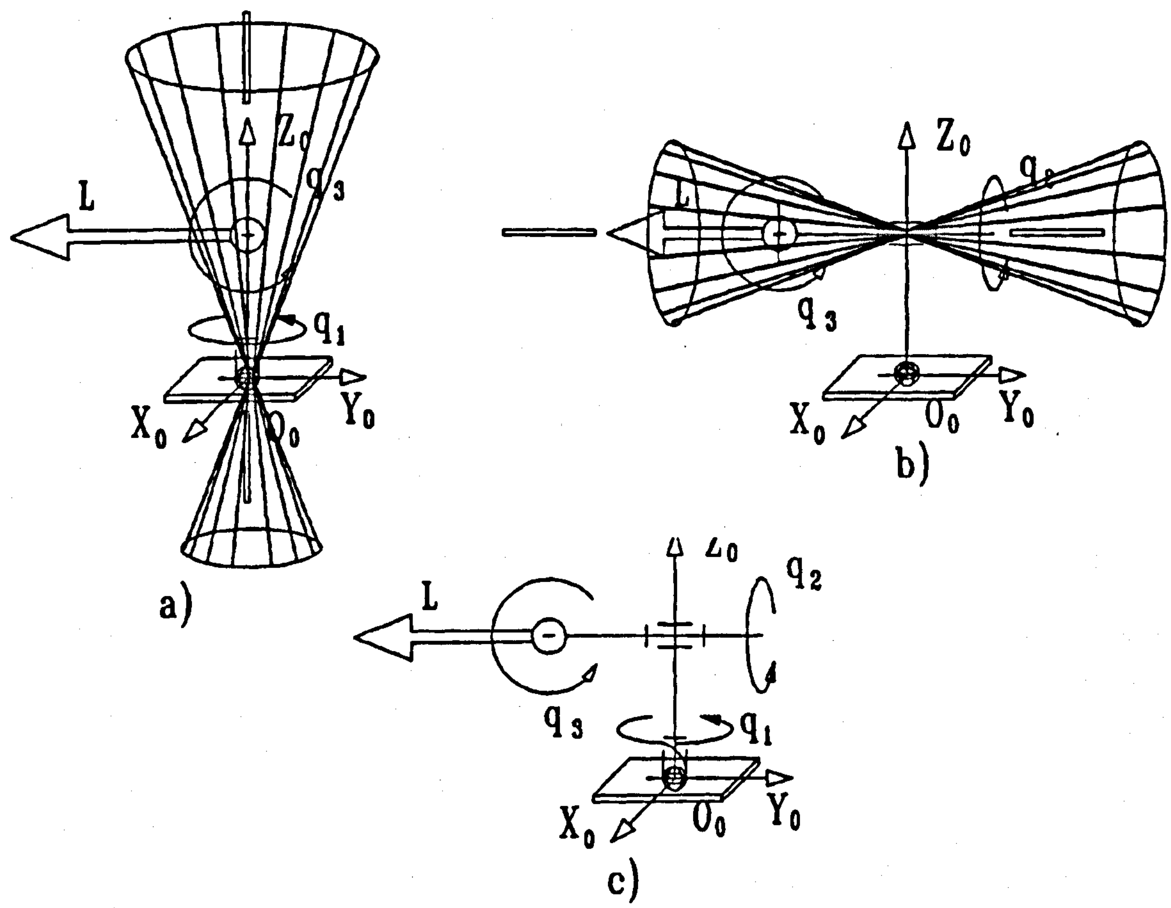


Figure 3.4 a Degeneration Area for a Two Axes Scanator with Azimuth-Angle of Position Axes

Figure 3.4 b Degeneration Area for a Two Axes Scanator with Two Angles of Position Axes

Figure 3.4 c Degeneration Area for a Three Axes Scanator with Mutually Perpendicular Axes

The main advantage of the three axes kinematic schemes of scanning systems is shown at Figure 3.3c. It follows from developed analysis of two axes scanning systems. Actually it is a two axes scanning system with the additional (kinematically redundant) azimuth axis. In same papers this scheme is named "rotary Cardano suspension". In practice, the kinematic redundant azimuth axis has to provide the removal of the degeneration area for a two-angle of position scanning system. Degeneration along the azimuth angle axis manifests only when $D \rightarrow 0$ with correct organisation of control due to *Nesterov V.* [1994]. Thus, the given kinematic scheme gives the possibility to reduce the degeneration area up to point (actually up to sphere of finite radius) as shown at Figure 3.4.c. It is important that rocking can be fully compensated with three drives of a scanning system, and it is not necessary to bring in additional degrees to compensate the rocking of the system. But to realise the advantages of this system it is necessary to synthesis control algorithms for a system with kinematic redundancy.

The results of executed kinematic analysis are presented in Table 3.1. The follow criteria have been chosen for consideration of the kinematic schemes of scanning systems shown in Figure 3.3 a-c: areas of possible application, reasons limiting application, difficulties of control realisation in real time. That is why the kinematic scheme of a scanning system with three mutually perpendicular axes has been chosen for the fast mechanical scanning system

In this chapter the following results are obtained.

1. A three axes scheme of scanning system is justified for application in a fast mechanical scanning system. Use of this scheme provides realisation of a view-finding system with the following characteristics in practice:

a) high fast action;

b) small power consumption;

c) augmented work zone.

2. Analysis of the kinematics for traditional schemes of scanning systems has been undertaken. The extent of limitation are explained for the practical application in view-finding systems of an MRS.

Kinematic scheme	Application	Limitation of application	Realisation of control
Two axes with azimuth-angle of position	Universal with limitations	Angle of position is limited	Controller
Two axes with two angle of position	Limited with limitations	Angle of azimuth is limited	Controller
Three axes with mutually perpendicular axes	Universal	Optimal control is required	Onboard computer is necessary

Table 3.1 Analysis of Kinematic Schemes for Scanning systems

CHAPTER 4

ALGORITHM FOR FAST SCANNING

4. ALGORITHMS FOR FAST SCANNING

This chapter is devoted to the synthesis of the trajectory task for the fast scanning for a survey sector. An estimating of the quality of the obtained control algorithm is carried out. It is shown that the offered algorithm provides uniformity of scanning frequency of subsectors with a reduction of the total scanning time of the view sector. This algorithm can be applied for scanning systems with any type of sensors.

Bellman's method of dynamic programming has been used for synthesis of this algorithm. The analysis of the application the developed algorithm in respect to use for the CCD vision system for an MRS has been carried out.

4.1 THE STATEMENT OF PROBLEM

The necessity for visualisation of a sufficiently large view sector and limitation of the view field for the sensors (it is determined with finiteness of the view value for the CCD vision system or the directional diagram for the locator) require the application of scanning systems: with electronic or mechanical spatial displacement of the view-finding beam. A typical example of electronic scanning is the phase antenna lattice. The setting of the sensor in the special mechanism of a spatial orientation- mechanism is a typical mechanical scanning system (*Kirsanov U.* [1997], *Vendik O.* [1965], *Volochatuk V. et. al.* [1971], *Popov E.*[1982]). Mechanical scanning is the more common method. Moreover, this method can be applied to any type of sensor. Therefore, the task to solved will be for mechanical scanning.

The scheme of a working scanning system is shown in Figure 4.1. The scanning system consists of a TV sensor fixed in the mechanism coincident with the azimuth angle of position axis. The right Cartesian coordinate system- OXYZ is rigidly connected with the base of mechanism. The mechanism has two degrees of mobility with angles: rotation - q_1 and position - q_2. The survey sector is given by a range of angles: rotation - $\pm\alpha$ and position - $\pm\beta$. The survey sector can be approximately represented at the form of plane rectangle within the O_{ij} coordinate system at the determined distance, which is given by the current value of the TV camera focal distance. In addition, the view angle of the TV camera in the O_{ij} plane is determined with linear coordinates: Δ_i, Δ_j and it sets the current map of environment (i, j). It is obvious from Figure 4.1, that the required linear sizes of the view sector: $n\Delta_i, m\Delta_j$ considerably exceed the possible sizes of the map sector. Accordingly, it is necessary to realise the displacement of a map subsector, while the given time for the whole sector visualisation, in other words, to scan a view sector.

4.1.1 Traditional Scanning Algorithm of the Survey of Environment Sector

Scanning of the given view sector is performed with the mechanism drive control. In practice, traditionally this task is solved with forming of programming control for movement of point of view-finding along the required trajectory of scanning. The program trajectory for the scanning system shown in Figure 4.1 is presented in Figure. 4.2. The scanning trajectory for direct movement around the environment sector is shown by continuous lines. The scanning trajectory with reverse direction around the environment sector is shown by dotted lines.

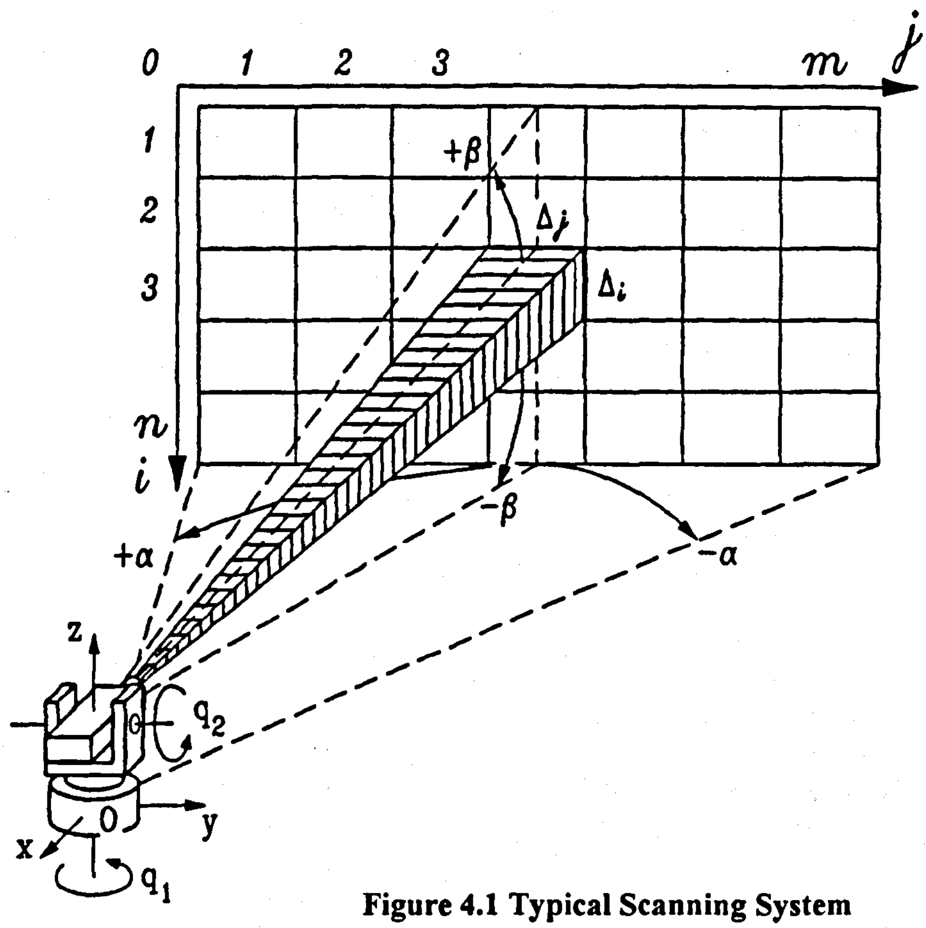


Figure 4.1 Typical Scanning System

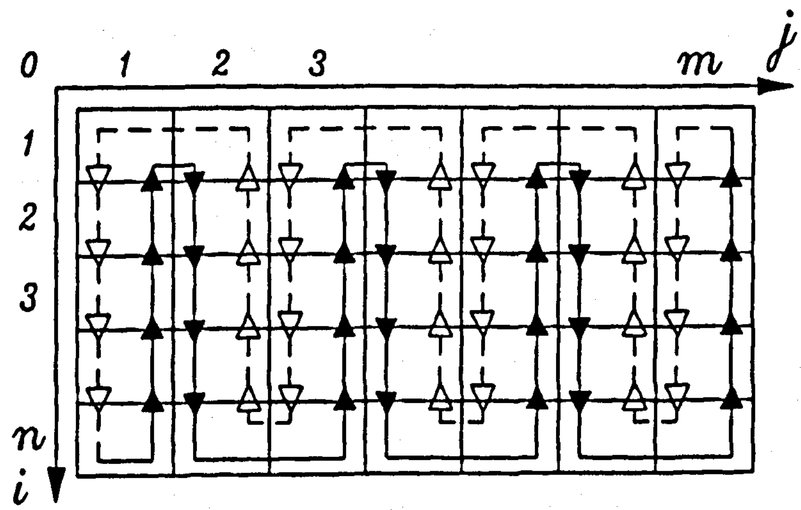


Figure 4.2 Traditional Algorithm of Survey for Environment Sector

The direction of movement between sectors is given by pointers in Figure 4.1. It is obviously from Figure 4.1, the traditional control for a scanning system mechanism consists of the realisation of discrete reflexive-translation movements along the azimuth angle, simultaneous with alternating signs evolvent along the angle of position. The change of movement direction along the corresponding axis is realised when exceeding the limits of view sectors.

4.1.2 Analysis of Traditional Algorithm

The main advantage of the traditional algorithm for view sector scanning is simplicity of realisation in practice. A simple calculator is enough to realise it. Moreover, another advantage of the traditional algorithm is that the power calculation for the kinematic scheme for the mechanism is a priori taken into account. The scheme of a mechanism with azimuth-angle of position axis (see Figure 4.1) has two degrees of mobility (with angles: rotation and position), each of them has a drive for the realisation of the required movement. In addition, it is obvious, that drive of rotation angle is more loaded as its load includes the whole angle of position orientation system. On the other hand, accuracy requirements to both drives are equal. Consequently, the practical conclusion is followed (according to *Popov E., et. al.* [1978], *Kochergin V.*[1988]) that, as a rule, the achieved velocity of the scanning system on the rotation angle is half as small as than the achieved velocity of the scanning system along the position angle. This relation is taken into account in the traditional algorithm (see Figure 4.2), where the movement along the angle of position is dominant.

In spite of considered advantages, at present time, there is a tendency to move away from the traditional scanning algorithm of the view sector. There are some reasons for this. One of them is that the modern information systems include the possibility for powerful onboard computers. It permits one to realise the complex control algorithms in real time. But the main reason is the consequence of the principle of reflexive-translation movement for the view-finding point.

The following example is considered to highlight the main disadvantage of the traditional scanning algorithm. In this example a view sector with size 3 × 10 is used. Scanning of this sector carried out according to the traditional scanning algorithm is shown in Figure 4.2. The qualitative result is of interest to us. The following assumptions are introduced for simplicity. The displacement periods from subsector of the current environment map along the horizontal and along the vertical in the O_{ij} plane are equal to 1. View periods of separate subsectors for view sector guaranteeing with the algorithm are shown in the Figure. 4.3, for accepted conditions. According to Figure 4.3, the essential difference between scanning frequency of individual subsectors is shown. It achieves a value of 1.93 in the considered example.

54	52	42	40	30	30	40	42	52	54
56	50	44	38	32	32	38	44	50	56
58	48	46	36	34	34	36	46	48	58

Figure 4.3 Scanning Periods of Different Subsectors for Environment Sector with Use of a Typical Algorithm of Survey

The shown peculiarities of the traditional algorithm are disadvantages for the design of fast systems, as the required scanning frequency of the view sector is determined by the smallest scanning frequency of each separate subsector.

On the basis of the performed analysis, it is possible to formulate the task in the following manner:

to undertake the trajectory movement of the view sector, such that a maximum value and equality of scanning frequencies for individual subsectors is provided. The given configuration of the view sector has to include a known finite number of subsectors (see Figure 4.1 and Figure 4.2) and the given values of the weight coefficients that show the relative cost of control when moving from one subsector to another.

4.2 SYNTHESIS OF THE FAST SCANNING ALGORITHM

4.2.1 Tasks of Mathematical Programming

The task of mathematical programming is the task in which the maximum or minimum value of some function with a number of restrictions is determined (*Eremin I., Astafiev N.* [1976]). It can be describes by definition of a vector: $x^* = (x_1^*, \ldots, x_n^*)^T$. This vector is the solution of a task for a definition of the f(x) function extremum with restrictions (*Kudrjavcev V.* [1991]).

$$g_i(x) \geq 0 \qquad i = 1, \ldots, m, \qquad \textbf{(4.1)}$$

$$h_j(x) = 0 \qquad j = 1, \ldots, p. \qquad \textbf{(4.2)}$$

If f, $\{g_i\}$ and $\{h_j\}$ functions are linear, than the task described above is called a task of linear programming. If at least one of these functions is non-linear than it is a task of non-linear programming (*Marchuck* [1980], *Moiseev N.* and *Ivanovsky J.* [1978]).

The methods of solving for the linear programming tasks are described by *Golshtein* E. and *Udin D.* [1966] *Lyashenko I.* [1975], *Udin D.* and *Golshtein E.* [1969], *Zuhovickiy S* and *Avdeeva L.* [1964] and others. Many tasks widely used in practice lead to linear programming tasks, which have singularities permitting one to obtain more easy solution algorithms, or new solving methods. In particular, the transport task. Its modification, as the so called distributive tasks, can be considered as a specific generalisation of the transport task, assignment problem *Lyashenko I.* [1975]. Linear programming with block structure, linear programming with parameter also belong to these tasks.

The tasks with the requirements of all (part) variables to be integer are separated from the practically important tasks the definition of conditional extremum for linear function. These tasks are called the tasks of integer (part integer) programming..

It is possible to select the tasks in which the desired extremum solution is described with some permutation (combination) of the known (admissible) set of numbers. These problems are called the tasks of combinatorial type.

4.2.2 Reduction of the Set Problem to the Transport Task

The task formulated in the section 4.1 above is considered in this section. The following notations are introduced: k is the number of subsectors of the environment sector; m, n are the number of subsectors for an environment sector corresponding to i and j axes (see Figure 4.2); t_{ij} is the motion time between i-th and j-th subsectors; $F(x)$ is the quality functional; x_{ij} is the auxiliary value determined below; C_{ij} is the matrix of motion time from i-th and *j*-th subsectors.

The next facts are considered as known. The known rectangular sector consists of *k* subsectors (*k* is known). A configuration of the received rectangle is known, it is $m \times n = k$. The motion time between subsectors is known and is respectively equal to t_{ij}, $i \neq j$, $i=1, 2, .., k$, $j=1, 2, .., k$ (motion time from i-th subsector to j-th one can be calculated with the known distance between subsectors and the known velocity of the scanning system, which is determined by its specification).

Starting the survey from an arbitrary subsector of the environment sector, it is necessary to look at all subsectors and return back to the initial subsector. Each subsector must be surveyed only once, therefore a view route forms a closed cycle without loops.

It is necessary to minimise the summary time of survey. The quality function F(x) has been introduced for this aim. It includes the sum of all the motion time between subsectors.

A set of *k* ordered pairs of subsectors that form the survey route, passing through each subsector only once, is named *S*.

$$S=[(i_1,i_2),(i_2,i_3),...,(i_{n-1},i_n),(i_n,i_1)].$$

Each pair of subsectors (i, j) form the communication of route.

The mathematical model of the task can be represented in the next form:

$$x_{ij}=\begin{cases}1, & \text{if scanning system moves from the i-th subsector to j subsector,}\\ 0, & \text{in another case;}\end{cases}$$

$$F(x)=\sum_{i=1}^{k}\sum_{j=1}^{k}t_{ij}x_{ij} \qquad (4.3)$$

with the following conditions:

$$\sum_{i=1}^{k}x_{ij}=1, \quad j=1,2,\ldots,k, \qquad (4.4)$$

$$\sum_{i=1}^{k}x_{ij}=1, \quad i=1,2,\ldots,k, \qquad (4.5)$$

$$u_i-u_j+kx_{ij}\leq k-1 \qquad (4.6)$$

u_i and u_j are arbitrary values.

The condition 4.4 means that the scanning system passes out from each subsector only once; the condition 4.5 means that the scanning system enters each subsector only once; the condition 4.6 provides closure of the route, *k* subsectors visited and on absence of loops.

The received model 4.3 - 4.6 is an integer transport task of combinatorial type. Hence, it can be solved by one of the specific methods for these tasks.

4.2.3 Methods of Solving for Integer Transport Task of Combinatorial Type

There are some methods of solving the integer transport task of combinatorial type. The most widespread are the branch and bound algorithm, Bellman's method and algorithm of random search with local optimisation.

The main idea of the branch and bound algorithm is based on matrix reduction and the method with reducing the C_{ij} matrix first. The branch and bound algorithm allows consideration of some closed cycle with no less than k steps, where the solution can be non-optimal. The complexity of the calculating process consists of the following: it is necessary to conduct the analysis of matrix elements (reduction) at each step to choose the zero elements and the procedure of choosing a claimant to branching, and estimating at each step. This task can not be solved with use of this algorithm when k is large, because of the increasing number of branches for the solution graph (*Lyashenko I.* [1975]). Accordingly, it is not expedient to apply branch and bound algorithm for solving the task as stated in section 4.2.2.

Another algorithm for the integer transport task of combinatorial type solution is the algorithm of random searching with local optimisation. In this searching algorithm the approximate decision is based on the following idea: a random search of the initial solution (initial sequence of the subsectors view); vicinity of this solution is isolated and a local optimum of the goal function is determined at this solution vicinity points. A search for a local optimum is conducted with common sorting, as the vicinity consists of small number of points. Subsequently, a random choice of another decision is carried out and the local optimum of the goal function is determined in its vicinity. This process is repeated. The sequence of subsectors viewed is considered as the relative optimal solution, under which the goal function reaches the minimum value of all considered sequences. The best sequence for one initial solution can be achieved with no more than *k*-1 substitutions. The main advantage of this algorithm is the possibility to search the optimal solution no longer than a desired time and moreover this algorithm and its modifications (simultaneous permutation in more than one subsectors is realised) permit a solution in any moment of time.

But this algorithm has a number of disadvantages. It does not allow estimation of the difference between the received solution and the desired optimal one at given step. Moreover, no rule exists to choose the optimal solution. It is therefore not expedient to use the of random search algorithm, with local optimisation, for solving the task as stated in section 4.2.2.

The algorithms based on the application of dynamic programming are used also to decide the integer transport task of combinatorial type. Bellman's optimum principle is fundamental to dynamic programming. According to Bellman's principle, the optimal control does not depend on the previous state of the system at any moment of time, it is determined by the control aim and the system state at the considered moment of time only. Therefore, if the penultimate state of the system is known independently of how the system went into this state, the control process at the last section would be the best of all the possible ones.

Algorithms of solving the integer transport task of combinatorial type are based on the following fact:

let $(i'_1, i'_2, \ldots. i'_{k-1}) = S_{k-1}(0, i_1, i_2, \ldots i_{k-1}, i_k)$,

$1 \le p \le k-1$, than

$(i'_1, i'_2, \ldots i'_p) = S_p(0, i_1, i_2, \ldots i_p, i_{p+1})$.

That is, the optimal sequence is optimal for any of its subsequence.

The advantage of algorithms, based on dynamic programming methods, in comparison with the direct sorting of variants, is the reduction of the solution graph when using the chosen preliminary optimal trajectories during the further stages. Such algorithms permit achievement of the optimal route after *k* steps. But, the complexity of a $2 \le p \le k$ step performance and necessary significant value of necessary RAM require modification of the algorithms by taking into consideration the specific character of the set task (*Lyashenko I.* [1975]). Moreover, the calculation time is great enough for large *k*. Therefore the algorithm, based on use the Bellman's method and with elimination of the described disadvantages, will be synthesised.

4.2.4 Decision Algorithm

To solve the set task in the section 4.2.2, an algorithm, based on use of Bellman's method has been offered. The model of view-finding system motion is shown in Figure 4.4. The following assumption is taken. The motion time in the horizontal direction (along j axes) between the neighbouring subsectors (from subsector with [m, n] coordinates to subsector with [m ± 1, n] coordinates) is equal among themselves and equal to k_1. Motion time in the vertical direction (along i axis) between neighbouring subsectors (from subsector with [m, n] coordinates to subsector with [m, n ± 1] coordinates) is equal among themselves and equal to k_2. The time of motion in a diagonal direction (see Figure 4.4) between neighbouring subsectors (from subsector with [m, n] coordinates to subsector with [m ± 1, n+1] or [m ± 1, n-1] coordinates) is equal among themselves and equal to k_3.

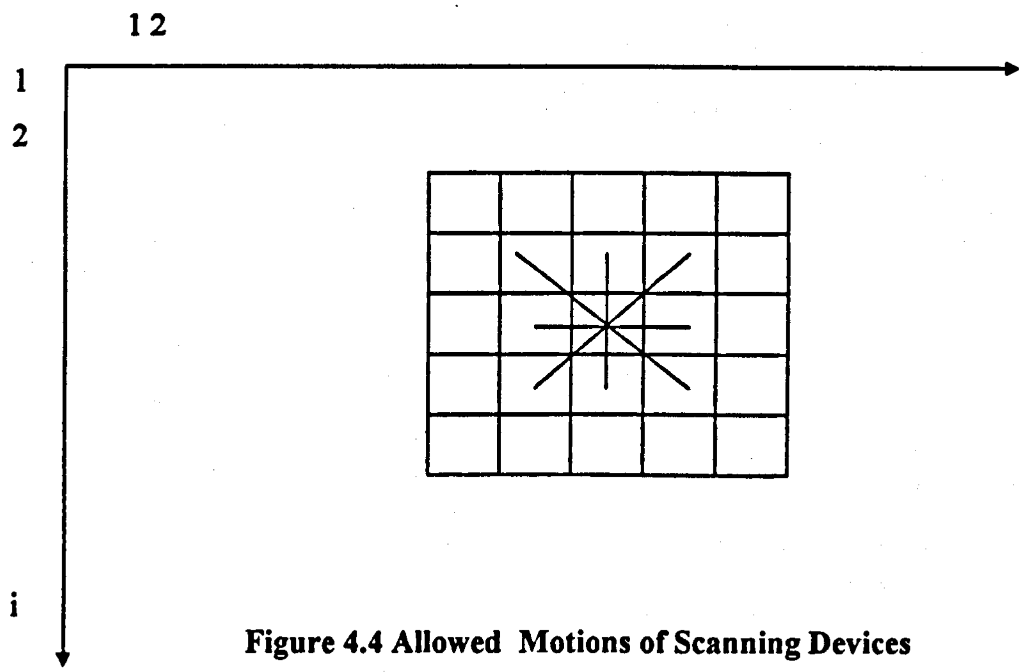


Figure 4.4 Allowed Motions of Scanning Devices

The next difference between the offered algorithm and traditional methods is prohibition of all motion except one between the neighbouring subsectors (the allowed motions are shown in Figure 4.4). Moreover in most cases it is possible to limit the number of «diagonal» motions. It is enough to permit one «diagonal» motion for a survey of the rectangular sector without «cuts». Such exclusion considerably limits the variants of sorting for possible solutions and, thus, considerably reduces calculation time. The optimal solution can not be lost. This fact will be shown below.

The algorithm of finding the optimal survey trajectory of subsectors for the environment sector is shown below. The following designations is used: S_k is the trajectory view for the environment sector; s is the cycle step; f_s is the value of the functional on the s-th step; l is the counter of diagonal number; l_{max},- the maximum value of allowed «diagonal» motions; m- is the number of subsectors along j axes (see Figure 4.2 and 4.4); n- is the number of subsectors along i axes; k is the number of subsectors; k_1, k_2, k_3 are weight coefficients mentioned above; q is an ordinal number of subsector; $k_{q-1\,q}$ is the weight coefficient for motion from (q-1)-th subsector to q-th subsector; i is an ordinal number of subsector along the vertical axes (see Figure 4.4); j is an ordinal number of subsector along the horizontal axes (see Figure 4.4). The following initial data must be known: coordinates of the first subsector, k, m, n ,k_1, k_2, k_3, l_{max}. If the environment sector is not rectangular or has «cuts», than l_{max} value must be more than 1. In the case of rectangular environment sector, the maximum value of the allowed «diagonal» motions is accepted as equal to 1.

0. To recalculate coordinates of each subsector: q=m(i-1)+j.

1.a To determine the allowed motion from first the subsector. To form possible routes $S_1(q_1, q_2)$, if «diagonal» motion is added then l=l+1.

1.b To define the value of the functional at the 1-st step for each possible route; f_1 is equal to the corresponding weight coefficient.

1c. To store the allowed routes, values of its functional and its diagonal counters.

.

.

.

s.a To determine the allowed motions out of the (s-1)-th subsector. To form the possible routes with adding the allowed subsector to S_{s-1}, which is absent in the corresponding S_{s-1}, $S_s(q_1, q_2, \ldots, q_{s-1}, q_s)$, if «diagonal» motion is more than l=l+1 for a corresponding route.

s.b To check the diagonal counters for each route; to exclude the routes with l more than l_{max}.

s.c To determine value of the functional on the s-th step for each possible route; $f_s = f_{s-1} + k_{q-1\,q}$.

s.d To apply Bellman's method to compare the possible routes; to exclude routes with nonoptimal value of the functional.

s.e To store the remaining allowed routes, their functional values and their diagonal counters.

•

•

•

k.a To determine the routes, for which motion from the q_{k-1}-th subsector to the q_1 subsector is allowed. To form possible routes with adding to the S_{k-1} q_1 subsector: $S_k(q_1, q_2, \dots, q_{k-1}, q_1)$, and if the diagonal» motion is included than l=l+1 for the corresponding route

k.b To check the diagonal counters for each route; to exclude routes with l more than l_{max}.

k.c To determine the value of the functional on the k-th step for all the possible routes; $f_k = f_{k-1} + k_{k-1\,1}$.

k.d To compare all functionals. The route that provides the minimum value of functional is the optimal one.

The offered algorithm has the advantages of methods, based on use of Bellman method. It provides the optimal solution after *k* steps.

The introduced coefficients can be useful for taking into consideration the dynamics of the mechanism when undertaking control algorithm synthesis. In contrast to traditional algorithms, the offered algorithm does not require knowledge of the motion time between the arbitrary subsectors (in the common case, it is a $k \times k$ dimension matrix). It requires knowledge of only three coefficients. The introduction of these coefficients considerably simplifies the initial data (especially for large *k*) required to calculate the optimal route.

The main disadvantage of such algorithms, based on Bellman method, is increasing calculation time for an optimal route when *k* grows in proportion to the $k!(k-1)!$ value

(*Lyashenko I.* [1975]). Calculation time becomes inadmissible high for practical application of the algorithms for view-finding systems of an MRS with high *k*. Therefore a number of additional constrictions of the possible motion of the view-finding device have been included in the offered algorithm. Accordingly the calculation time in the offered algorithm is approximately dependent on 2^k. The same dependence is observed for the necessary RAM of the chosen computer.

4.3 RESEARCH STUDY OF THE DEVELOPED ALGORITHM

4.3.1 The Influence of the Number of Subsectors to the Calculation Time

The main criterion of estimation for the offered algorithm is the calculation time for an optimal route. This criterion has been chosen with taking into consideration the specific character of the MRS. The time to chose a view route must not exceed a determined value. The offered algorithm, which is the method of discrete programming, is critical to *k* (the number of subsectors in the environment sector). According to *Lyashenko I.* [1975], the algorithms based on the Bellman method permit an optimal solution after *k* steps. But the calculation time is especially great for large *k*. This fact does not allow one to apply these algorithms for such type of tasks where the calculation time must be less than some known value. The development of fast computer technology allows a reduction in the calculation time with algorithms based on Bellman's method. It permits one to use these algorithms for solving such tasks, where its application was previously inexpedient.

The limits of applicability for the offered algorithm as a function of the number of subsectors within a given environment sector has been estimated. A mathematical simulation of the offered algorithm has been carried out on computers with different processors. The results of the simulation are shown in the Tables 4.1 a-c.

The calculation time for an optimal view route with respect to the number of subsectors for a given environment sector is shown in Figure 4.5 a-c.

Configuration of environment sector	Number of subsectors	Calculation time on computer with 386 SX-33 processor (s)
3×12	36	0.5
3×14	42	2.5
3×16	48	19
4×12	48	19.5
5×10	50	28

Table 4.1 a. Dependence of Calculation Time of the Developed Algorithm from Number of Subsectors with Use the Computer with 386 SX-33 Processor

Configuration of environment sector	Number of subsectors	Calculation time on computer with 486 DX4-100 processor (s)
3×14	42	0.25
3×16	48	3.0
4×12	48	3.2
5×10	50	5.5
5×12	60	60

Table 4.1 b Dependence of Calculation Time of the Developed Algorithm from Number of Subsectors with Use the Computer with 486 DX4-100 Processor

Configuration of environment sector	Number of subsectors	Calculation time on computer with Pentium 200 MMX processor (s)
4×12	48	0.1
9×10	90	0.9
5×19	95	3.5
5×20	100	20
10×10	100	20

Table 4.1 c Dependence of Calculation Time of the Developed Algorithm from number of Subsectors with Use the Computer with Pentium 200 MMX Processor

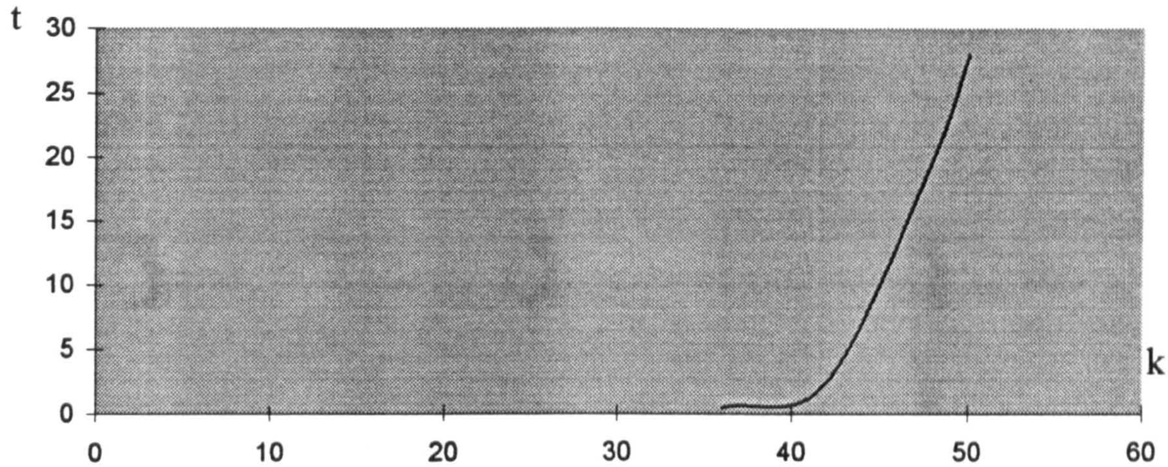


Figure 4.5 a Dependence of Calculation Time of the Developed Algorithm from Number of Subsectors with Use the Computer with 386 SX-33 Processor

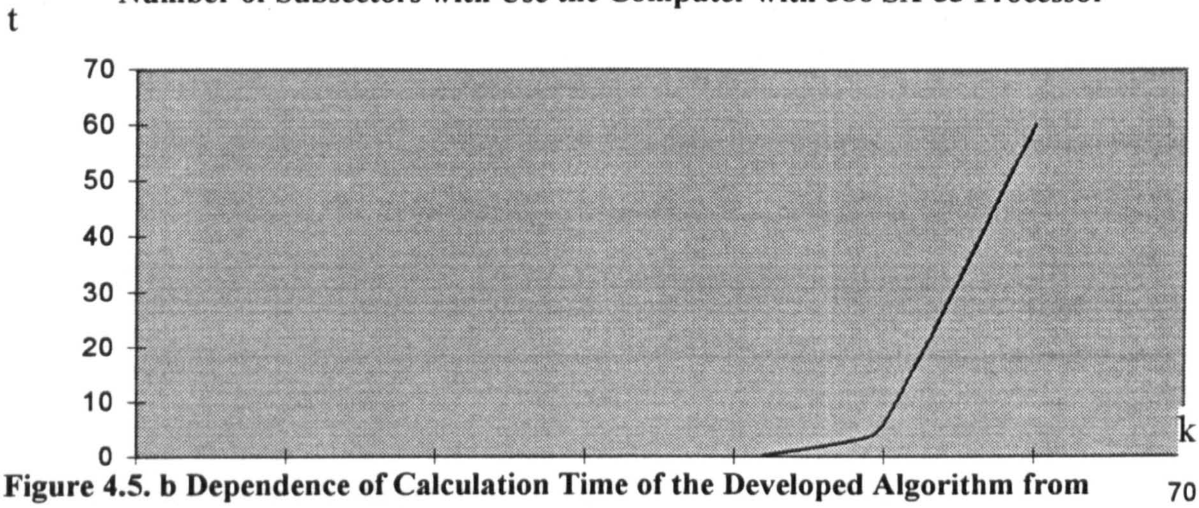


Figure 4.5. b Dependence of Calculation Time of the Developed Algorithm from Number of Subsectors with Use the Computer with 486 DX4-100 Processor

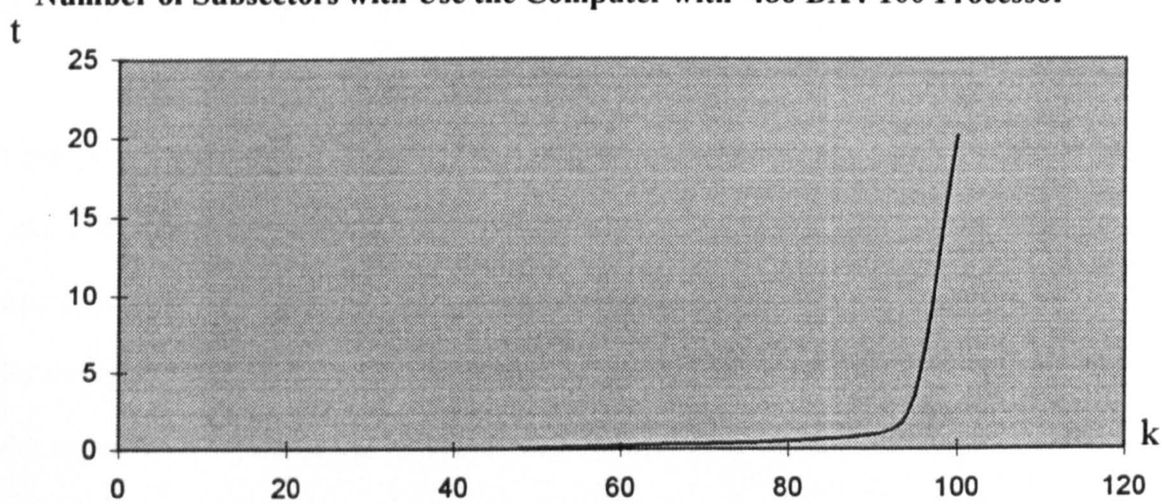


Figure 4.5 c Dependence of Calculation Time of the Developed Algorithm from Number of Subsectors with Use the Computer with Pentium 200 MMX Processor

The calculation time of optimal route for the environment sector with 100 subsectors does not exceed 20 seconds when using computer with a Pentium 200 MMX processor. It is fast enough for implementation of the offered algorithm for an MRS view-finding system. The calculation time sharply rises with increasing number of subsectors (see Figure 4.5). Accordingly it is not expedient to use this algorithm to determine an optimal view of the environment sector where there one more than 100 subsectors. The types of sensors, for which application of offered algorithm is useful, is determined below.

In practice, the typical work mode for a CCD vision system of an MRS is the following: focus of camera is 4.0^0; view along angle of azimuth is $\pm 45^0$; view the along the angle of position is $\pm 15^0$. A sector of view with such size, scanned by TV camera with such focus, has the follow configuration - 4×12 - 48 subsectors. According to Figure 4.5a, it takes 20 seconds to calculate an optimal view route of a given environment sector when using a computer with a 386 SX-33 processor. This result is plausible for realisation with an onboard computer of an MRS. Hence, it is possible to draw a conclusion about possibility of application of the offered algorithm in an onboard computer of an MRS to define the optimal view route of a given environment sector using a CCD vision system.

In practice, the typical work mode for a radio locator channel of an MRS is the following: width of direction diagram is 0.5^0; view along angle of azimuth is $\pm 20^0$; view along the angle of position is $\pm 10^0$. The survey sector with such a size, scanned by locator with such a width of direction diagram, has the follow configuration - 80×40- 3200 subsectors. Calculation time with use offered algorithm for determination optimal view sector for a given sector is of the order of hours (see Figure 4.5). Thus, it is inexpedient to use the offered algorithm in an onboard computer of an MRS to define the optimal view route of a given environment sector with it one is using a radio-locator channel.

The following results have been obtained in this section:

1) the offered algorithm can be recommended for application in an onboard computer of an MRS for calculation of an optimal view route of a given environment sector when using a CCD vision system;

2) It is inexpedient to apply the offered algorithm for computation of an optimal view route of given environment sector when using a radio locator channel.

4.3.2 Research of the Developed Algorithm Work with Implementation the Different Weight Coefficients

One advantage of the offered algorithm is the possibility to take into consideration the drive dynamics of the scanning device. It realisation allows use of different weight coefficients k_1, k_2, k_3. The influence of weight coefficients to the optimal view route has been analysed for typical practical configurations of the view sectors for a CCD vision systems. Three different configurations of the environment sector, typical for a CCD vision system of an MRS - 3×10, 6×10 and 4×12 have been considered.

The choice of weight coefficients is dependent on the kinematic scheme of the mechanism and on the performance characteristics chosen drives. According to *Kochergin V.* [1988], for the mechanisms with azimuth-angle of position axes, a drive of rotation angle is more loaded from the power point of view, as all of the angle of position system is a load for it. It was shown above, that, as a rule, the velocity along the rotation angle half as small as the velocity along the position angle. Therefore the following coefficients: $k_1=1$, $k_2=0.5$ and $k_3=1.2$ (see Figure 4.4) have been chosen for the mathematical simulation. At a given time, the motion velocity along the rotation angle axes can be considered equal to the motion velocity along the angle of position axes (see Figure 3.3c), for a mechanism with three mutually perpendicular axis. The following coefficients: $k_1=1$, $k_2=1$ and $k_3=1.5$ have been chosen. The optimal view routes with weight coefficients - $k_1=1$, $k_2=0.5$ and $k_3=1.2$ for configuration of given sector- 3×10, 4×12, 6×10 are shown in fig 4.6 a, b and c respectively. The optimal view route with weight coefficients - $k_1=1$, $k_2=1$ and $k_3=1.5$ for configuration of the given sectors - 3×10, 6×10, 4×12 are shown in Figure 4.7 a, b and c respectively.

The equal optimal trajectories have been received for three different sectors of environment, which are typical for a CCD vision system of an MRS, with the application of the different weight coefficients. It allows one to draw a conclusion that the introduction of different weight coefficients k_1 and k_2 for different kinematic schemes of MRS mechanisms for CCD vision systems give little benefit.

Consequently, it can be deduced that weight coefficients where k_1 and k_2 equal 1 are acceptable. This considerably simplifies the task of determination of an optimal view route. The results of the mathematical simulation are shown in Table 4.2: maximum period of scanning for the offered algorithm and for the traditional algorithm. The weight coefficients k_1 and k_2 have been set equal to 1, the weight coefficient k_3 has been set equal to 1.5. The offered algorithm increasing the maximum period for scanning by $\frac{2k-2}{k}$ time (i.e. approximately twice).

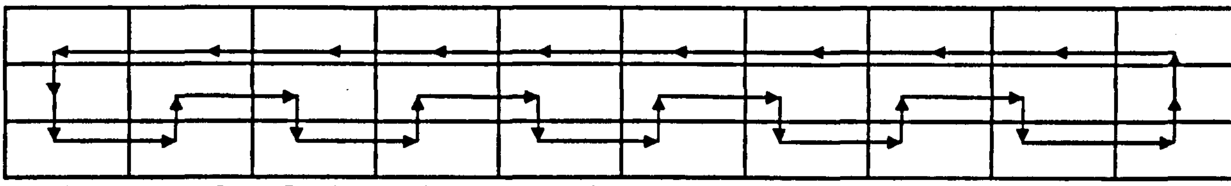


Figure 4.6 a The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=0.5 and k_3=1.2 for Configuration of a Given Sector - 3×10

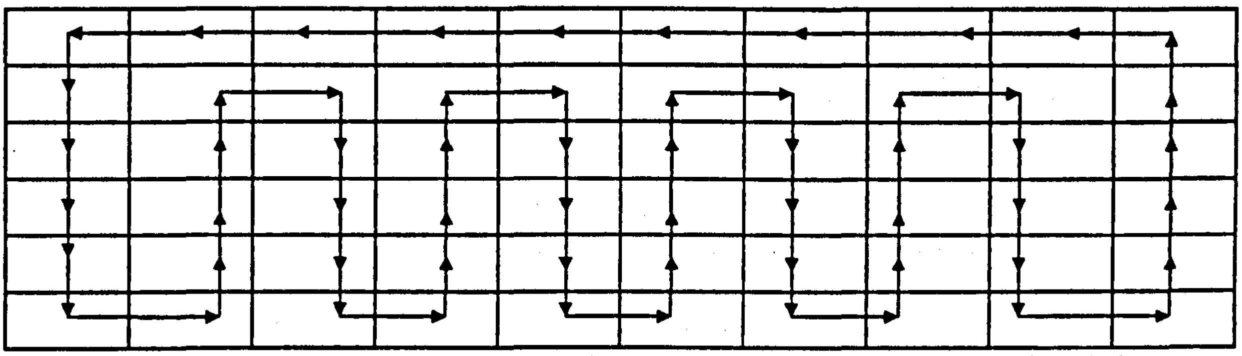


Figure 4.6 b The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=0.5 and k_3=1.2 for Configuration of a Given Sector - 6×10

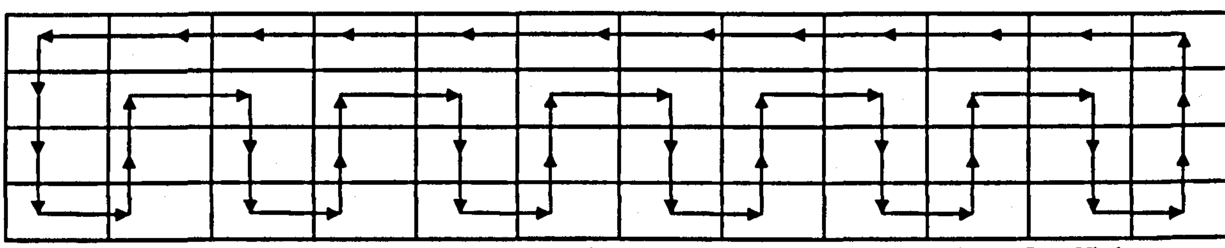


Figure 4.6 c The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=0.5 and k_3=1.2 for Configuration of a Given Sector - 4×12

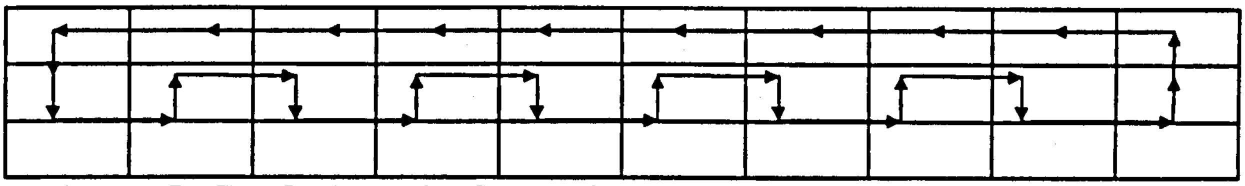


Figure 4.7 a The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=1 and k_3=1.5 for Configuration of a Given Sector - 3×10

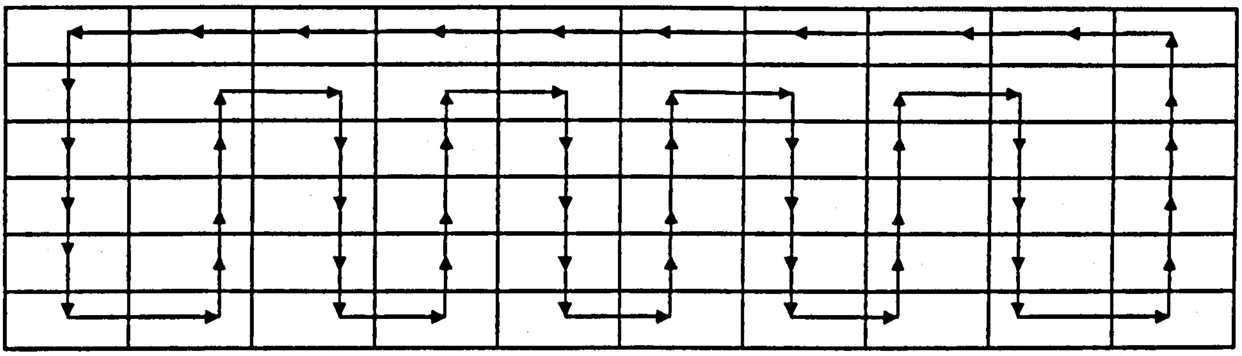


Figure 4.7 b The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=1 and k_3=1.5 for Configuration of a Given Sector - 6×10

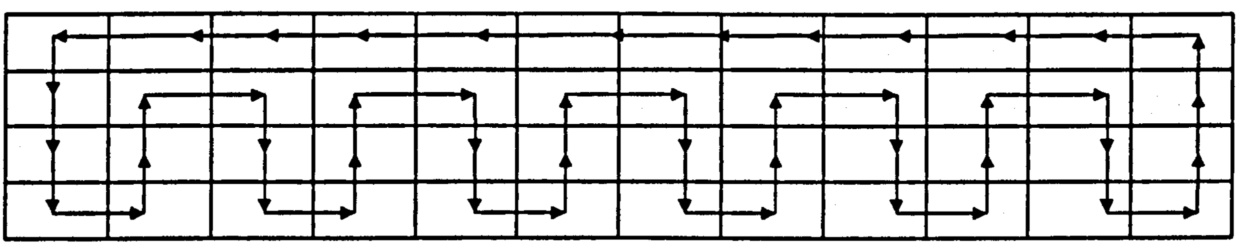


Figure 4.7 c The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=1 and k_3=1.5 for Configuration of a Given Sector - 4×12

Configuration of environment sector	Maximum period of scanning for any subsector	
	the traditional algorithm	the offered algorithm
3×10	58	30
6×10	118	60
4×12	94	48

Table 4.2 Maximum Periods of Scanning

Notation

The aim of introduction the different weight coefficients is to provide the expedient scanning trajectory. This scanning trajectory must provide optimal distribution of load for the scanning system actuators. To clarify this idea the following example has been considered. Our task is to provide optimal distribution of load for environment sector shown in Figure 4.6. b. It is necessary to make 58 simple movements to scan this sector. The above shown introduction of weight coefficients permits to choose trajectory with 40 vertical and 18 horizontal movements. In the case, when the task is to find such trajectory, which provides maximum of horizontal movements, it is necessary that $k_2 \rangle k_1$.

4.3.3 Research of Trajectory Form of the Optimal View Route of the Environment Sector for CCD vision system

Analysis of the necessary calculation time for an optimal view route for a set of environment sectors with varying number of subsectors has been carried out and is described in section

4.3.1 above. The analysis shows that if the sector of environment includes more than 100 subsectors then calculation time exceeds 20 seconds, even when using a computer with a Pentium 200 MMX processor. In some cases, it is desirable to reduce the calculation time for the optimal view route to 0. This is possible where preliminary knowledge of the optimal view route for an environment sector of arbitrary configuration exists. The task of finding this preliminary known optimal view route for an environment sector with arbitrary configuration is therefore apparent.

The optimal trajectories for typical configurations of the environment sector have been determined in sections 4.3.1 and 4.3.2. The possibility to consider weight coefficients k_1 and k_2 equal to 1 is shown in 4.3.2 section for the mechanism of scanning devices of arbitrary configurations.

Three different configurations of environment sector have been selected:

- a rectangular environment sector without «cuts» with k subsectors, where $k=m \times n$ and at least one of the values, m or n, is even;
- a rectangular environment sector without «cuts» with k subsectors, where $k=m \times n$ and both of the values, m or n ,are odd;
- A rectangular environment sector with «cuts», or an environment sector which is not rectangular, with or without «cuts».

Determination of the optimal view route for an environment sector of the first configuration

The optimal trajectories for the environment sector of such configuration are shown in Figure 4.6 and 4.7. Analysis of these figures and other results of the mathematical simulation for

The optimal trajectories for the environment sector of such configuration are shown in Figure 4.6 and 4.7. Analysis of these figures and other results of the mathematical simulation for finding the optimal view route gives a possibility to create an algorithm to obtain an optimal view route for a rectangular sector without «cuts». The fast algorithm for the optimal view route for a rectangular sector without «cuts» is described below.

Consider that m is even; the first subsector has coordinates i=1 and j=1 (see Figure 4.4). Initial data for this algorithm are i, j.

1. To scan a subsector with coordinates i=2, j=1.

2. To scan a subsector with coordinates i=2, j=2.

.

.

.

n-1. To scan a subsector with coordinates i=2, j=n-1.

n. To scan a subsector with coordinates i=3, j=n-1.

n+1. To scan a subsector with coordinates i=3, j=n-2.

.

.

.

2+2(n-2) To scan a subsector with coordinates i=3, j=1.

3+2(n-2) To scan a subsector with coordinates i=4, j=1.

4+2(n-2) To scan a subsector with coordinates i=4, j=2.

.

.

.

k-(n-1)-(m-1)-(n-2) To scan subsector a with coordinates i=m, j=1.

.

.

.

k-(m-1)-(n-1) To scan subsector with coordinates i=m, j=n.

.

.

.

k-(n-1) To scan a subsector with coordinates i=1, j=n

.

.

.

k-1 To scan a subsector with coordinates i=1, j=2.

k To scan a subsector with coordinates i=1, j=1.

The offered algorithm provides the view time for the environment sector – k, that is this algorithm ensures an optimal solution. It includes k steps and does not require any calculation, hence, the optimal view route can be computed very quickly. This fact permits one to apply the offered algorithm for determination of the optimal view trajectories for arbitrary information channels of an MRS. In particular, the offered algorithm can be used for radio-locator channels of an MRS.

Determination of the optimal view route for an environment sector of the second configuration

In contrast to the above considered configuration, in this case, it is impossible to offer a closed trajectory of view equal to k. Different types of trajectories can ensure a view time equal to k+0.5. Optimal trajectories of environment sectors with 3×9 and 5×11 subsectors, are shown in Figure 4.8 a-b respectively. All these trajectories ensure a view time equal to k+0.5.

The fast algorithm for the optimal view route for a rectangular sector without «cuts» is described below. Consider that the first subsector has coordinates i=1 and j=1 (see Figure 4.4). The initial data for this algorithm are i, j.

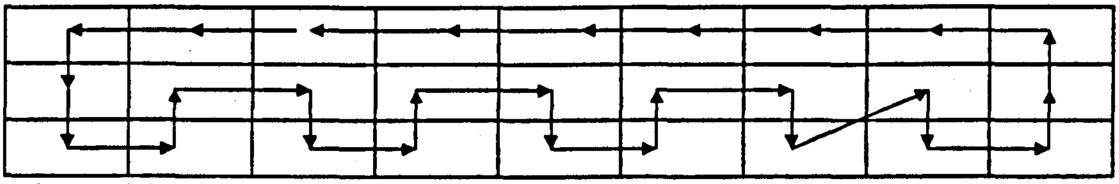


Figure 4.8 a The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=1 and k_3=1.5 for Configuration of a Given Sector - 3×9

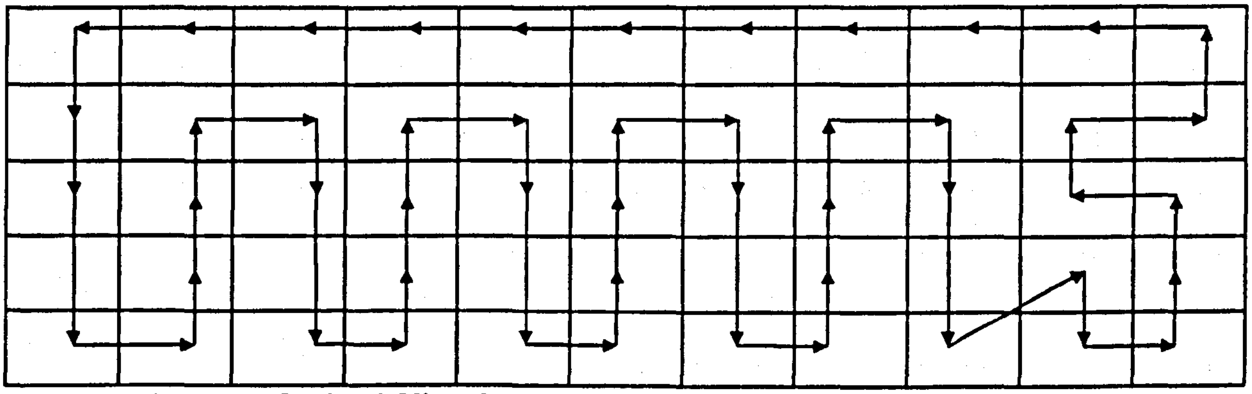


Figure 4.8 b The Optimal View Route with Implementation the Weight Coefficients - k_1=1, k_2=1 and k_3=1.5 for Configuration of a Given Sector - 5×11

1. To scan a subsector with coordinates i=2, j=1.

2. To scan a subsector with coordinates i=2, j=2.

.

.

.

n-1. To scan a subsector with coordinates i=2, j=n-1.

n. To scan a subsector with coordinates i=3, j=n-1.

n+1. To scan a subsector with coordinates i=3, j=n-2.

.

.

.

2+2(n-2) To scan a subsector with coordinates i=3, j=1.

3+2(n-2) To scan a subsector with coordinates i=4, j=1.

4+2(n-2) To scan a subsector with coordinates i=4, j=2.

.

.

.

k-2(n-1)-(m-1)-3 To scan a subsector with coordinates i=m-1, j=1.

k-2(n-1)-(m-1)-2 To scan a subsector with coordinates i=m-1, j=2.

k-2(n-1)-(m-1)-1 To scan a subsector with coordinates i=m, j=1.

k-2(n-1)-(m-1) To scan a subsector with coordinates i=m, j=2.

.

.

.

k-(m-1)-(n-1) To scan a subsector with coordinates i=m, j=n.

.

k-(m-1)-(n-1) To scan a subsector with coordinates i=m, j=n.

k-(m-1)-(n-1)+1 To scan a subsector with coordinates i=m-1, j=n.

.

.

.

k-(n-1) To scan a subsector with coordinates i=1, j=n.

k-n To scan a subsector with coordinates i=1, j=n-1.

.

.

.

k-1 To scan a subsector with coordinates i=1, j=2.

k To scan a subsector with coordinates i=1, j=1.

The offered algorithm provides a view time of the environment sector – k+0.5, that is this algorithm ensures the optimal solution. It includes k steps and does not require any calculation, hence, the optimal view route can be computed very quickly. This fact permits one to apply the offered algorithm for determination of an optimal view trajectory for arbitrary information channels of an MRS. In particular, the offered algorithm can be used for the radio-locator channels of an MRS.

Determination of the optimal view route for an environment sector of the third configuration

Sectors of environment of this configuration include all non-standard configurations of environment sectors. Fast development of onboard computers, enhancement and complication of the stated tasks for an MRS require to application of a digital map of the environment for navigation purposes and for target search. The task of target search in a given known sector of an environment has appeared often. In this case, probabilities of target appearance are different. For the example, in question, then the sector of an environment includes beaten track and a marshy part, on which the probability of target appearance is small, it is therefore inexpedient to scan all subsectors of the environment sector. Possible configurations of an environment sector to view are shown in Figure 4.9 a-c. Obviously, it is most difficult to consider all variants of configuration for environment sectors. Hence, it is not possible to synthesis typical optimal trajectories of view for such configuration of environment sectors. It is much better to use the algorithm, offered in section 4.2.2, for determination of the optimal view route for the environment sectors with such configuration.

It was shown in section 4.3.1 that this algorithm can be applied for CCD vision systems of an MRS and it inefficient of used in radio-locator channels of an MRS.

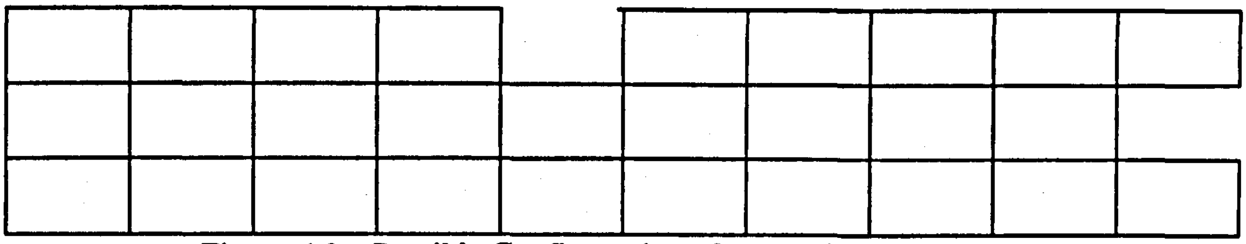


Figure 4.9 a Possible Configuration of an Environment Sector

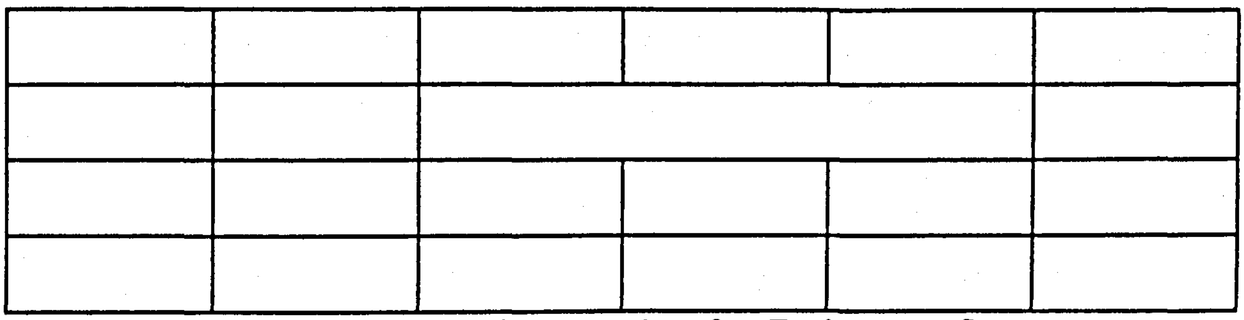


Figure 4.9 b Possible Configuration of an Environment Sector

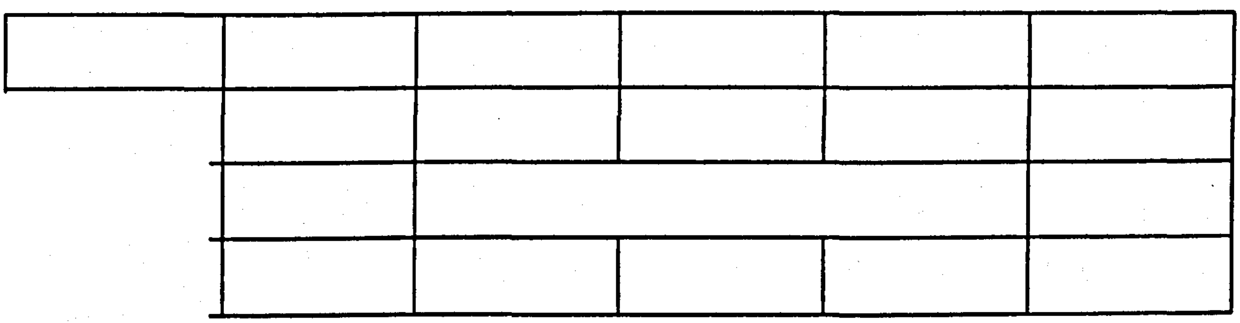


Figure 4.9 c Possible Configuration of Environment Sector

4.4 SUMMARY

The algorithm for an optimal view trajectory for a given environment sector has been offered in this chapter. It provides a closed trajectory of view without «loops», maximum value and equality of scanning frequencies for some subsectors. The offered algorithm permits a to reduction in calculation time and to increase the view frequency. The conclusion reached is the possibility for the practical application of the offered algorithm for an MRS CCD vision system.

The fast algorithms for definition of the optimal view route for rectangular sectors of an environment are offered. These algorithms reduce the calculation time of an optimal view route for a given environment sector. It allows for extended area of application for these algorithms. These algorithms can be applied for determination the optimal view route for a radio-locator channel of an MRS.

CHAPTER 5

SYNTHESIS OF THE CONTROL ALGORITHMS FOR THE SCANNING SYSTEM

5 SYNTHESIS OF THE CONTROL ALGORITHMS FOR THE SCANNING SYSTEM

5.1 REVIEW OF PREVIOUS WORK FOR CONTROL REDUNDANT MECHANISM

Redundant mechanisms are devices with more degrees of freedom (DOF) than required for the realisation of a prescribed task in a work environment. Industrial robotics is a common field in which redundant mechanisms-manipulators may be used. Redundant DOFs for manipulators are very useful in order to avoid obstacles, singular positions, avoidance of joint limit and joint torque minimisation. But such manipulators also have significant disadvantages. The main problem being the choice of reasonable control and suitable initial mechanism configuration for undertaking of the given task. Much efforts have been devoted to the development of efficient procedures to solve the inverse kinematic problem for redundant manipulators. The main difficulty in solving such problems, according to *Mao and Hsia* [1997], *Martin D.P* [1989], is that they are highly non-linear and multiple solutions exist. Each solution provides a different manipulator posture. For such manipulators the numbers of inverse kinematic solutions are infinite, and, in general, close form solutions are impossible to find.

Wu [1994] has suggested that this problem can be divided into two categories: (1) the problem of movement along a prescribed path and (2) the movement between two end points. Recently several solutions to the first problem have been offered. The second problem presents difficulties. No definitive solutions exist at the present time, in this chapter an approach to resolving the particular problem is presented.

There are two basic approaches for the inverse kinematic solution of a redundant manipulator: one is local optimisation approach, and the other is the global optimisation method, after *Cho et al.* [1995].

Many papers deal with local control problems. Some researches have involved the resolution of the movement using the pseudo-inverse of the Jacobian matrix. *Whitney* [1969] offered the approach to determine a joint velocity using the pseudo-inverse matrix and then to incrementally determine a joint displacement. *Liegeois* [1977] proposed a modification to the pseudo-inverse approach to avoid joint at its limit. *Oh et al* [1984] present a numerical procedure for solving the inverse kinematic problem with using the constraints of the manipulator. *Baker and Wampler II* [1988] have reported that main disadvantage of these schemes based on the pseudo-inverse of the Jacobian matrix is the difficulty to provide conservative joint motion for a task of tracking a closed path in task space. *Baillieul* [1985] proves that without further modification, the pseudo-inverse method cannot avoid singularities.

Sciavicco and Sicilliano [1988] present a close-loop algorithm. *Chang* [1986] presents a closed form solution which resolves the redundancy of the mechanism at a joint angle level instead of at a joint velocity level. Unfortunately, the classes of manipulators for which the closed form solutions are guaranteed are very limited. *Goldenberg et al* [1985] present a generalised solution for inverse kinematics. They have used modified a Newton-Raphson iterative technique to solve the system of non-linear kinematic equations. The main disadvantage of this method is the necessity of a good initial guess in order to ensure convergence to the desired point.

Some researchers offer to resolve this problem by using a local torque minimisation method. Its initial formulation was proposed by *Hollerbach* and *Suh* [1987]. *Kazerounian* and *Nedungadi* [1987] present an alternative solution. They have used Lagrangian multipliers. The main disadvantage of these methods is the instability of these schemes. *Ma* and *Nenchev* [1996] consider control over the speed of the homogenous joint. They attempt to attain correct formulation of problem and to avoid instability.

Kazerounian and Wang [1988] show the differences between the local and the global optimal control methods. They have shown the cost function is smallest for global optimal control in comparison with the local optimal control methods. The global optimisation method has intrinsic merits over the local optimisation for it optimises an integral cost criterion providing a global optimal solution from the whole task information. Global optimal control methods need all the required data before the motion can be determined. Such global optimal control methods are based both on Pontryagin's maximum principle or on the calculus of variation.

Nacamura and *Hanafusa* [1986] solve the global optimisation problem by means of Pontryagin's maximum principle. In this work the cost function is the norm of the speeds over the entire integration interval. Authors show a cost functional where they have used a manipulability index. This index has been added to the standard velocity norm function. *Hollerbach and Suh* [1987] offer a global torque optimisation method using the parameterisation of the redundancy of a manipulator. *Kyriakopoulos* and *Saridis* [1988] present a control algorithm, which minimises jerks, the first time derivatives of the relative accelerations of the links. This method is based on idea of minimisation of the changes of the manipulability index in the process of trajectory generation. *Gothlin et al* [1995] offer global optimal control, which is based on calculus of variations with given boundary constraints. *Kazerounian and Wang* [1988] propose global versus local resolution optimisation in resolving the redundancy by means of minimisation of the norm of joint velocity. *Martin et al* present a global optimisation method for cyclic tasks. They have used Euler-Lagrange equations to obtain the necessary periodic boundary conditions. *Won J. H. et al* [1993] propose a unified approach to obtain an optimal joint velocity, derived by using the necessary condition for optimality. The optimal solution they have achieved with minimum number of first-order differential equations which requires a minimum search dimension.

Recently, some researchers have attempted to use neural-networks (NN) for solution of the inverse kinematic task for a redundant mechanism. Multi-layer NN can form continuous non-linear mapping from one domain to another. *Lee* and *Kil* [1990] present a multi-layer NN with sinusoid activation functions to model a given forward kinematic mapping. *Guez* and *Ahmad* [1988], *Hou* and *Utama* [1992] propose a multi-layer NN to simulate inverse kinematic mappings. They have shown that for manipulator with three DOFs, elbow-up and elbow-down inverse kinematic solutions can be trained with the choice of different initial weights in the NN.

A number of different methods based on the Kohonen's "self-organisation mapping algorithm", Hopfield's network and the multiplier perception model. *Guo* and *Cherkassky*

[1989] propose a closed loop dynamic system for redundant manipulators. The local minimum of the Hopfield network energy function is achieved by a process of re-estimation of the neurone weights. *Lin Z. et al* [1990] present a counter-propagation network and a counter-propagation network architecture. It is based on splitting Kohonen layer model and a modified training procedure. *Kieffer et al* [1991] resolve the same problem by means of Kohonen self-organising mapping network. They have formed the Window-Holf error correction rule and have used the Euclidean distance between the desired and the actual end-effector position as a convergence criterion. *Dermatas et al* [1996] present an error-back propagation algorithm. The local optimal problem is solved by adding white noise to the weights of the at the neurones output layers.

The main disadvantage of these methods is its complexity and ability to realise it in real time for space redundant mechanisms. To resolve this problem the follow algorithm has been offered.

5.2 CHOICE AND VERIFICATION OF APPROACH FOR THE CONTROL FORMING

To synthesise control algorithm for mechanism with kinematic redundancy it is necessary to choose quality criteria and to minimise (or to maximise) it. The minimum of the instantaneous sum of the scanning system drive power is chosen as such criteria. The reasoning for this is as follows:

the degeneration area of a scanning system appear as a result of the power limitation for joint conducting drives according to *Kochergin V.* [1988].

Conducting drives of large power have high mass-inertia characteristics (*Pelpor S.*

[1989]).

It is necessary to take into consideration the autonomy of an MRS power supply (usually it is limited) during the synthesis for the optimal control of the scanning system conducting drives.

Hence, the criteria of optimum for the scanning system (generalised power of the mechanism by *Nesterov V.* [1994]) can be introduced in the followed form:

$$K(t)=\sum_{i=1}^{3} P_{li}(t) \tag{5.1}$$

where $P_{1i}(t)$- is the instantaneous power of the conducting drive motor for i-th degree of the scanning system.

Traditionally, electric motors are applied as the conducting drives of a scanning system (*Belyansky P.* and *Sergeev B.* [1980], *Bairashevsky A.* and *Nicheporenko N.* [1982]). *Kochergin V.* [1988] showed that during analysis of the power balance equations for a mechanism the power consumption of each drive is determined by twice the kinematic energy of «moving masses». It is possible to write for each individual drive whilst considering the separate interval of quantification, the quality criteria 5.1 is minimised inside it:

$$P_1(t)=\Delta A_1/\ \Delta t=2\ \Delta T/\ \Delta t,- \tag{5.2}$$

where Δ t- is time of interval of quantification;

ΔA_1- is energy expenditure of conducting drive on interval of quantification;

Δ T - is increment of kinetic power for «moving masses» on interval of quantification.

According to *Nesterov* V. [1994] the kinetic power of each drive for the scanning system is determined by the following equation:

$$T(t)=\frac{d(t)(\dot{q}(t))^2}{2},- \tag{5.3}$$

where – $\dot{q}$ -is the velocity of rotation for output shaft;

d – is the reduced moment of inertia for the «moving masses» of a scanning system relative to the axes of the output shaft.

Equation 5.3 has been transformed to the following form:

$$P_1(t)=\frac{1}{\Delta t}(d(t)(\dot{q}(t))^2-const(t)) \tag{5.4}$$

Taking into consideration the equation 5.4, the criteria equation 5.1 converts to the following form:

$$K(t)=\frac{1}{\Delta t}\sum_{i=1}^{3}(d_i(t)(\dot{q}_i(t))^2-const(t),- \tag{5.5}$$

where const(t)- is some constant, which does not depend on $\dot{q}_i(t)$ and is determined by the kinetic power of the mechanism at the interval of quantification.

Taking into consideration that the functional dependency of the two generalised coordinates

q_3, q_2 on the redundant coordinate q_1 is a characteristic for a scanning system with kinematic redundancy (see the section 3.2). Hence, the (5.5) criteria can be converted to the next function.

$$F(\dot{q}_i, \dot{q}_2, \dot{q}_3) = \sum_{i=1}^{3} d_i (\dot{q}_i(t))^2, - \quad (5.6)$$

where $[\dot{q}_2,]=f(\dot{q}_i(t))$ - the minimisation of which in the current moment of time (relative to the velocity of the redundant coordinate) permits one to formulate control for a scanning system with kinematic redundancy (optimal by way of the minimisation of the equation (5.1)). That is, this allows a minimum instant sum of the scanning system drives power.

5.3 Optimal Control Law of Scanning System for Mobile Robotic System

The obtained quality function 5.6 allows to define an optimal control law for the scanning system drives when surveying a given environment sector with the given kinematic characteristics.

To solve the set task it is possible to use the pseudo-inverse matrix methods. The solution of the direct kinematic task can be described in the next form, as expressed previously in section 3.2:

$$V^0 = J(q)\, \dot{q}_i, - \quad (5.7)$$

where V^0 – is the vector of the linear velocity for the goal point in SCS;

$q=[q_1, q_2, q_3, D]$ – is the vector of the generalised coordinates for the scanning system;

$J(q)$ – is the rectangular Jacobi matrix of the system (it is derived from the expression

$V^0= \dot{T}_4\,[0, 0, 0, 1]$.

The following expression follows from the expression 5.7, using term of pseudo-inverse matrix $J(q)$.

$$\dot{q} = [J(q)]^{+}V^0, - \qquad (5.8)$$

where $[J(q)]^{+}$ - is the designation of the pseudo-inverse matrix $J(q)$.

Gantmacher F. [1988] has shown that the equation 5.8 determines the vector $\dot{q}$ thus, the norm of it is minimised. In other words, the equation 5.8 ensures a minimum of the following expression:

$$F_1(\dot{q}) = \sum_{i=1}^{3}(\dot{q}_i)^2 + (\dot{D})^2, - \qquad (5.9)$$

The current value of $\dot{D}$ in the expression 5.9 is determined only by motion parameters and it does not depend on value $\dot{q}_i$ (see section 3.2), with neglect of the geometric size of the scanning system. Hence, equation 5.8 minimises the follow expression:

$$F_2(\dot{q}_i, \dot{q}_2, \dot{q}_3) = \sum_{i=1}^{3} (\dot{q}_i(t))^2, - \qquad (5.10)$$

To provide minimum for function 5.6, equation 5.7 has been converted to the following form:

$$V^0=J(q)\dot{q} = J(q)[Z(d)]^{-1}Z(d)\dot{q} \text{ ,-} \tag{5.11}$$

where $Z(d)=\text{diag}[(\sqrt{d_1},\ldots,\sqrt{d_N},1)-(N+1)\times(N+1)$ – is the diagonal matrix of coefficients of equation 5.6.

It follows from equation 5.11:

$$\dot{q} = [Z(d)]^{-1}[J(q)[z(d)]^{-1}]^{+}V^0\text{,-} \tag{5.12}$$

where $[J(q)[z(d)]^{-1}]^{+}$ - is the pseudo-inverse matrix of the modified Jacobi matrix.

According to the property of pseudo-inverse matrixes, the expression (5.12) determines the minimum of relation of the next form (after modification of Jacobian system):

$$F_3(\dot{q}) = \sum_{i=1}^{3} d_i(\dot{q}_i)^2 + (\dot{D})^2. \tag{5.13}$$

Control 5.12 ensures the minimum of the function 5.5 with neglect of geometric size of scanning system.

The practical realisation of the optimal control 5.12 presents computing difficulties. Greville's method is used to determination of pseudo-inverse matrixes. But, analytical calculation of the equation 5.12 requires a very large value of RAM for an MRS on-board computer. Moreover, the problem of «infinitesimal» values in a digital computer appears.

The task of solving the control 5.12 is reduced to task of the direct minimisation of the functional 5.5 relative to the velocities in the redundant degrees of the mechanism, when the analytical solutions of inverse kinematic tasks relative to the generalised coordinates of the scanning system are known.

It was shown in the section 3.2, that the following relations are correct for mechanismss with three degrees of freedom:

$$\left.\begin{aligned} \dot{q}_2 &= s_{21}\dot{q}_1 + s_{22} \\ \dot{q}_3 &= s_{31}\dot{q}_1 + s_{32} \end{aligned}\right\}, \tag{5.14}$$

where:

$$\left.\begin{aligned} s_{21} &= \frac{Cq_2Cq_3}{Sq_3}, \\ s_{22} &= \frac{-Cq_1Cq_2V_x^0 - Sq_1Cq_2V_y^0 + Sq_2V_z^0}{DSq_3}, \\ s_{32} &= \frac{(-Sq_1Sq_3 - Cq_1Sq_2Cq_3)V_x^0 + (Cq_1Sq_3 - Sq_1Sq_2Cq_3)V_y^0 - Cq_2Cq_3V_z^0}{D}. \end{aligned}\right\} \tag{5.15}$$

The functional 5.5 has been minimised taking into account the equation 5.14. The expression 5.5 belongs to the class of minimisation of functions with one variable. Great attention is devoted to this question in the literature (*Ivanov* V. and *Faldin N.* [1981], *Ganshin* G. [1987] and so on). The following relation follows from application for the necessary condition of a minimum to the equation 5.5:

$$\frac{dF(\dot{q}_1,\dot{q}_2,\dot{q}_3)}{d\dot{q}_1}=0,- \tag{5.16}$$

and

$$d_1\dot{q}_1+d_2\dot{q}_2\frac{\partial\dot{q}_2}{\partial\dot{q}_1}+d_3\dot{q}_3\frac{\partial\dot{q}_3}{\partial\dot{q}_1}=0. \tag{5.17}$$

It follows, taking into account equation 5.14:

$$\dot{q}_1=\frac{d_2s_{21}s_{22}+d_3s_{31}s_{32}}{d_1+d_2(s_{21})^2+d_3(s_{31})^2},- \tag{5.18}$$

It is easy to show, that the sufficient condition of the minimum is correct for equations 5.14 and 5.18:

$$\frac{d^2F(\dot{q}_1,\dot{q}_2,\dot{q}_3)}{d(\dot{q}_1)^2}\rangle 0,- \tag{5.19}$$

Thus, the equation 5.18 express the desired control for the redundant degree of mobility for the scanning system with three mutually perpendicular axes.

Substitution of equation 5.14 to the expression 5.18 gives the final control law for the considered kinematic scheme.

$$\left.\begin{aligned}
\dot{q}_1 &= \frac{d_2 s_{21} s_{22} + d_3 s_{31} s_{32}}{d_1 + d_2 (s_{21})^2 + d_3 (s_{31})^2},\\
\dot{q}_2 &= \frac{s_{22}(d_1 + d_3 (s_{31})^2) - d_3 s_{21} s_{31} s_{32})}{d_1 + d_2 (s_{21})^2 + d_3 (s_{31})^2},\\
\dot{q}_3 &= \frac{s_{32}(d_1 + d_2 (s_{21})^2) - d_2 s_{21} s_{31} s_{22})}{d_1 + d_2 (s_{21})^2 + d_3 (s_{31})^2}.
\end{aligned}\right\},- \qquad (5.20)$$

The final equation for determination: $\dot{q}_1$ is obtained with the substitution of the coefficients s_{21}, s_{22}, s_{31}, s_{32} into the equation 5.20:

$$\dot{q}_1 = \frac{Cq_3(d_2(Cq_2)^2 + d_3(Sq_2Sq_3)^2(Cq_1V_x^0 + Sq_1V_y^0)}{D(d_1(Sq_3)^2 + d_2(Cq_2Cq_3)^2 + d_3(Sq_2Sq_3)^2} + \frac{d_3Sq_2(Sq_3)^2(Sq_1V_x^0 - Cq_1V_y^0) + Sq_2Cq_2Cq_3(d_3(Sq_3)^2 - d_2)V_z^0}{D(d_1(Sq_3)^2 + d_2(Cq_2Cq_3)^2 + d_3(Sq_2Sq_3)^2} \qquad (5.21)$$

Thus the equation permitting one to find the $\dot{q}_1$ value of the redundant degree velocity has been obtained. That is, the optimal control, which minimises the instantaneous sum of the scanning system drive power. That is the set task is solved.

Notation.

There are some methods to define values of the functions d_1, d_2, and d3 in the equation 5.11. The value of d_i=1,2 ..., N (where N – is the number of mechanism joints) is called the «effective moment of inertia for the mechanism in relative to generalised coordinate q_i» in the literature devoted to the control of complex space mechanisms (*Nesterov V.* [1994], *Popov E.* [1978], *Paul R.* [1972]).

Popov E. [1978] offered a calculating algorithm of instantaneous values of q_i relative to the control of joint mechanisms of robot manipulators. To enable fast processing on an on-board computer it is desirable to consider the values of d_i as constant (it is not dependent on time and configuration of the mechanism) in equation 5.21. In this case, the values of d_i can be a maximum, or a mean probability, or values of the effective moments of inertia near a «dead zone'. This question requires additional research in each separate case. The problem of the choice coefficient d_i connected with the fact that they determine as expressed in control 5.21 and therefor it is necessary to simulate work of the scanning system to test the correspondence of the requirements. In this thesis, it is offered to apply as values d_i in the equation 5.21 the values of the moment of inertia for the scanning system mechanism relative to the output shafts of the separate conducting drives (without taking into consideration the connections between the links).

The synthesised optimal control of the scanning system mechanism with kinematic redundancy is resolved relative to the instantaneous values of the velocities of the scanning system joints. It does not solve the question concerning choice of the optimal configuration for the moment of a scanning start.

To determine the values of the generalised coordinates of a scanning system mechanism for the moment of a scanning start, the follow additional condition is offered.

The values of rotation angles in the joints of the kinematic redundant motion degrees of the scanning system are defined from the condition of providing a minimum for the function by substitution of the synthesised control law. It is impossible to solve this task analytically. That is why, research of the equation 5.5 with respect to function q_i (with substitution the equations 5.14, 5.20, 5.21 with taking into consideration decision of the inverse kinematic task into positions – the equation 3.20) has been conducted by mathematical simulation on a computer. The sighting scheme is shown in Figure 3.1. The equation below follows according to the scheme shown in Figure 3.1.

$$V^0 = V + w_{\Sigma} \times r_{04}^{-1} - \dot{r}_0^{-1}, - \quad (5.22)$$

where ω_Σ - is the vector of velocity sum for an MRS rocking in ASC.

The defining influence forces the second summand in the equation 5.22 (which is a corollary of MRS rocking), with scanning of environment on distances of the order of 1 km. Taking in the equation 5.22 the values $r_{04}^{-1} = \dot{r}_0^{-1} = 0, V = 0$. It is derived with neglect of the geometric size of the scanning system:

$$\left|V^0\right|_{max} = \left|w_{\Sigma}\right|_{max} D, - \quad (5.23)$$

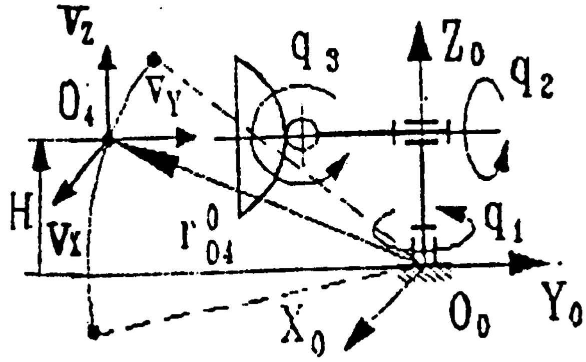
The research scheme is shown in Figure 5.4. Mathematical simulation of the scanning system work has been carried out. The following relations are introduced

$$V_x^0 = V_y^0 = V_z^0, - \quad (5.24)$$

where H- is the height of scanning in SCS,

it is accepted D=2000 m.

The results of mathematical simulation are shown in Figure 5.2 for heights H={3,300,1000} (m). The change step of the course angle for the scanning system is accepted equal to $\Delta q_1 = \pi / 200\,(radian)$.



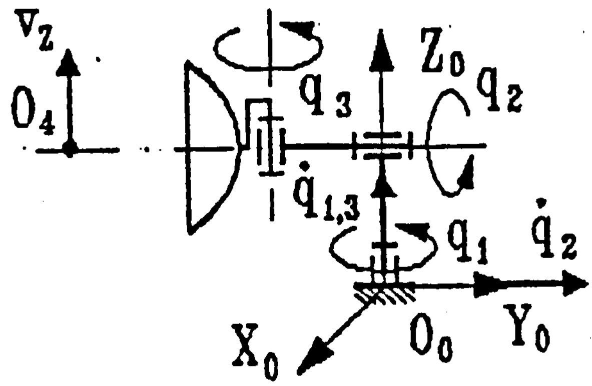


Figure 5.1 Structure Scheme of Scanning System Figure 5.3 Degeneration Position of Scanning System

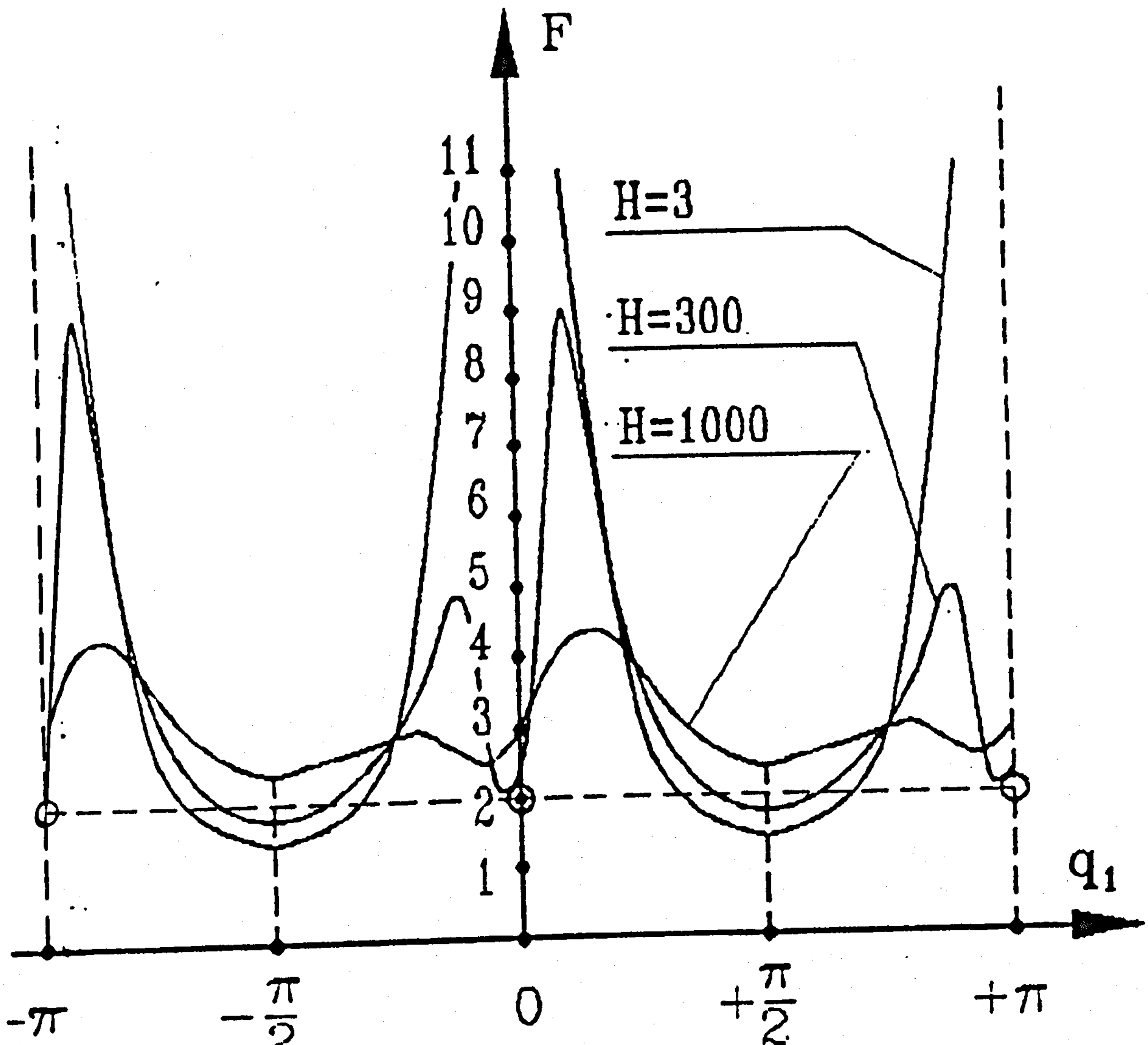


Figure 5.2 Simulation Results

The analysis of received results permits one to conclude that the global minimum of the function is achieved (in all cases) in the configuration, when the rotation axis of angle q_2 is perpendicular to the projection of the radius-vector for the goal point on the plane $(O_0X_0Y_0Z_0)$ in SCS. Hence, condition for definition of the initial configuration for scanning is determined by the following expression for a scanning system with three mutually perpendicular axis.

$$q_1 = \arctan(\frac{y_{04}^0}{x_{04}^0}). \tag{5.25}$$

Moreover, it follows from Figure 5.2, that the value of the function tends to infinity in the case when a goal point approximates to the rotation axes q_2. In this case, the values of all the required velocities tend to infinity. But, there is a local minimum, which corresponds to the control algorithm of a two-degree scanning system with azimuth-angle of position axis. The reason of degeneration of the function is the following. Analysis of possible degeneration conditions of the equation 5.20 permits one to conclude that required rotary velocities in the joints of a scanning system with three mutually perpendicular axes can be degeneralised when:

$$\left.\begin{aligned} q_2 &\to \frac{\pi}{2}, \\ q_3 &\to 0. \end{aligned}\right\} \tag{5.26}$$

It appeared with V_z^0 and it is developed from the equation 5.20. The degeneralised configuration of the scanning system is shown at Figure 5.3. In this case, the scanning system loses one degree of mobility and it can not realise the desired motion along axis (O_0X_0).

In practice, the possibility of continuous scanning of a scanning system with three degrees is defined by a step quantification and by the rocking parameters of the MRS .

The created algorithm of the optimal control synthesis for the drives of an MRS scanning system with kinematic redundancy includes the following steps:

1. To determine control, which provides a minimum of the functional 5.5 using the analytical decision of the inverse kinematic task of velocities

2. To calculate the values of weight coefficients d_i.

3. To define the additional condition for the scanning system configuration at the moment of scanning start, as a result of the minimisation of the function relative to the values of the generalised coordinates of redundant degree of mobility with control under the synthesised control law.

5.4 Recommendations for Practical Realisation of Optimal Control of Mobile Robotic System Scanning System in an On-board Computer

According to *Bessekersky A.* and *Fabrikant E.* [1968] in real scanning systems the degrees of mobility for scanning systems are closed with the main feedback from the rotation angles of the drive output shafts. Hence an MRS on-board computer has to form the vector of action for the next form at the current moment of time.

$$q_f=[q_1, q_2, q_3]^T. \qquad (5.27)$$

where q_f is the vector of the desired generalised coordinates.

Approaches for synthesised control algorithm realisation in an on-board computer (the equations 5.20 and 5.21) are considered below. The main aim to is form the control 5.27. The following vectors have been introduced:

$$\left.\begin{aligned} q_c &= [q_{c1}, q_{c2}, q_{c3}]^T, \\ d &= [d_1, d_2, d_3]^T, - \end{aligned}\right\} \qquad (5.28)$$

where Δt - is the step of quantification; D – is the distance to the goal point; r_{04}^0, V^0 - are the radius-vector of the Cartesian coordinates for the goal point and its derivative in respect to time in the SCS.

The direct realisation of the control laws 5.20, 5.21 in the on-board computer is described by the structural scheme shown in Figure 5.4 under known values of q_f, d, D, V^0. Which are calculated in Block of Forming Velocities (BFV) the values $\dot{q}_1, \dot{q}_2, \dot{q}_3$ according to the equations 5.20 and 5.21.

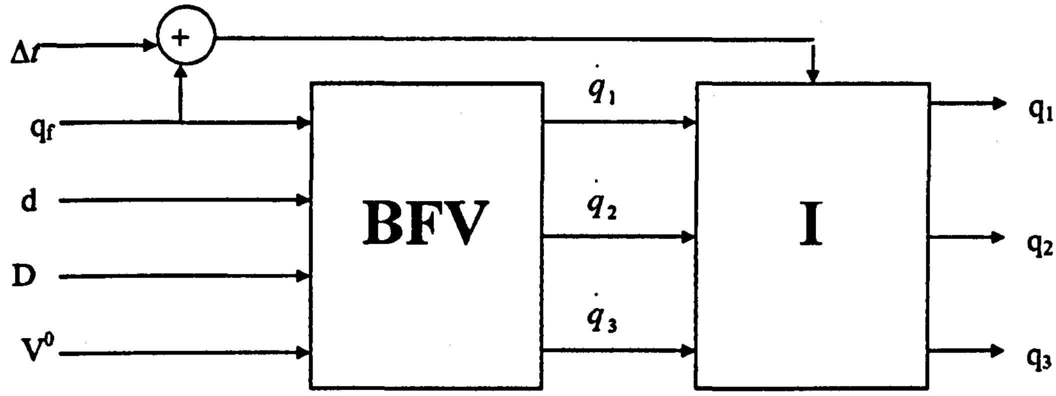


Figure 5.4 First Technical Realisation Scheme of Developed Control

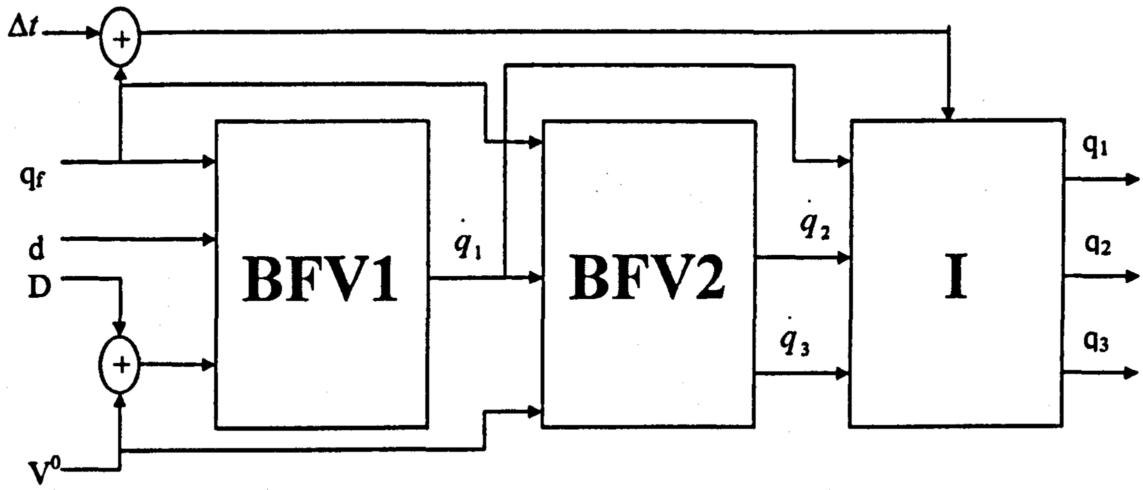


Figure 5.5 Second Technical Realisation Scheme of Developed Control

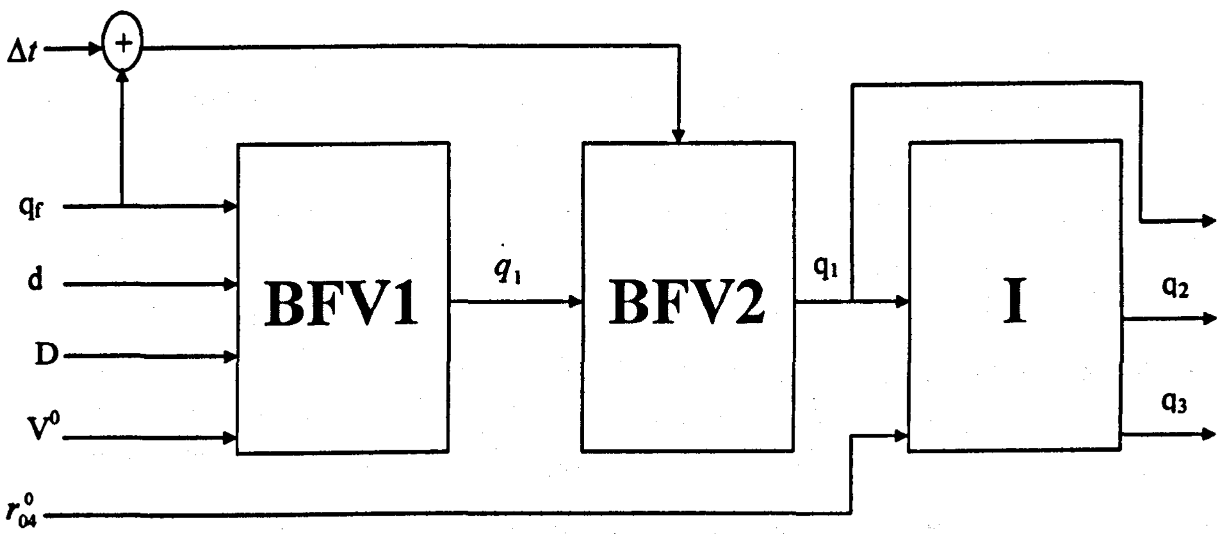


Figure 5.6 Third Technical Realisation Scheme of Developed Control

The received equations are integrated in the block of Integrating (I) taking into account q_c and Δt.

To facilitate on-board computer fast action, it is possible to apply the analytical solutions of the inverse kinematic task in velocities (see equation 3.34). In this case, the control algorithm has the following form – see Figure 5.5. The value of $\dot{q}_1$ is calculated in the block (BFV1) due to known values q_f, d, D, V^0, according to the equation 5.21. The values of $\dot{q}_2, \dot{q}_3$ are computed in the block (BFV2) due to the $\dot{q}_1$ calculated value and known values q_f, d, D, V^0, are according to equation 3.32. The received values are integrated in the block (I) taking into consideration values q_c and Δt.

This algorithm provides the same accuracy of calculation q_f in equation 5.27 as that in scheme 5.4, but the calculation time of scheme 5.5 is less than the calculation time scheme 5.4 by 2.5 time.

Application of scheme 5.6 can bring further enhancement for fast action. This algorithm uses the analytical solution of the inverse kinematic task in positions. The value of $\dot{q}_1$ is calculated in the block (BFV1) due to the known values q_f, d, D, V^0, according to equation 5.21. The received values are integrated in the block (I) taking into consideration values q_c and Δt. The resulting q_1 transfers to the Block of Forming Angles (BFA), where the values q_{2c} and q_{3c} are computed taking into consideration r_{04}^0 according to equation 3.20.

This algorithm ensures a calculation time less than calculation time of scheme 5.5 by 1.8 times, and it requires the minimum of possible resources for fast action for an on-board computer. But, in this case, the components q_{2c} and q_{3c} of the vector of setting actions include extra errors dictated with the function 3.29 dependencies. The question of application of the technical realisation scheme for optimal control in an MRS onboard computer has to be solved in relation to individual requirements, accuracy and fast action.

5.5 SUMMARY

The approach for control using the criteria of minimum instantaneous power and the algorithm of optimal control synthesis for an MRS scanning system mechanism with kinematic redundancy have been created.

The developed algorithm permits one to formulate control, which ensures «dead zones» of closed types.

Recommendations for the practical realisation of the developed algorithm in the MRS onboard computer have been offered. Three different schema for practical realisation of the developed control algorithms are offered.

CHAPTER 6

EXPERIMENTAL RESEARCH

6. EXPERIMENTAL RESEARCH

The results of the experimental research are presented in this chapter. Experimental verification of the capacity for work of the synthesised optimal control for a three degrees scanning system has been obtained by experiment. The reliability of the mathematical model with respect to external disturbances on the base of the scanning system when the MRS is moving is established. The advantages of a three degree scanning system relative to two degrees scanning systems in the survey mode of an environment sector when the MRS moving is also shown.

6.1 TEST OF THE MATHEMATICAL MODEL FOR EXTERNAL DISTURBANCE ON THE BASE OF SCANNING SYSTEM WHEN THE MOBILE ROBOTIC SYSTEM IS MOVING

To test mathematical model of consideration the external disturbance developed in the second chapter the experimental research has been carried out using real road conditions.

6.1.1 Object of test

The wheel chassis of an MRS has been chosen as the object of the test. A photograph of the robot is represented in Figure 6.1. The characteristics of the MRS are shown in Table 6.1.

Parameter	Dimension	Value
Mass of MRS	*kg*	3500
Inertia moment of MRS in relative to cross axis passing through mass centre of MRS	$kg \times m^2$	5950
Pressure air of tyres of MRS	*atmosphere*	2.5
Length of supporting part of MRS chassis	*m*	3.4

Table 6.1 Main Characteristics of the Chosen Mobile Robotic System

6.1.2 Conditions and order of test conducted

The tests have been carried out on a special experimental road. This road is situated in Leningrad region. MRS has moved along road full of put-holes travelling 500 meters. Heavy track and wheel vehicles had earlier travelled on this road. This road consists of a hard nondeformed surface. Irregularities in the road are form wave situated approximately equi-spaced step. The soil is loam with parts of gravel. Intensity of action of this part of the road is I=38 sm^2/m. Analytical expression of spectral density of this road is the following:

$$S = \frac{2}{\pi}\int_{o}^{\infty} k(\tau)\cos(w\tau)d\tau. \qquad (6.1)$$

where S is the spectral density;

w is the frequency.

The longitudinal profile of road is described by the next expression:

$$\left.\begin{aligned} P(\tau) &= e^{-0.4500(\tau)}\cos(0.4140\tau), \\ S_h(w) &= 14.82\left(\frac{1}{0.2074+(w+0.414)^2}+\frac{1}{0.2074+(w-0.414)^2}\right). \end{aligned}\right\} \quad (6.2)$$

where P(τ) is the correlation function;

S_h is the spectral density of the longitudinal profile. It is represented in Figure 6.2.

The angle of the transverse inclination for this road is described by the following equations

$$\left.\begin{aligned} p(\tau) &= 0.815e^{-0.032(\tau)}\cos(0.089)+0.815e^{-0.359(\tau)}\cos(1.046), \\ S_{\psi} &= 0.1703\times10^{-4}\left(\frac{1}{0.001+(w+0.089)^2}+\frac{1}{0.001+(w-0.089)^2}\right)+ \\ & 0.4338\times10^{-4}\left(\frac{1}{0.1289+(w+1.046)^2}+\frac{1}{0.1289+(w-1.046)^2}\right. \end{aligned}\right\} \quad (6.3)$$

where S_{ψ} is the spectral density for the angle of the transverse inclination. It is shown in Figure 6.3. The longitudinal profile of the road is represented in Figure 6.4.

The velocities of the MRS during experiment varied from 1 m/s to 9 m/s with 1 m/s steps.

6.1.3 The measured parameters, devices and metrology tools.

The following parameters have been received by an oscillograph, of type K-20-22, during the test.

The velocity of the longitudinal rotary oscillations of the MRS body is measured by the sensor DUS-90. To set the correct initial position of this sensor rotary table KLA-5 is used.

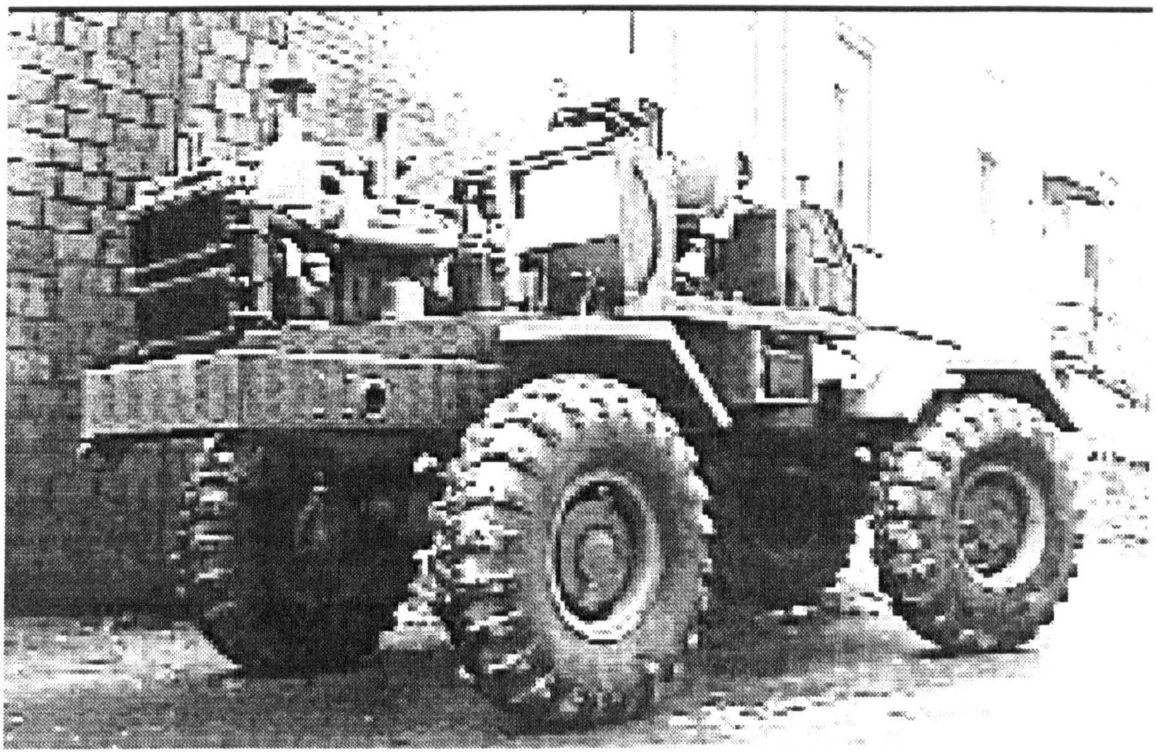


Figure 6.1Object of Test. Wheel Mobile Robotic System of Middle Size.

The accelerations of longitudinal rotary oscillations of the MRS body are measured by the sensor GIUS-17.

The angles of longitudinal rotary oscillations of the MRS body are measured by a potentiometer.

Information from these sensors has been transferred to the onboard computer by means of the analogue digital converter DS2001. Received data have been written by means of using MATLAB SIMULINK software.

The MRS velocity is determined by time of motion along a measured part of the road.

6.1.4 Test results

The results of the experiments provided the value of longitudinal angles and rotary velocities of the MRS as a function of time in the form of data files. They are represented in Appendix B. The calculation of the average values for the necessary effective power drives normalised by the moment of inertia of the mobile masses of the stabiliser (see equation 2.8) has been calculated according to experimental values determined during nine tests of the MRS with different velocities. Further, the value of "A" coefficient has been calculated of equations 2.14 and 2.15. The measured data of maximum values of the rotary speeds and accelerations of external disturbance have been used for calculation A coefficients.

The differences between the "A" coefficient values for different velocities of the MRS do not exceed 5% according to analysis shown in Table 6.2. Thus, verification of the developed model 2.14 and 2.16 has been obtained.

Further the received experimental data have been applied for physical simulation of the optimal control algorithms in laboratory conditions.

Table 6.2 Experimental Results

V m/s	P_{ef}/J_{ef}	A
1	0.00461	0.08276
2	0.03727	0.08321
3	0.1264	0.08343
4	0.2994	0.08345
5	0.5924	0.08392
6	1.0291	0.08414
7	1.6591	0.08478
8	2.5092	0.08525
9	3.7041	0.08689

6.2 Experimental Research of the Synthesised Control Algorithms

The analysis of the quality of the synthesised optimal control for a three degrees scanning system when conducting two modes simultaneously: moving along a given program trajectory and stabilisation for base rocking has been produced. The quality characteristics of a three degrees scanning system have been compared with the characteristics of an equivalent scanning system with two degrees of mobility.

6.2.1 Experimental unit

The scheme of experiment is shown in Figure 6.5. Where 1 - is the controlling computer; 2- is the measuring screen; 3 - is a PUMA robot with the laser diode; 4 - is the TV camera; 5 - a video recorder. The experimental research of the three degrees scanning system has been conducted by the following means. PC computer has been connected with PUMA robot via the special cable. The three upper degrees of the PUMA robot is emulated work of a scanning system with three degree of freedom (fourth, fifth and sixth degrees). The control of the robot is produced by an IBM PC computer. The program of robot control has been written in the Pascal programming language. A laser diode with wave length 650 nanometers has been connected on the end effector of the PUMA robot. The emitted beam falls onto the measuring screen. The width of beam provides a spot of 3 millimetres diameter at the measuring screen. The given program trajectory has been drawn on the paper with a measuring mesh (with 1 millimetre step). The beam of the laser must be moved along this given trajectory. The given velocity of movement has been calculated due to the desired time of scanning for the given sector. The TV camera has been directed to the measuring screen. The image from the TV camera which shows the relative location of the laser light spot with respect to the mesh has been recorded on the video recorder. Simultaneously a signal of the time step throughout the test has been transferred from the computer to the audio input of the video recorder. The aim of this action is to define correspondence between a current position of beam and a desired point of the given trajectory.

6.2.2 Conducting of the test

The line of sight must be moved along the program trajectory (see subsection 6.2.1). This line is given in the coordinate system connected with the robot base. Data of angles and rotary velocities for rotary longitudinal rocking of the MRS is used as initial data for the test. This data has been received experimentally when the MRS moved along the experimental road (see subsection 6.1). This data has been written as file into the controlling computer and has been applied for control of second and third degrees of the robot. Motion of these degrees is a simulated external disturbance to the scanning system (fourth, fifth and sixth degrees of the robot). Information from the sensors of the second and third degrees of the robot has been transferred into the computer as imitation data of the MRS navigation system. The necessary trajectory of sight line in the coordinate system connected with the scanning system base has been formed on the base of information from the MRS navigation system and the given program trajectory. Further, optimal control of scanning system (vector of desired velocities with use of equations 5.21 and 3.32) has been formed with taking into consideration information about the current location of the scanning system (this information is measured by the sensors of fourth, fifth and sixth degrees of the robot). The structure scheme realisation of this algorithm is shown in Figure 5.5 and is shown in Figure 6.8 with more detail. During the control process, a file is formed. It includes the following information: data of desired trajectory (a priori given), external disturbance (measured by the sensors of second and third degrees of the robot), coordinates of the scanning system (measured by fourth, fifth and sixth degrees of the robot).

6.2.3 Processing of test data

After completing the experiments, the information recorded on video cassette has been entered to the computer via special equipment (framegrabber, device to enter the time steps). Information about real location of the sight line in coordinate system connected with the robot base has been added to the file on the base of data of laser spot location at the measuring screen. The difference between the given program and real motions has been calculated using this data file.

6.2.4 Analysis of the experiment results

The accuracy of the synthesised control algorithms has been proved at the test unit described in subsection 6.2.1. External disturbance has been given in the form of data file. It includes value of rotary longitudinal rocking of the MRS when it moves with constant velocity - V_{MRS} on experimental road (see 6.1 subsection). Two files of rotary disturbances have been applied for V_{MRS}=5 m/s and V_{MRS}=8 m/s. The distance between the robot base and the measurement screen has been chosen at 4 meters. The equations of the chosen program motion in the coordinate system connected with robot has the following form:

$$\left.\begin{aligned} x &= 0.05t - 0.75 \\ y &= -4 \\ z &= 0.05t + 0.5 \end{aligned}\right\} \qquad (6.4)$$

The disturbance to the sight line for V_{MRS}=5 m/s and V_{MRS}=8 m/s is shown in Figure 6.6a - 6.7 a respectively. The difference between the given program trajectory and measured trajectory as a result of the test has been calculated into angles and is represented in Figure 6.6b -7.b respectively for V_{MRS}=5 m/s and V_{MRS}=8 m/s. This difference is obtained be use of the synthesised control algorithms for a three degrees scanning system.

To conduct a comparative analysis, analogous tests have been carried out for a two degrees scanning system. In this case, external disturbances and program trajectory have been chosen equal to external disturbances and program trajectory of the previously described test. The external disturbances have been formed by the second and third degrees of the PUMA robot. The scanning system work has been realised by two degrees (fourth and fifth) only. Values of velocities for fourth and fifth degrees of robot have been calculated into the computer using a the typical control algorithm.

The difference between the given program trajectory and measured trajectory as a result of the test, for a two degrees scanning system controlled by the typical algorithm, has been calculated into angles and is represented in Figure 6.6c - 6.7c respectively for V_{MRS}=5 m/s and V_{MRS}=8 m/s. The advantage of a three degrees scanning system, controlled by the synthesised optimal algorithm, in relative to a two degrees scanning system controlled by the typical algorithm, is obvious. To conduct comparative analysis the integral errors of moving along a desired trajectory have been calculated. The calculation results are shown in the Table 6.3.

V_{MRS}, m/s	ε, %	ε, %
5	8.5	13.8
8	11.0	18.1

Table 6.3 Comparison of Integral Errors

The analysis of the obtained results shows the error of motion along a given trajectory when using the three degrees scanning system, controlled by the synthesised optimal algorithm, 163% better in relative to that obtained using two degrees scanning system, controlled by a typical algorithm.

6.3 SUMMARY

The following results have obtained in this chapter.

The reliability of the mathematical model for an external disturbance on the base of scanning system when the MRS is moving is determined. The calculation of error does not exceed a given value.

Experimental verification of the performance of the synthesised optimal control for a three degrees scanning system, which minimises the instant total power of scanning system actuators is obtained. The scanning system can stabilise the external disturbance and realise motion of the sight line along a given trajectory.

The advantages of a three degree scanning system, controlled by the synthesised optimal algorithm, in relation to a two degrees one, controlled by a conventional algorithm, for the survey mode of the environment sector when MRS moving are shown from the experiments conducted.

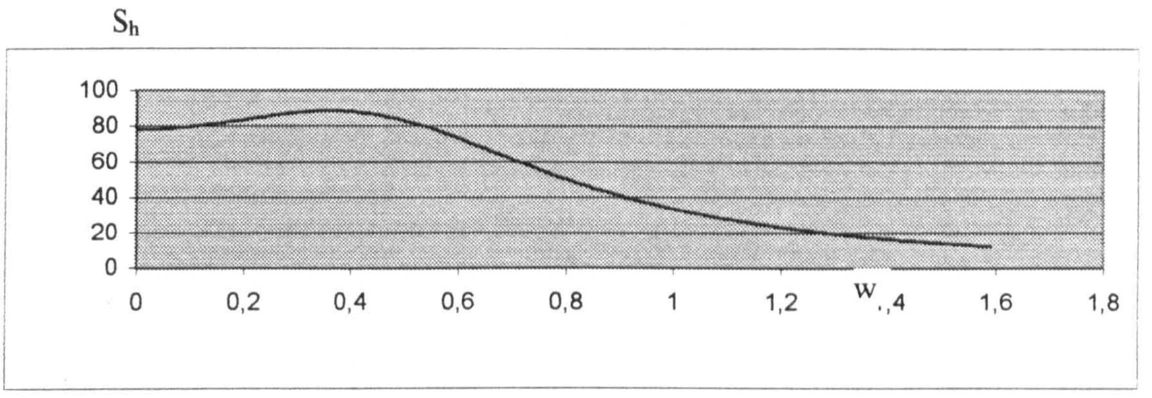


Figure 6.2 Spectral Density for Road of Longitudinal Profile

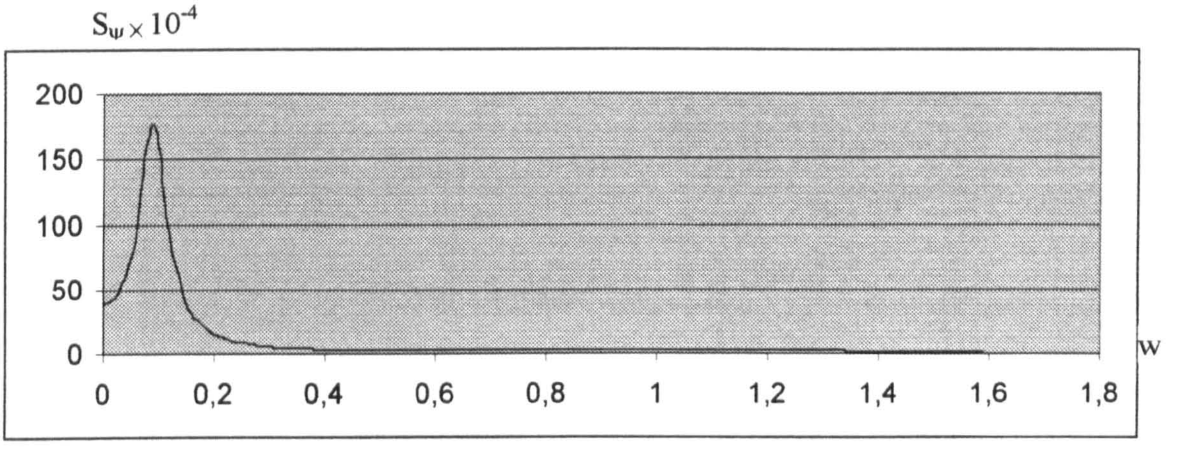


Figure 6.3 Spectral Density for Road of Transversal Profile

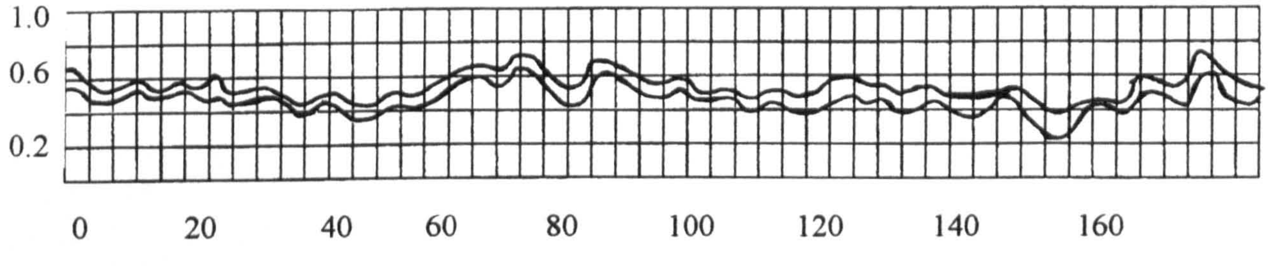


Figure 6.4 Longitudinal Profile of Road

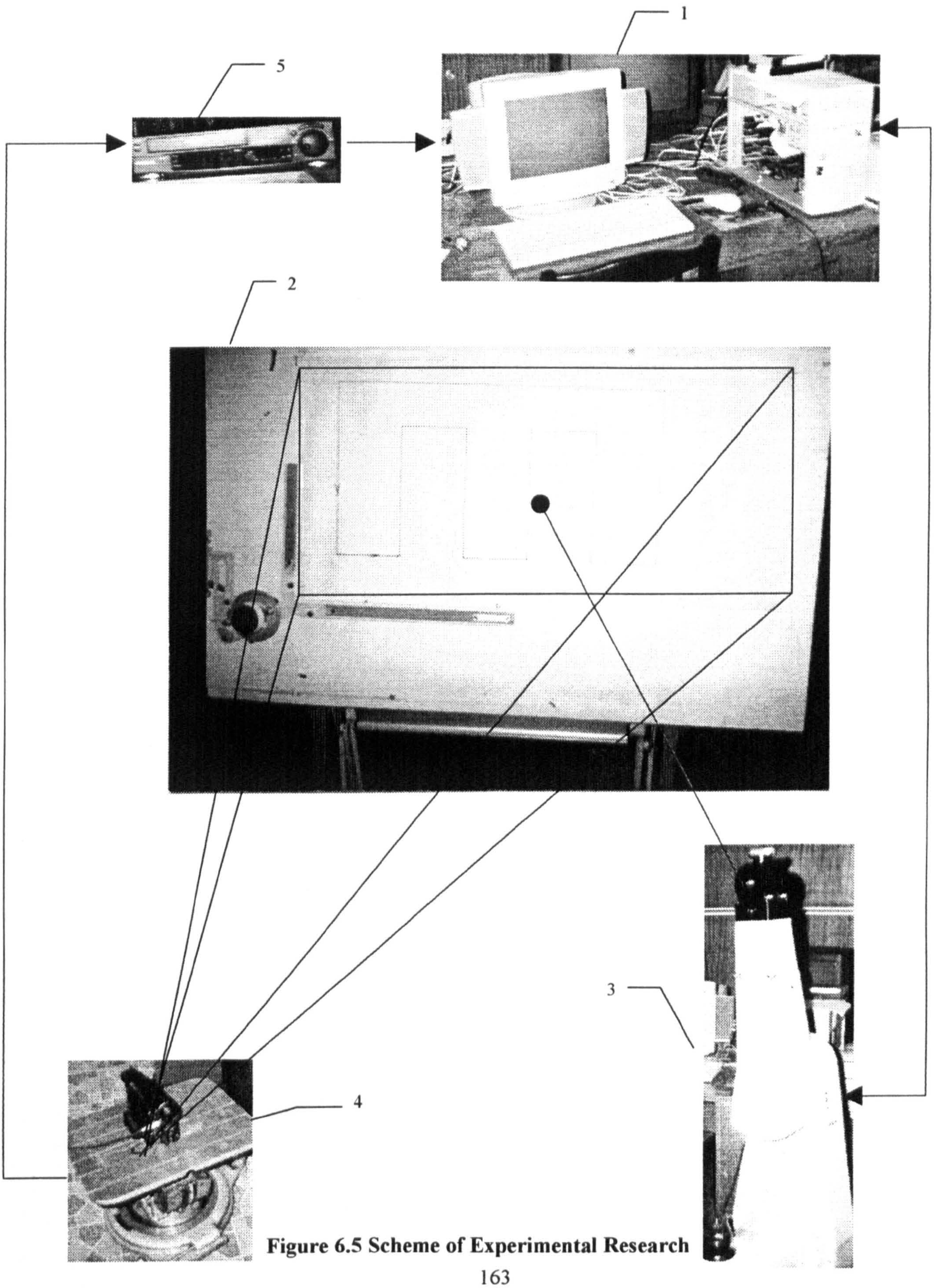


Figure 6.5 Scheme of Experimental Research

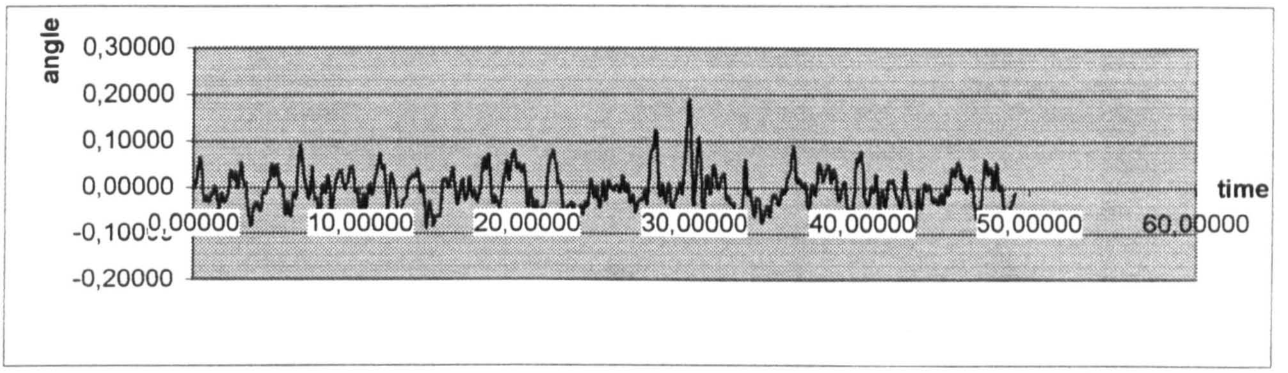


Figure 6.6 A Disturbance of Scanning system in Angles when Robot Velocity is Equal to 5 m/s

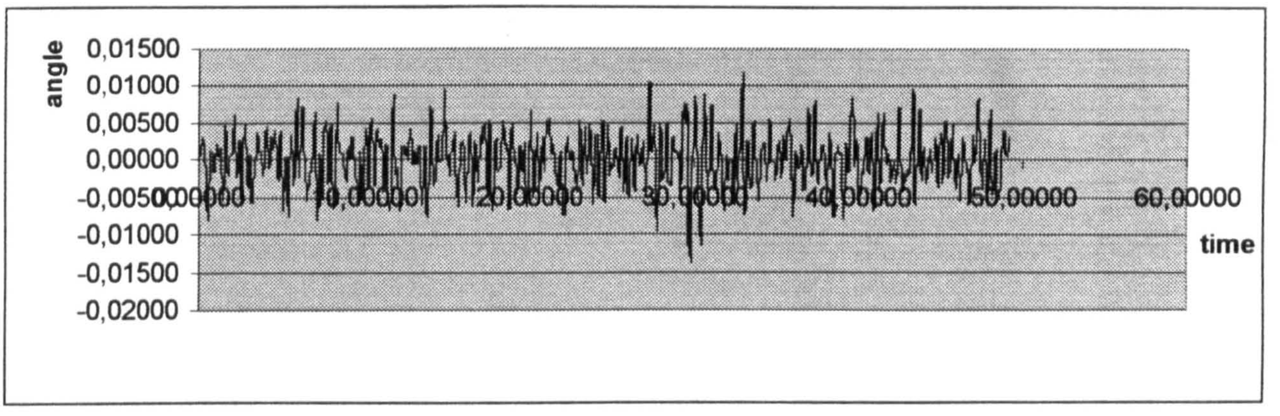


Figure 6.6 B Difference Between Given and Real Motion of Scanning system when MRC Robot is Equal to 5 m/s with Use Synthesised Control Algorithms for a Three Degrees Scanning system

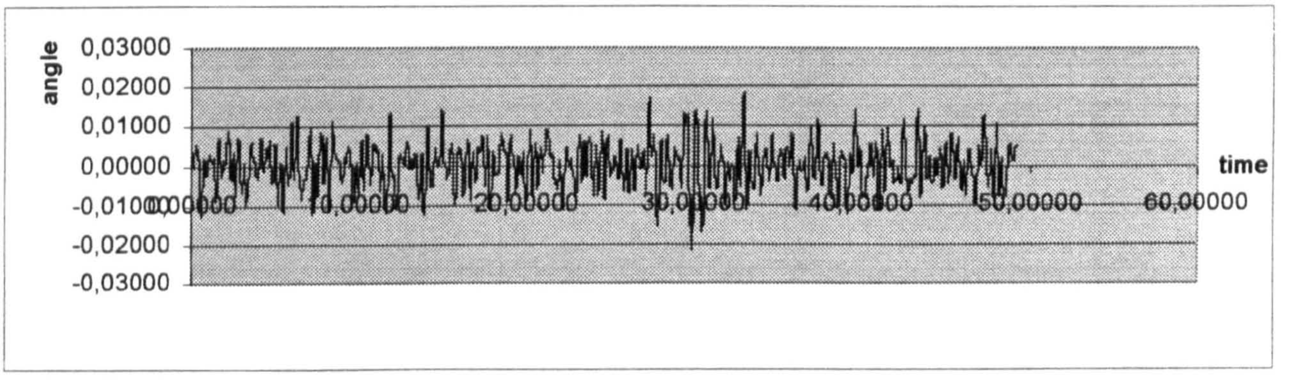


Figure 6.6 C Difference Between Given and Real Motion of Scanning system when MRC Robot is Equal to 5 m/s with Use Traditional Control Algorithms for a Two Degrees Scanning system

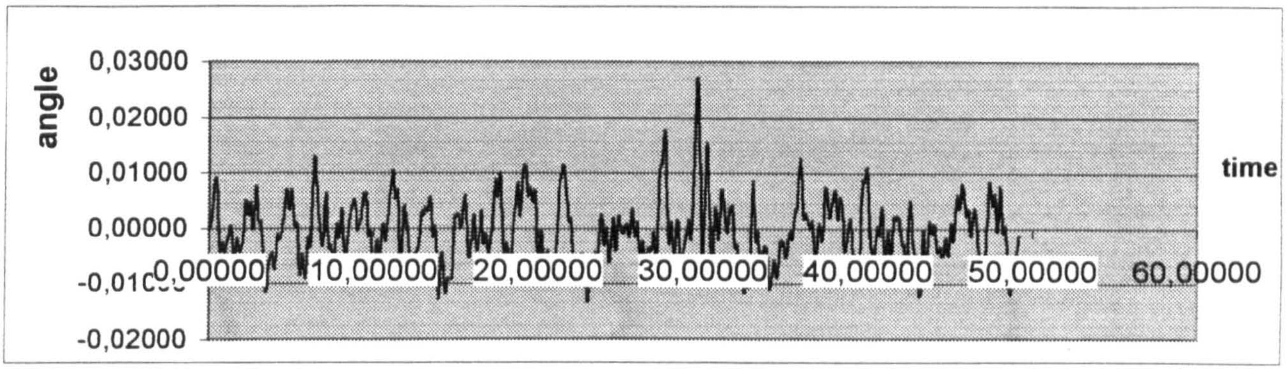


Figure 6.7 A Disturbance of Scanning system in Angles when Robot Velocity is Equal to 8 m/s

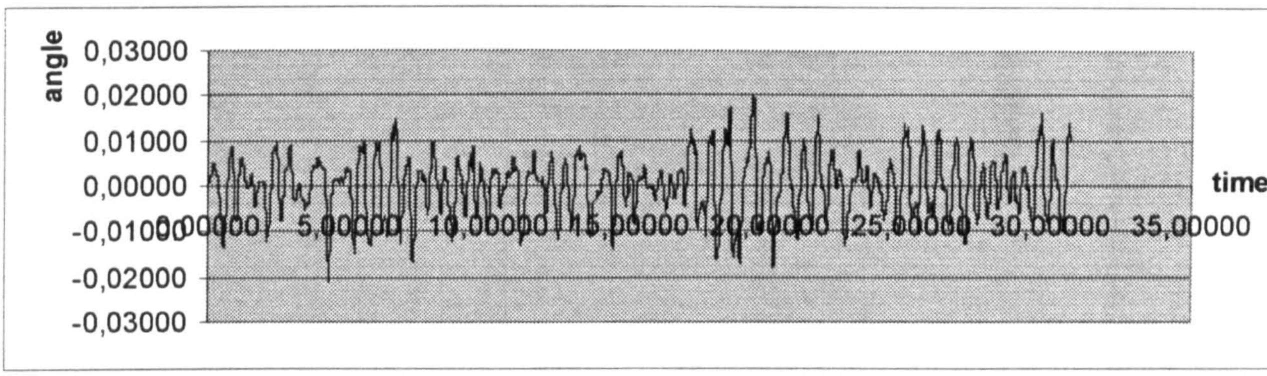


Figure 6.7 B Difference Between Given and Real Motion of Scanning system when MRC Robot is Equal to 8 m/s with Use Synthesised Control Algorithms for a Three Degrees Scanning system

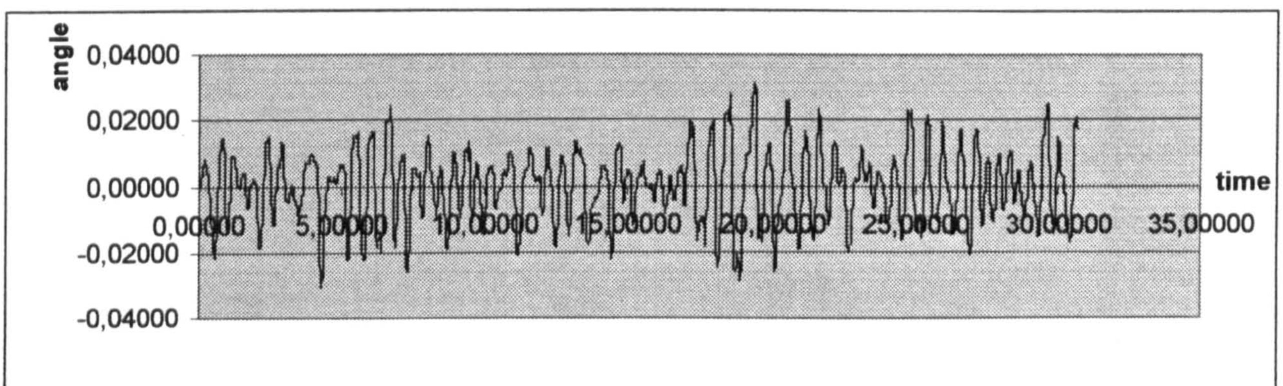


Figure 6.7 C Difference Between Given and Real Motion of Scanning system when MRC Robot is Equal to 8 m/s with Use Traditional Control Algorithms for a Two Degrees Scanning system

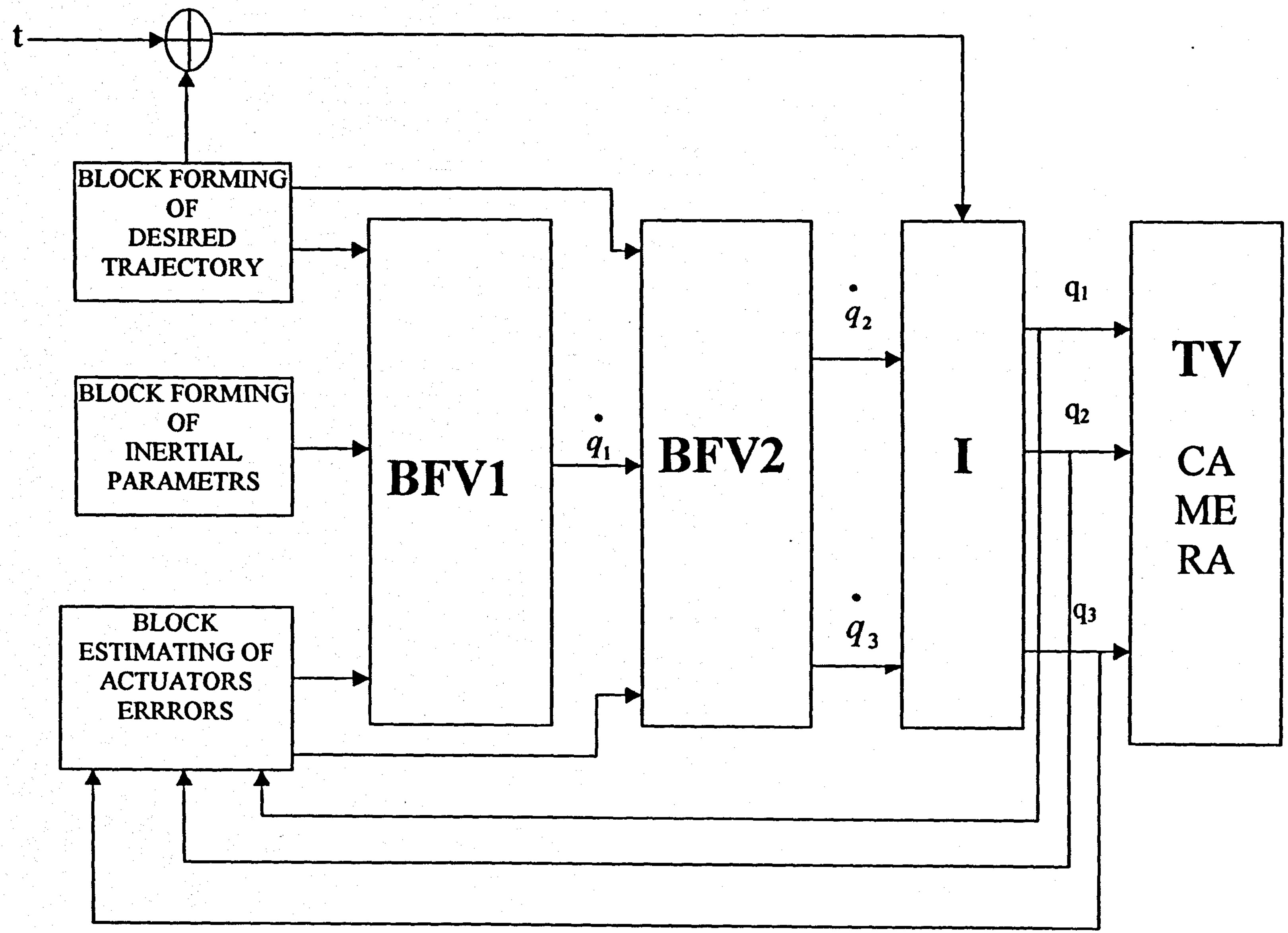


Figure 6.8 Scheme of Practical Implementation of the Control Algorithm for the Scanning System

CHAPTER 7

DISCUSSION

7. DISCUSSION

This thesis is devoted to problems of design of a mechanical scanning system and the synthesis of the fast scanning algorithms for a CCD vision system of a mobile robot system. The following results have been obtained in this work:

- The mathematical model of a transport subsystem for an MRS subject to an external disturbance has been created during this research.
- An effective configuration of the kinematic scheme for a fast mechanical scanning system for an MRS has been proved and verified.
- The fast scanning algorithms for an environment sector have been produced for different types of sensors;
- Optimal control algorithms for the developed scanning system are synthesised.

These results are discussed in this chapter.

Mathematical model of transport subsystem for MRS as a subject of external disturbance

The mathematical model of the transport subsystem for an MRS is created in this thesis. The correctness of the model and identification of regression coefficients have been shown on the basis of the experimental results. Accuracy of this model has been estimated. The approach offered relates to the initial data extracting for the power calculation stage of the scanning system actuators. This approach has been produced on the basis of the developed model of an MRS motion. Application of such an approach allows to avoid creating a general mathematical dynamic model of the MRS transport subsystem. It considerably reduces the design time of scanning system actuators in comparison with the application of traditional methods. An analysis of the MRS mathematical model is carried out in this thesis. The main peculiarities of the ground mobile systems are determined together with action on to the mechanisms of CCD vision system.

Choice and verification of the expedient configuration of kinematic scheme for fast mechanical scanning system of MRS

An expedient kinematic scheme for a fast mechanical scanning system has been selected. This choice is based on the analytical decision of the inverse kinematic task in terms of positions, velocities and accelerations for the mechanism with kinematic redundancy. This kinematic schema has three axes (see Figure 3.1).

This solution does not limit possible movement of the goal point. The received equations have been used for synthesis of the developed control algorithm, which is capable being realised in an onboard computer. Also, these equations can be used at the stage of analysis for traditional schemes of mechanisms with two axes. Further, it can be used modelling to avoid the necessity for digital integration thereby reducing calculation time and increasing the accuracy of results. Algorithm modification of the used methods has been carried out in this thesis. It allows to reduce requirements for an onboard computer's memory.

The analysis of kinematics for traditional schemes of scanning systems has been conducted using the developed methods. According to the carried out analysis a three axes scheme of mechanism is justified for application in a fast mechanical scanning system. Use of this scheme provides realisation of the view-finding systems with the following practical characteristics:

- high fast action;
- small power consumption;
- augmented work zone.

Moreover, implementation of this scheme and the synthesised control algorithms allow the avoidance of using gyro stabilising platforms (this traditional solution to stabilise the base of a scanning system). This solution permits the reduction of the mass-inertia and linear size characteristics of a scanning system. It is important for an autonomous MRS for which the problems of weight reduction and size of components is desirable.

Algorithms of fast scanning for the environment sector

The algorithm of an optimal review trajectory for the given environment sector is offered in this thesis. It provides a closed trajectory of survey without "loops", maximum value and equality of scanning frequencies for some subsectors. The offered algorithm permits the reduction in the calculation time and to increase the frequency of view of some subsectors by a factor of 2 in comparison with the application of traditional algorithms. The possibility of the practical application of the offered algorithm for MRS CCD vision system is demonstrated.

The algorithms for fast definition of an optimal view route for rectangular sectors of environment are offered. These algorithms essentially reduce the calculation time of an optimal view route for a given environment sector. It allows an extended area of application for these algorithms. These algorithms can be applied for determination of the optimal view route for the different sensors used with informational channels of an MRS.

Synthesis of an optimal control for the kinematically redundant scanning system

There are many approaches to resolve redundancy for space mechanisms. Such as synthesis of optimal control using the local torque minimisation method control algorithm which minimises jerks, the first time derivatives of the relative accelerations of the links and etc. Singularities of scanning systems of an MRS are the following:

- The degeneration areas of scanning system appear as a result of a power limitation for the conducting drives in joints.
- Conducting drives of larger power have high mass-inertia characteristics.
- It is necessary to take into consideration the autonomy of the MRS power supply (as a result it is limited) during synthesis of the optimal control for the scanning system conducting drives.

That is why minimum of instant sum of the scanning system drive power is chosen as criteria of quality. The developed optimal control algorithms minimise this criteria.

The approach for control with use of the criteria of minimum instantaneous power and the algorithm for optimal control synthesis for the MRS scanning system mechanism have been created. The developed algorithm permits formulation of control that ensures "dead zones" of closed types.

Recommendations for the practical realisation for the developed algorithm in the MRS onboard computer have been offered. Three different schemes for the practical realisation of the developed control algorithms are offered.

The accuracy of this control algorithm and the scheme for practical realisation have been shown by experimental research. The advantages of this control algorithm for a three degrees scanning system have been shown by experimental research. The developed control algorithm for the three degrees scanning system provides high accuracy relative to an equivalent two degrees scanning system is more than 163%.

CHAPTER 8

CONCLUSION

8. CONCLUSION

This thesis is devoted to problems of the mechanical scanning system design and the synthesis of the fast scanning algorithms for the different sensors for a mobile robotic system. The following results are presented.

The mathematical model of transport subsystem for MRS subjected to an external disturbance has been created during this research. The reliability of this model has been proved by the experimental investigation.

The approach of initial data forming for the power calculation stage has been offered. Analysis of this approach shows, in general, that it is support to the general mathematical model of MRS transport subsystem. It reduces design time considerably.

The main singularities of external disturbances for mobile robotic systems have been determined. An estimation of the different factors for external disturbance allows analysing number of technical solutions for the realisation in MRS sensors.

Practical recommendations for the implementation of the different kinematic schemes for scanning systems have been given. An effective kinematic scheme of a fast mechanical scanning system for an MRS has been produced and verified.

Solutions for the inverse kinematics in terms of velocities and accelerations for the 3 degree of freedom scanning system are presented in this thesis.

The algorithms for fast scanning of the environment sector have been produced for different types of sensors.

Optimal control algorithms for the developed scanning system are produced. This control minimises the instant sum of actuator power of the scanning system. The control algorithm has been verified by experiment.

CHAPTER 9

FUTURE WORK

9. FUTURE WORK

Some problems connected with the creation of the perspective scanning systems for an MRS have not been considered in this thesis. Research of the external disturbance when an MRS is moving shows requirements to actuators of scanning systems are high with respect to part dynamics. It seems interesting to research the problem of the implementation perspective piezoelectric actuators for MRS scanning system. These actuators have high dynamic characteristics, small weights and small sizes (*Bouer A., Moller F* [1994], *Andersen B., Millar C. E.* [1994], *Hoertling G.* [1994]). They do not require using of any reducers and systems of damping (*Erofeev A.* [1994], *Erofeev* and *et al* [1994]).

Another problem is studied in this thesis partly is the synthesis of the scanning algorithms for the known environment sector with use digital map. Now, a digital map of environment is used to solve the navigation problem of MRS often. In this case, a scanning algorithm must take into consideration probability characteristics of target appearance into the visible zone of MRS.

REFERENCES

Andersen B., Millar C. E. Performance of Multiplayers Actuators Based on Piezoelectric and Electrostrictive Materials. Proceedings of Actuators '94 New Orleans 1994. pp. 167-170.

Artobolevsky I. (1988) Theory of Machines and Mechanisms. Moscow. Nauka. ISBN 5-88347-002-5.

Astapov. U., Medvedev V. (1982) Statistical Theory of Automatical Regulationa and Control Systems . Moscow. Nauka. ISBN 5-217-02661-8

Baillieul J. Kinematic Programming Alternatives for Redundant Manipulators. Proc of IEEE Int. Conf. on Robotics and Automation New Orleans. 1985 pp722-728.

Bairashevsky A. and Nichiporenko N. (1982) Leningrad. Ship Radiolocation Systems. Transport. ISBN 87343-053-5.

Baker D. R., Wampler II C. W. (1988) On the Inverse Kinematics of Robot Manipulators with Redundant Manipulators. Int. Journal Robotic Research. Vol 7, pp. 2-21.

Belyansky P. and Sergeev B. (1980) Control of Ground Antennas and Radiotelescopes. Moscow. Sovetskoe Radio.

Bessekersky A. and Fabrikant E. (1968) Theory of Automatic Control. Moscow. Nauka.

Bouer A., Moller F Piezo Actuator Special Design. Proceedings of Actuators '94.Chicago 1994. pp.128-132.

Bronshtein I. and Semendyaev K. (1988) Handbook of Mathematics for Engineers and Students. Moscow. Nauka.. ISBN 5-217-02659-6

Chang P. H. A Closed Form Solution for Tyhe Control of Manipulators with Kinematic Redundancy. Proc of IEEE Int. Conf. on Robotics and Automation San Francisco, CA, 1986 pp. 9-14.

Chelyshev V. (1995) Development of Conception for Basis Technologyand Experimental Patterns of Multipurpose Mobile Robot. Moscow. BMSTU. ISBN 5-317-05745-6

Cho D. K., Choi B. W., Chung M. J. (1995) Optimal Conditions for Inverse Kinematics of a Robot Manipulator with Redundancy. Robotica. Vol 13, pp. 95-101.

Chuyan R. (1988) Methods of Mathematical Modelling for Drivers of Flying Apparats. Moscow. Mashinostroenie. ISBN 5-418-45623-2

Denavit J., Hartenberg R. S. (1955) A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices. J. Appl. Mech. Vol 77, pp. 215-221.

Dermatas E., Nearchou A., Asparathos N. (1996) Error-back-propagation Solution to the Inverse Kinematic Problem of Redundant Manipulators. Robotics & Computer-Integrated Manufacturing. Vol 12, N_0 12, pp. 303-310.

Dimentberg F. (1978) Theory of screws and They Application. Moscow. Nauka..

Eremin I., Astafiev V. (1976) Introduction to the Theory of Linear and Convex Programming. Moscow. Nauka..

Erofeev A. (1994) Piezoelectic Motors. Moscow. Nauka. ISBN 5-318-45612-5

Erofeev A., Proklin A., Ulanov V. (1994) Piezoelectonics. Moscow. Radio I Svyaz. ISBN 5-345-02715-6

Frolov K.(1988) Mechanics of industrial robots. Vol 1. Moscow. Vyshaya Shkola. ISBN 5-989-20775-5

Fu K. S., Gonzalez R. C., Lee C. S. G. (1989) Robotics: Control, Sensing, Vision and Intelligence. Moscow. Mir. ISBN 5-65611-5

Ganshin G. (1987) Optimal Control. Moscow. Nauka. ISBN 5-437-03644-8

Gantmacher F. (1988) Theory of Matrixes. Moscow. Nauka. ISBN 5-634-23314-8

Goldenberg A., Benhabib B., Fenton R. (1985) A Complete Generalized Solution to the Inverse Kinematics of Robots. IEEE J. Robotics Automn. RA-1(1), pp. 14-20.

Goldshtein E. and Udin D. (1966) Linear Programming. Moscow. Nauka.

Gotlin K., Troch I., Jezernik K. (1995) Global Optimal Control of Redundant Robot. Robotica. Vol 14, pp. 131-140.

Guez A., Ahmad Z. Solution to the Inverse Kinematics Problem in Robotics by Neural Networks. IEEE Int. Conf. on Neural Networks. San-Diego, CA Vol II, 1988 pp. 617-621.

Guo J., Cherkassky V. (1989) A Solution to the Inverse Kinematic Problem in Robotics Using Neural Network Processing. IEEE International Conf. on Neural Networks. Washington, DC, Vol II, pp. 299-304.

Hanevsky G. (1951) Mechanisms of Antennas Revolution. Moscow. Sovetskoe Radio.

Hoertling G. (1994) Ultra High Displacement Actuator. American Society Bulletin. Vol.73, pp.93-96.

Hollerbach J. M., Suh K. S. Local Versus Global Torque Optimization of Redundant Manipulators. Proc. of IEEE Conf on Robotics and Automation Ohio State University, Columbus,.1987 pp. 619-624.

Hou E. S., Utama W. An Artificial Neural Networks for Redundant Manipulator Inverse Kinematics Computation. Proc. of the SPIE - The International Society for Optical Engineering. Chicago. 1992 pp. 668-677.

Ishlinskiy A. (1963) . Mechanics of Gyroscopic System. Moscow. Nauka..

Ishlinskiy A. (1976) . Orientation of Gyroscop and Inertional Navigation. Moscow. Nauka.

Ivanov V. and Faldin N. (1981) Optimal control. Moscow. Nauka.

Kazerounian K., Nedungadi A. Redundancy resolution of robotic manipulators at the acceleration level. Proc. 7th World congress of IFToMM, Sevilia, Spain, 1987 pp. 1207-1211.

Kazerounian K., Wang Z. (1988) Global versus local optimization in redundancy resolution of robotic manipulators. Int. J. Robotics Research. Vol7(5), pp. 3-12.

Kieffer S., Morellas V., Donath M. Neural network learning of the inverse kinematic relationships for a robot arm. IEEE International Con. on Robotics and Automation. Sacramento, CA, 1991 pp. 2418-2425.

Kirsanov U. (1997) English Anti-aircraft Rocket Complex "Rapira-2000". Moscow. Zarubezhnoe Voennoe Obozrenie. Vol. 2 pp. 27-29.

Klimov D. and Rudenko V. Research of Complex Spatial Mechanisms with Use of Analytical Expessions on Computer. Processing of International Congress for Analytical Calculations with Use of Computer and It's Application in Theoretical Physics. Dubna. 1985 pp. 276-288.

Kobayashi A. (1996) Abstract of Report of Disaster Robot System for Forest Fire. Robot. Vol. 50, pp.58-64.

Kochergin V. (1988) The Following System with Direct Current Motor. Moscow. Elektroatomizdat. ISBN 5-534-06743-6

Kudrevich B. (1963) . The Theory of Gyroscopic Devices. Leningrad. Sudpromgiz.

Kudryavcev V. (1991) Mathematical Analysis. Moscow. Vyshaya Shkola. ISBN 5-428-87695-5

Kulakov F. (1980) Supervision Control of Robot Manipulators. Moscow. Nauka..

Kyriakopoulos K. J., Saridis G. N. Minimum Jerk Path Generation. IEEE Int. Conference of Robotics and Automation. Barcelona 1988 pp. 364-369.

Lebedev P. (1982) Vectorial Equations of Interdependency for Kinematic Parameters of Spatial Mechanisms. Moscow. Mashinovedenie. Vol. 4, pp. 54-58.

Lee S., Kil R. M. Robot Kinematic Control Based on Bi-directional Mapping Neural Networks. Int Joint Conf. on Neural Networks. Chicago 1990 pp. 327-335.

Liegeois A. (1977) Automatic Supervisory Control of the Configuration and Behaviour of Multibody Mechanisms. IEEE Trans. Systems. Man. Cybern. Vol 7, pp. 868-871,

Lin Z., Khorasani K. , Patel R. V. (1990) A Counter Propagation Neural Network for Function Approximation. IEEE International Con. on Systems. Man and Cybernetics. SMC-133, pp. 328-384.

Lyashenko I. (1975) Methods of Linear Programming. Moscow. Nauka

Ma S., Nenchev D.N. (1996) Local Torque Minimization for Redundant Manipulators: a Correct Formulation. Robotica. Vol 14 pp. 235-239.

Mao Z., Hsia T. C. (1997) Obstacle Avoidance Inverse Kinematics Solution of Redundant Robots by Neural Networks. Robotica, Vol 15, pp. 3-10.

Marchuk G. (1980) Methods of Calculations Mathematics. Moscow. Nauka.

Martin D.P, Baillieul J., Hollerbach (1989) Resolution of Kinematic Redundancy Using Optimization Techniques. IEEE Trans. on Robotics and automation Vol 5, No 4, pp. 529-533.

Moroshkin U. (1952). Determination of Mechanism Configuration. Report A. N. USSR No. 4. Vol. 4 pp. 533-536.

Moiseev N and **Ivanovsky J.** (1978) Methods of optimisation. Moscow. Nauka.

Nacamura Y., Hanafusa H. (1987) Optimal Redundancy Control of Robot Manipulator. Int. J. Robotic Research. Vol 6, No 1, pp. 32-42.

Nesterov V. Automatized output of the analytical expressions of inverse kinematic tasks for complex spatial mechanisms. Proc. Conf. Actual Problems of Fundamental Sciences. B. II, Moscow 1991 pp. 147-149.

Nesterov V. (1994) Methods of Optimal Control Synthesis for Ship Sighting Systems Mechanisms at Spherical -Type Dead Band. Moscow State Technical Univesity.

Oh S.Y., Orin D., Bach M. (1984) An Inverse Kinematic Solution for Kinematically Redundant Robot Manipulators. J. Robotic Research. Vol 1, pp 235-249.

Ovakimov A. (1971/a) The Analytical Method for Determination of Speeds and Accelerations for Spatial Mechnisms with Some Degrees of Freedom. Mechanika Machin. Vol 35-36, pp.45-62.

Ovakimov A. (1971/b) The Kinematic Research of Spatial Chain for Controlling Mechanism of Manipulator. Izvestiya Vuzov. Mashinostroenie. Vol 4, pp.58-62.

Pantushin S., Nazaretov V., Tyagunov O. (1986) Modelling of Robotic Systems and Flexible Automated Manufactures. Vishaya Shkola. ISBN 5-677-13983-5

Paul R.P. (1972) Modelling, Trajectory Calculation and Servoing of a Controlling Computer Arm. Memo AIM-177 Stanford Artificial Intelligence Laboratory, Palo Alto.

Paul R.P. (1979) Manipulator Cartesian Path Control. IEEE Trans. Systems, Man, Cubern, SMC-9, Vol. 11, pp. 702-711.

Peisach E. (1981). Problems of Kinematics for Robots - Manipulators and Program System "Robot". Roboty and Robototechnicheskie Systemy pp. 22-40. ISBN 5-677-65768-6

Pelpor S. (1989) Gyroscopic Devices. Moscow, Vyshaya Shkola. ISBN 5-221-43535-6

Platonov A., Stepanov U., Trubitsin O. and **Kirilchenko A.** (1996) Perspective of Development and Analysis of Singularities of Control Systems for Mobile Robots. Preprint of Institute Prikladnyh Matematicheskih Nauk. Vol.3 pp1-32.

Podobry G., Faldin K. and Kirsanov I. (1969) Calculation of Following system. Moscow, Vyshaya Shkola.

Popov E. (1978) Theory of Linear Systems of Automatic Regulation and Control. Moscow, Nauka..

Popov E. (1979) Theory of Nonlinear Systems of Automatic Regulation and Control. *Nauka.*

Popov E. (1982) Following Systems of Antena's device.Moscow State Technical Univesity. ISBN 5-463-77399-5

Popov E., Vereschagin A. and Zenkevich S. (1978) Robot Manipulators: Dynamics and Algorithms. Moscow, Nauka..

Rivkin S. (1978) . Stabilization of Sensoring Devices on Rocking Base. Moscow, Nauka..

Rogozinnikov A. (1996) Spring with Vibration Smoothing of Small Telecontrolled Track Vehicle - Mobile Robot. Abstract of Dissertation Thesis, Moscow State Technical Univesity pp. 9-10.

Romanovsky P. (1980) Fourier Series. Theory of Field. Analytical and Special Functions. Laplas Tranformation. Moscow, Nauka.,

Sciavicco L., Sicilliano B. (1988) A Solution Algorithm to the Inverse Kinematic Problem of Redundant Manipulators. Proc. of IEEE Trans. on Robotics and Automation. Vol 4, pp. 403-410.

Smirnov G. (1990) Theory of Motion for Wheel Machine. Moscow, Mashinostroenie, ISBN 5-746-43526-5

Stepanov U., Trubitsin O., Kirilchenko A., Platonov A. and **Pryanichkov V.** (1996) Expert Support of Control Systems for Mobile Robots. Preprint of Institute Prikladnyh Matematicheskih Nauk. Vol.3 pp 32-57.

Storozhenko V. and Temchenko M. (1971) About Application of Theory of Finite Rotations to task of autonomous definition of position coordinates for moving objects. Izvestiya AN USSR, Mechnika Tverdogo Tela. Vol 3, pp. 3-10.

Timoshenko S. (1959) Oscillations in Engineering. Moscow, Phisico-Matematicheskaya Literatura.

Udin D. and **Goldshtein E.** (1969) Linear and Dynamic Programming. Moscow, Nauka.

Umertani Y., Hirano S. (1991) Summary Report on Concepptual Research of Advanced Robot Systems. Robot. Vol. 82 pp. 60-66.

Urevich E. (1984) Dynamics of Robot Control. Moscow, Nauka. ISBN 5-435-84536-6

Vendik O. (1965) Antennas with Nonmechanical Movement of Beam. Moscow, Sovetskoe Radio

Volochatuk V., Kochetkov V. and Krasovsky R. (1971) Questions of Optical Location. Moscow, Sovetskoe Radio.

Vorobiev E., Dimentberg F. (1991) .Space Joint Mechanisms. Open and Close Kinematic Chains. Moscow, Nauka., ISBN 5-563-45678-5

Whitney D.E. (1969) Resolved Motion Rate Control of Manipulator and Human Prosthesis. IEEE Trans. on man-Machine Systems. Vol 10, pp 47-53.

Won J. H., Choi B. W., Chung M. J. (1993) A Unified Approach to the Inverse Kinematic Solution for a Redundant Manipulator. Robotica, Vol 11, pp 159-165.

Wong J.Y. (1982). Theory of Ground Vehicles. Mashinostroenie, ISBN 5-746-76483-5

Wu C. (1994) A Numerical Approach for Time Optimal Path Planning of Kinematically Redundant Manipulators. Robotica, Vol 12, pp 401-410.

Zeldovich S. and Okon I. (1974) Application of Methods, Based on Theory of Finite Rotations, to Analysis of Geometry for Hyroscopic Devices. Izvestiya Vuzov, Priborostroenie. Vol 4, pp. 80-86.

Zufrin A. (1970) Methods of Building The Ship Automatic Angle- Measuring Systems. Leningrad, Sudostroenie.

Zuhovickiy S. and Avdeeva L. (1964) Linear Programming. Moscow, Nauka.

APPENDIX A

CHARACTERISTICS OF IRREGULARITIES ON A ROAD SURFACE

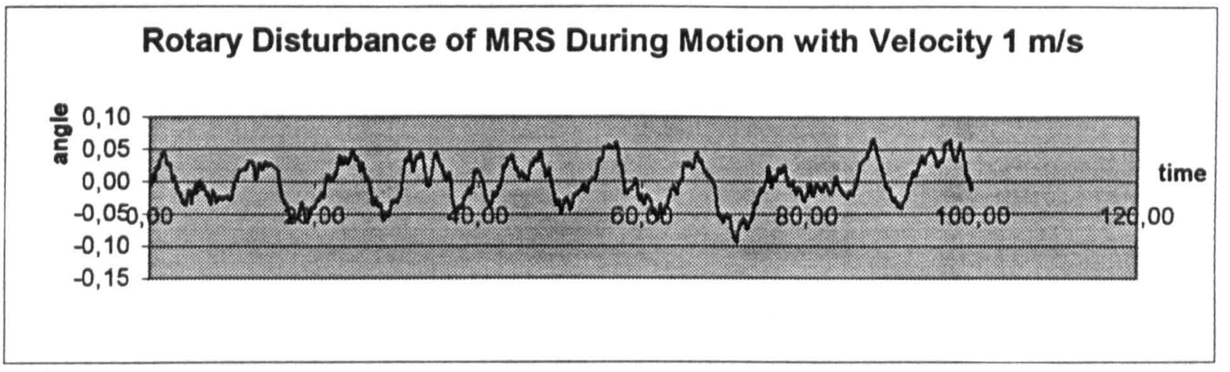
CHARACTERISTICS OF IRREGULARITIES ON A ROAD SURFACE

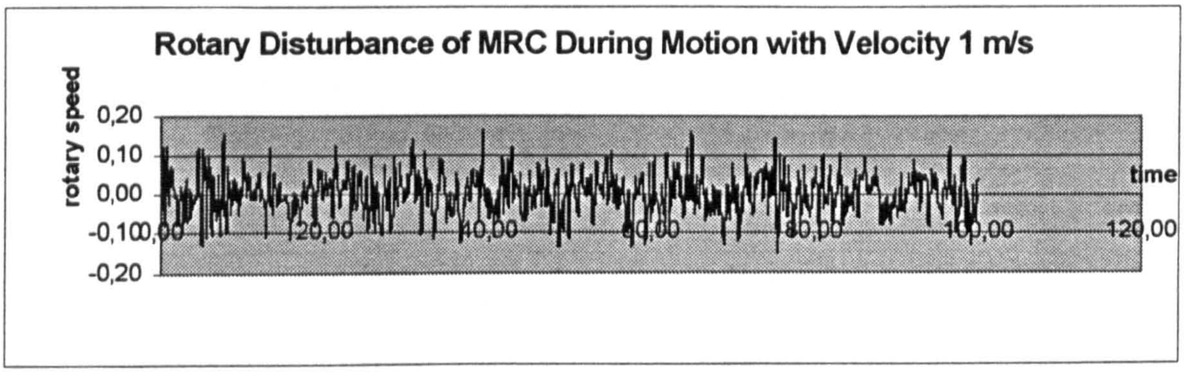
index of irregularities	low worn out road	high worn out road	damaged road	without road place
Short impulse with length up to 3 sm.				
number on 1 km.	20-50	50-100	100-200	200
height, sm.				
maximum	5	5-7	7-10	10-15
probably	3-4	3-5	5-7	7-10
ruts				
number on 1 km.	200	200-500	300-500	200-300
probably length, sm.	50-150	100-250	150-300	150-500
depth, sm.				
maximum	10	10-20	20-30	30
probably	3-5	5-10	10-15	15
pot-holes				
number on 1 km.	5	5-10	10-20	20
probably length, m	6-9	6-10	6-12	8-16
depth, sm.				
maximum	10	30	100	200
probably	3-5	10-20	30-50	70-120

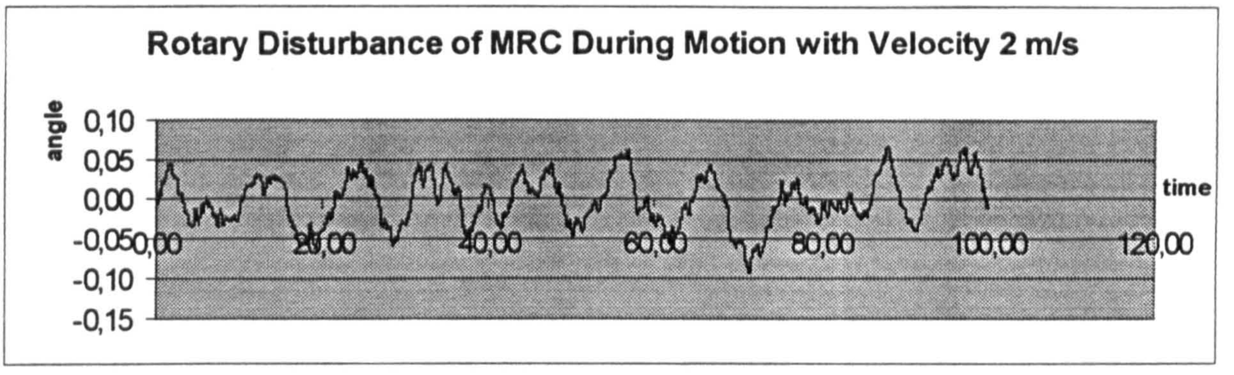
APPENDIX B

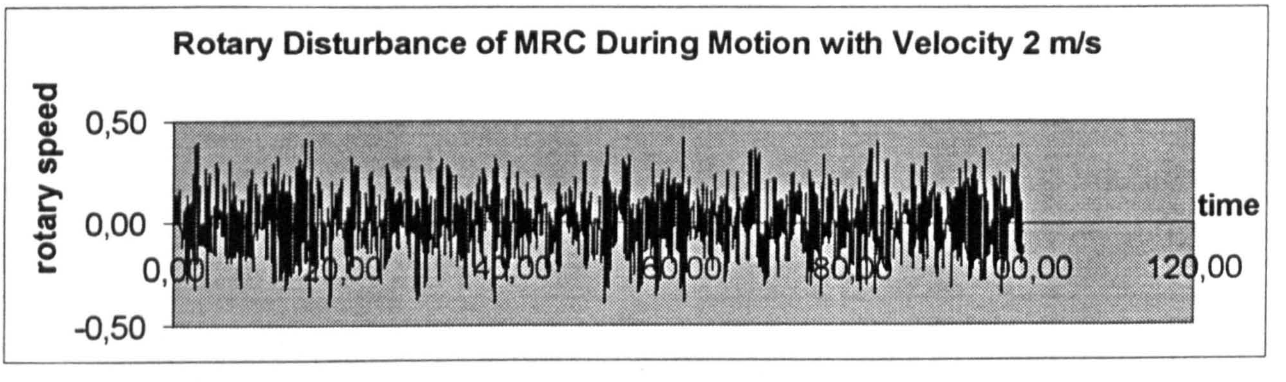
EXPERIMENTAL RESULTS OF EXTERNAL DISTURBANCES FOR MOBILE ROBOTIC SYSTEM

EXPERIMENTAL RESULTS OF EXTERNAL DISTURBANCES

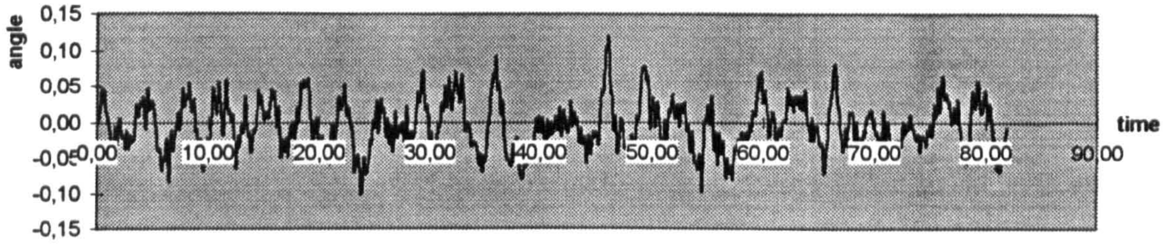




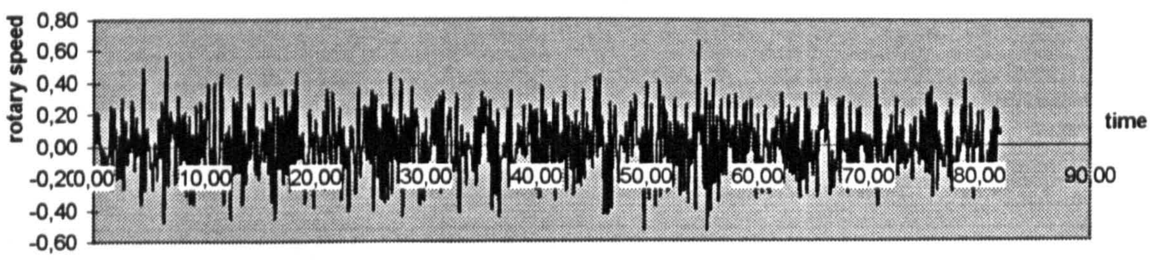

Rotary Disturbance of MRC During Motion with Velocity 2 m/s
angle
0,10
0,05
0,00
-0,05
-0,10
-0,15
0,00
20,00
40,00
60,00
80,00
100,00
120,00
time

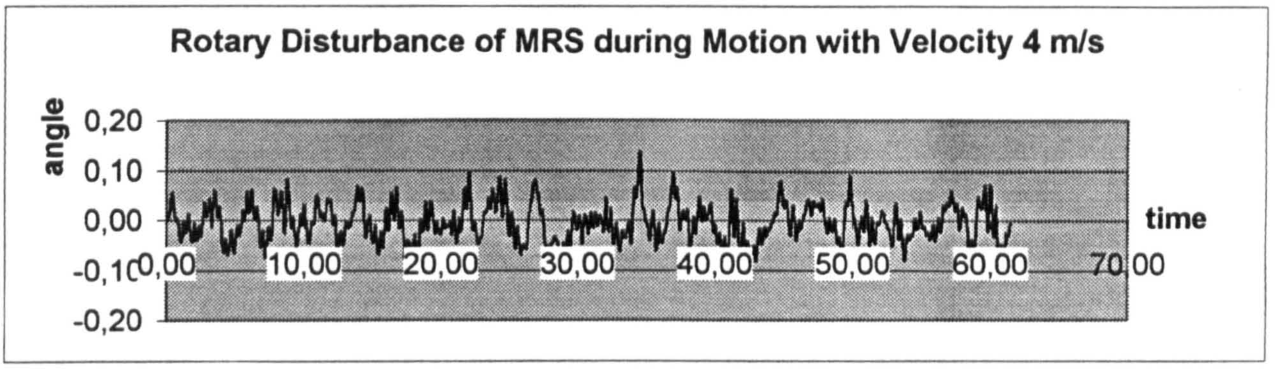

Rotary Disturbance of MRC During Motion with Velocity 2 m/s
rotary speed
0,50
0,00
-0,50
0,00
20,00
40,00
60,00
80,00
100,00
120,00
time

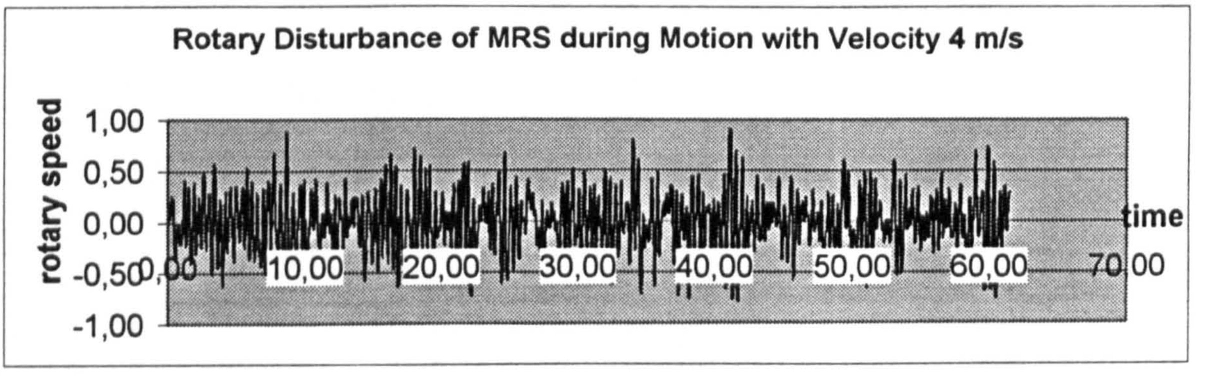
Rotary Disturbance of MRS Motion with Velocity 3 m/s

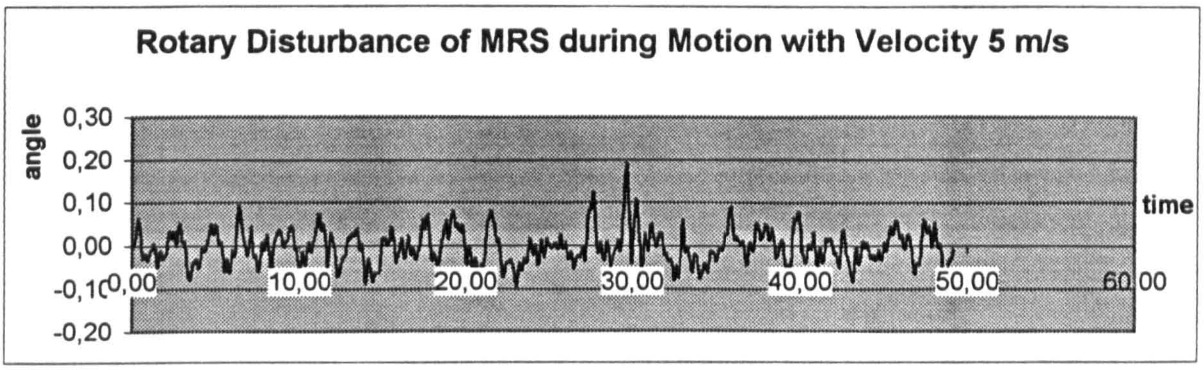


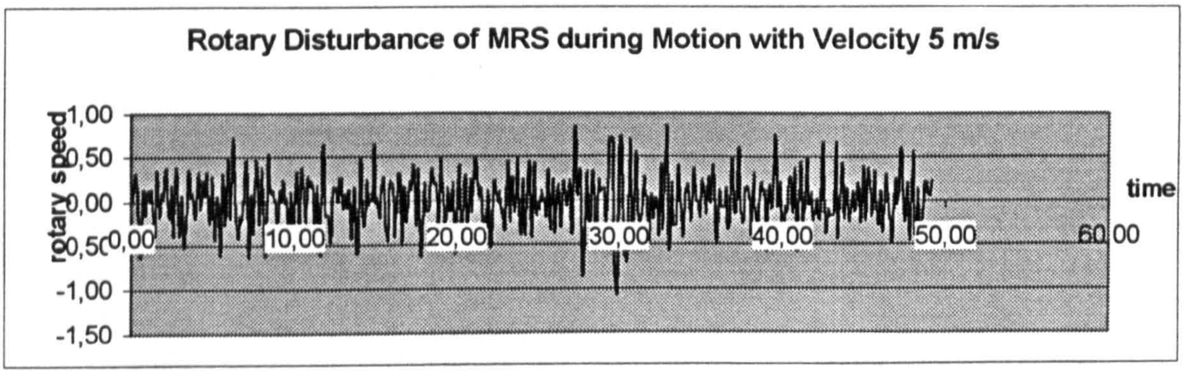
Rotary Disturbance of MRS Motion with Velocity 3 m/s

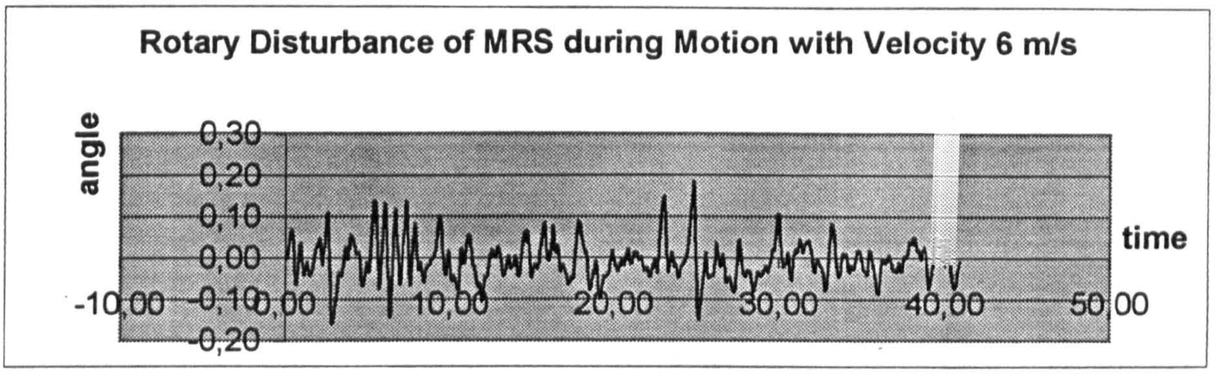



Rotary Disturbance of MRS during Motion with Velocity 4 m/s
angle
0,20
0,10
0,00
-0,10
-0,20
0,00
10,00
20,00
30,00
40,00
50,00
60,00
70,00
time

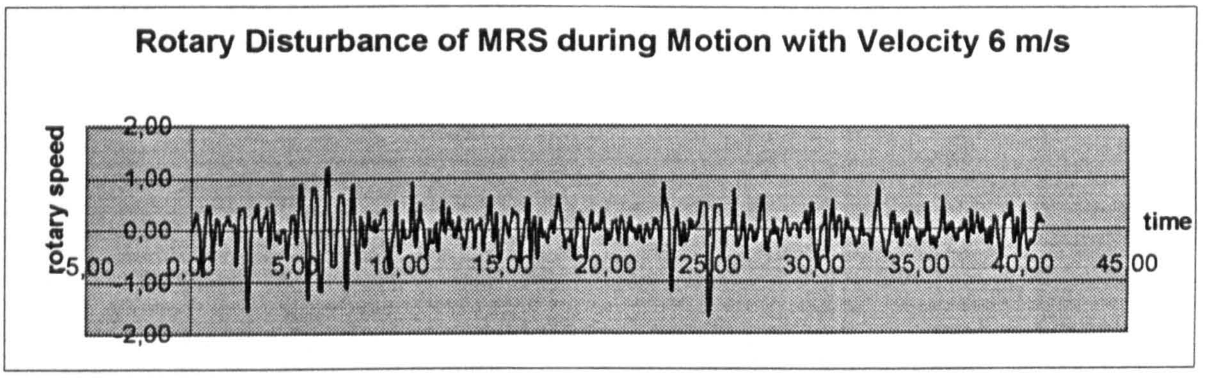

Rotary Disturbance of MRS during Motion with Velocity 4 m/s
rotary speed
1,00
0,50
0,00
-0,50
-1,00
0,00
10,00
20,00
30,00
40,00
50,00
60,00
70,00
time


Rotary Disturbance of MRS during Motion with Velocity 5 m/s
angle
0,30
0,20
0,10
0,00
-0,10
-0,20
time
0,00
10,00
20,00
30,00
40,00
50,00
60,00

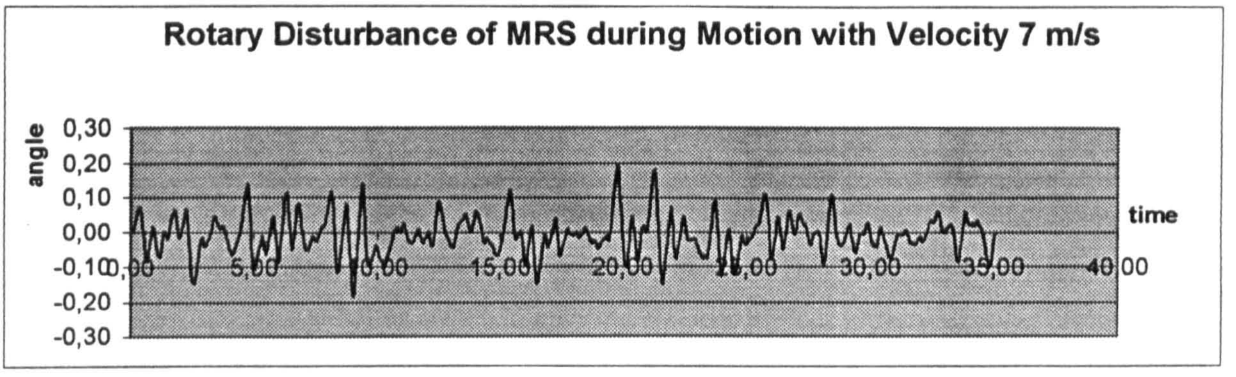

Rotary Disturbance of MRS during Motion with Velocity 5 m/s
rotary speed
1,00
0,50
0,00
-0,50
-1,00
-1,50
time
0,00
10,00
20,00
30,00
40,00
50,00
60,00

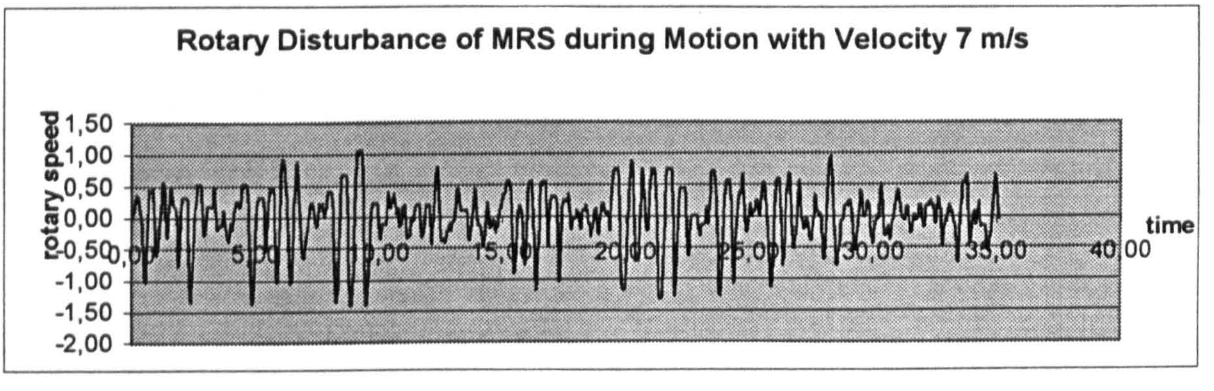


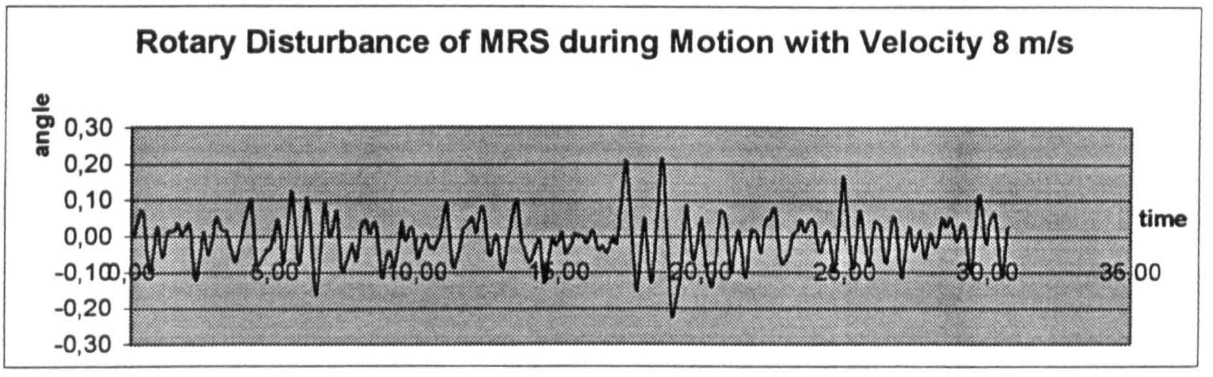
Rotary Disturbance of MRS during Motion with Velocity 6 m/s
angle
0,30
0,20
0,10
0,00
-0,10
-0,20
-10,00
0,00
10,00
20,00
30,00
40,00
50,00
time

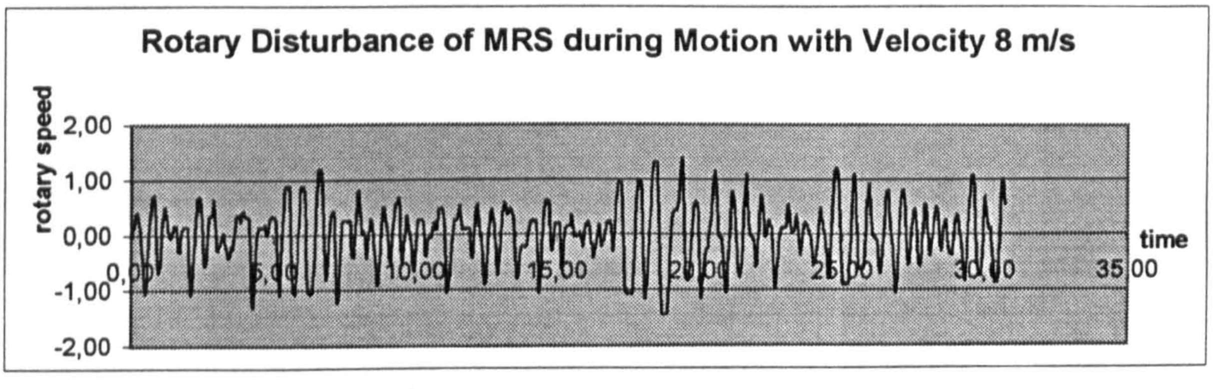


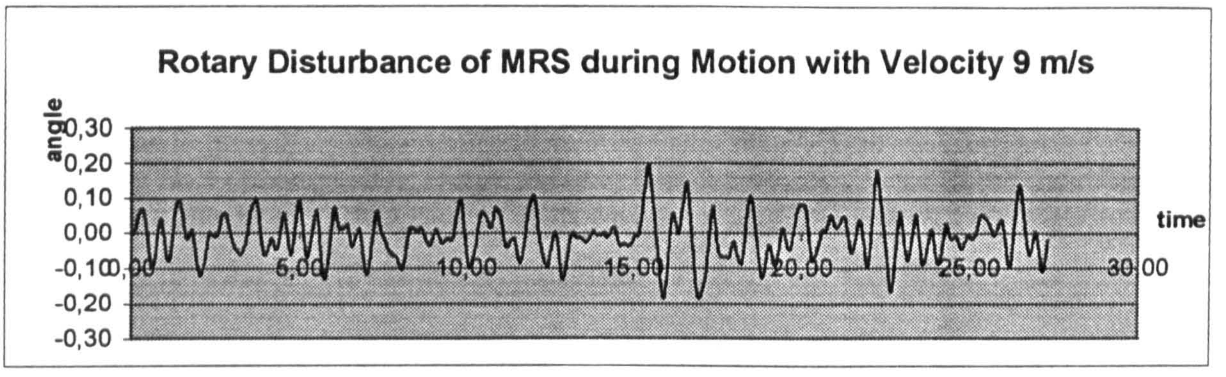
Rotary Disturbance of MRS during Motion with Velocity 6 m/s
rotary speed
2,00
1,00
0,00
-1,00
-2,00
-5,00
0,00
5,00
10,00
15,00
20,00
25,00
30,00
35,00
40,00
45,00
time

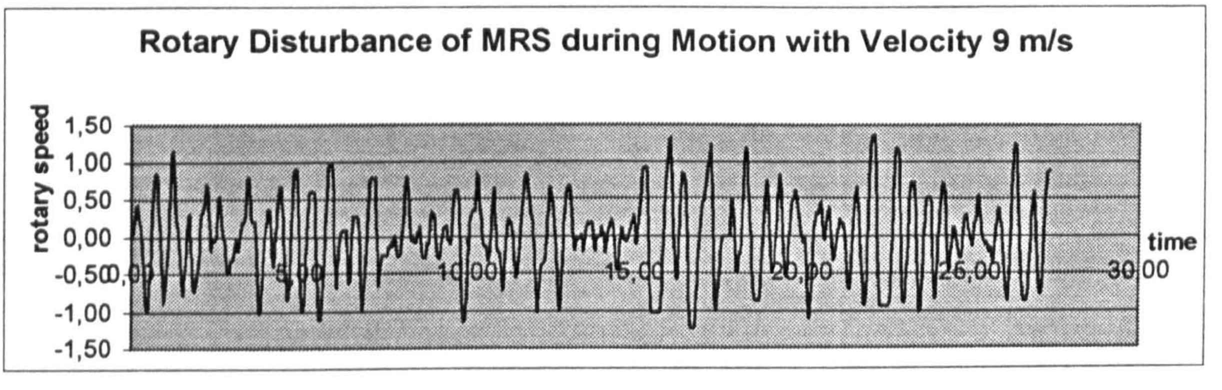

Rotary Disturbance of MRS during Motion with Velocity 7 m/s
angle
0,30
0,20
0,10
0,00
-0,10
-0,20
-0,30
0,00
5,00
10,00
15,00
20,00
25,00
30,00
35,00
40,00
time


Rotary Disturbance of MRS during Motion with Velocity 7 m/s
rotary speed
1,50
1,00
0,50
0,00
-0,50
-1,00
-1,50
-2,00
0,00
5,00
10,00
15,00
20,00
25,00
30,00
35,00
40,00
time


Rotary Disturbance of MRS during Motion with Velocity 8 m/s
angle
0,30
0,20
0,10
0,00
-0,10
-0,20
-0,30
0,00
5,00
10,00
15,00
20,00
25,00
30,00
35,00
time


Rotary Disturbance of MRS during Motion with Velocity 8 m/s
rotary speed
2,00
1,00
0,00
-1,00
-2,00
0,00
5,00
10,00
15,00
20,00
25,00
30,00
35,00
time


Rotary Disturbance of MRS during Motion with Velocity 9 m/s
angle
0,30
0,20
0,10
0,00
-0,10
-0,20
-0,30
0,00
5,00
10,00
15,00
20,00
25,00
30,00
time


Rotary Disturbance of MRS during Motion with Velocity 9 m/s
rotary speed
1,50
1,00
0,50
0,00
-0,50
-1,00
-1,50
0,00
5,00
10,00
15,00
20,00
25,00
30,00
time

APPENDIX C

SOFTWARE LISTING FOR SIMULATION OF DIFFERENT KINEMATIC SCHEMES FOR A SCANNING SYSTEM

```
C     Determination of degeneration zones for 2 degrees
scanator
C     with taking into account oscillation
      INTEGER   i,j,k
      DOUBLE PRECISION
betgr,bet,df1bet,df2bet,Cbet,Sbet,Abet,Wbet,
     +
gamgr,gam,df1gam,df2gam,Cgam,Sgam,Agam,Wgam,
     +              dz0, dzgr ,dz ,df1dz ,df2dz ,Cdz ,Sdz ,
     +                q1gr ,q1 ,df1q1 ,df2q1 ,Cq1 ,Sq1 ,
     +                q2gr ,q2 ,df1q2 ,df2q2 ,Cq2 ,Sq2 ,
     +                   D  ,df1D  ,
     + radius , hich , Hp , Lp , Parm1 , Parm2 , time , hag ,
timek ,
     + pi , Mvetr , sign , Jrot3 , x1 , y1 , z1
      DOUBLE PRECISION    Rc(4,1),Vc(4,1),Ac(4,1),
     +                    T0(4,4,1),DF1T0(4,4,1),DF2T0(4,4,1),
     +                    T(4,4,2),DF1T(4,4,2),DF2T(4,4,2),
     +                    A(4,4,2),B(4,4,2),C(4,4,2),
     +                    TETA(4,4,2),G(4,4,1),
     +                    H(4,4,2),F(4,4,2),L(4,4,2),
     +                    MT(3),Md(3),Mtr(3),P2(3),PR(3),
     +
Matr1(4,4,1),Matr2(4,4,1),Matr3(4,4,1),Matr4(4,4,1),Matr5(4,4,
1)
C************************************************************
      REAL Exit(14,3501) , rhag , Pekv(3) , Mekv(3)
      DATA k/0/
C************************************************************
C
      time =  0.00d0
      timek=         4.99d0
      hag  =  0.01d0
```

```
      radius= 20000.0D0
      hich  =     0.0D0
      READ(*,1002) hich
      dz0   = 3.0d0*3.1415926d0/2.0d0
      Hp= 0.4D0
      Lp= 0.0D0
      Mtr(1)  = 0.830d0
      Mtr(2)  = 0.002d0
      Mtr(3)  = 23.0d0
      Mvetr   = 110.0d0
         Jrot3 = 180.0d0
           Abet= 3.1415926D0*10.0D0/180.0D0
           Wbet= 2.0D0*3.1415926D0/7.0D0
                Agam= 3.1415926D0*20.0D0/180.0D0
                Wgam= 2.0D0*3.1415926D0/5.0D0
      df1dz= 2.0D0*3.1415926D0
      df2dz= 0.0D0
      do 6 i=1,4,1
           do 5 j=1,4,1
             T0(i,j,1)= 0.0D0
5          continue
6     continue
      do 8 i=1,4,1
           do 7 j=1,4,1
             A(i,j,1)= 0.0D0
             A(i,j,2)= 0.0D0
7          continue
8     continue
      do 10 i=1,4,1
            do 9 j=1,4,1
              DF1T0(i,j,1)= 0.0D0
9           continue
10    continue
```

```
      do 12 i=1,4,1
            do 11 j=1,4,1
               TETA(i,j,1)= 0.0D0
               TETA(i,j,2)= 0.0D0
11          continue
12    continue
      TETA(1,2,1)= -1.0D0
      TETA(2,1,1)= +1.0D0
      TETA(1,2,2)= -1.0D0
      TETA(2,1,2)= +1.0D0
      do 14 i=1,4,1
            do 13 j=1,4,1
               DF2T0(i,j,1)= 0.0D0
13          continue
14    continue
      do 16 i=1,4,1
            do 15 j=1,4,1
               H(i,j,1)= 0.0D0
               H(i,j,2)= 0.0D0
15          continue
16    continue
C*************************************************************
      H(1,1,1)= 1.4D0/4.0D0
      H(2,2,1)= 5.0D0*0.4D0**2*(1.0D0/12.0D0+1.0D0/4.0D0)
      H(2,4,1)=-5.0D0*0.4D0/2.0D0
      H(3,3,1)= 1.4D0/4.0D0
      H(4,2,1)=-5.0D0*0.4D0/2.0D0
      H(4,4,1)= 5.0D0
C
      H(1,1,2)= 5.0D0*0.8D0**2/12.0D0
      H(2,2,2)= 5.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(3,3,2)= 5.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(4,4,2)= 5.0D0
```

```
C
      G(3,4,1)= -9.81D0
C
      call MULT ( G,1,H,1,F,1 )
      call MULT ( G,1,H,2,F,2 )
C***************************************************************
666   C O N T I N U E
C***************************************************************
C
      bet= Abet*DSIN( Wbet*time )
      gam= Agam*DSIN( Wgam*time )
      betgr= 180.0d0 * bet / 3.1415926d0
      gamgr= 180.0d0 * gam / 3.1415926d0
C
      df1bet= ( Abet*Wbet )*DCOS( Wbet*time )
      df1gam= ( Agam*Wgam )*DCOS( Wgam*time )
C
      df2bet=-( Abet*Wbet*Wbet )*DSIN( Wbet*time )
      df2gam=-( Agam*Wgam*Wgam )*DSIN( Wgam*time )
C***************************************************************
C
         dz= dz0 + df1dz*time
3     if( dz.le.(2.0D0*3.1415926D0) ) goto 4
          dz= dz-2.0D0*3.1415926D0
      goto 3
4        dzgr= 180.0d0 * dz / 3.1415926d0
C***************************************************************
C
      Cgam= DCOS(gam)
      Sgam= DSIN(gam)
      Cbet= DCOS(bet)
      Sbet= DSIN(bet)
      Cdz = DCOS(dz )
```

```
      Sdz = DSIN(dz )
C
       Rc(1,1)= -Sdz*Cbet*radius-Sbet*hich-30.0d0
       Rc(2,1)= +Cdz*radius+82.0d0/5.0d0*Sgam
            Rc(3,1)= -Sdz*Sbet*radius-
      82.0d0/5.0d0*Cgam+Cbet*hich-0.7d0
       Rc(4,1)= +1.0d0
C
       Vc(1,1)= +Sdz*Sbet*radius*df1bet-Cdz*Cbet*radius*df1dz-
      :          Cbet*hich*df1bet
       Vc(2,1)= -Sdz*radius*df1dz+82.0d0/5.0d0*Cgam*df1gam
       Vc(3,1)= -Sdz*Cbet*radius*df1bet-Cdz*Sbet*radius*df1dz+
      :          82.0d0/5.0d0*Sgam*df1gam-Sbet*hich*df1bet
       Vc(4,1)=  0.0d0
C
       Ac(1,1)= +Sdz*Sbet*radius*df2bet+Sdz*Cbet* radius*
      +df1bet**2+ Sdz*Cbet*radius*df1dz**2+2.0d0*
      +Cdz*Sbet*radius*
      +          df1bet*df1dz-Cdz*Cbet*radius*df2dz+Sbet*hich*
      +         df1bet**2-Cbet*hich*df2bet
       Ac(2,1)= -Sdz*radius*df2dz-Cdz*radius*df1dz**2+
      +          82.0d0/5.0d0*Cgam*df2gam-
      +82.0d0/5.0d0*Sgam*df1gam**2
       Ac(3,1)=
      +Sdz*Sbet*radius*df1bet**2+Sdz*Sbet*radius*df1dz**2-
      +     Sdz*Cbet*radius*df2bet-Cdz*Sbet*radius*df2dz-2.0d0*
      +     Cdz*Cbet*radius*df1bet*df1dz+82.0d0/5.0d0*Cgam*
      +   df1gam**2+82.0d0/5.0d0*Sgam*df2gam-Sbet*hich*df2bet-
      +          Cbet*hich*df1bet**2

       Ac(4,1)=  0.0d0
C************************************************************
C
```

```
      q1= DATAN( -Rc(1,1)/Rc(2,1) )
      q2= DATAN( -(Rc(3,1)-Hp)/DSQRT(Rc(1,1)**2+Rc(2,1)**2) )
      D = DSQRT( Rc(1,1)**2+Rc(2,1)**2+(Rc(3,1)-Hp)**2 )-Lp
C****************************************************************
C
      pi  = 3.1415926D0
      if( (Rc(1,1).eq.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=+pi
      if( (Rc(1,1).gt.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=+pi+q1
      if( (Rc(1,1).lt.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=-pi+q1
        x1=-DSIN(q1)*Rc(1,1)+DCOS(q1)*Rc(2,1)
        y1= Rc(3,1)-Hp
        z1= DCOS(q1)*Rc(1,1)+DSIN(q1)*Rc(2,1)
      if( (y1.eq.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi
      if( (y1.lt.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi+q2
      if( (y1.gt.0.0d0).and.(x1.gt.0.0d0) ) q2=-pi+q2
           q1gr= 180.0d0*q1/pi
           q2gr= 180.0d0*q2/pi
C****************************************************************
C
         T0(1,2,1)= +1.0D0
         T0(1,4,1)= -82.0D0/5.0D0*Sgam
         T0(2,1,1)= -Cbet
         T0(2,3,1)= -Sbet
         T0(2,4,1)= -30.0D0*Cbet-7.0D0/10.0D0*Sbet-
     +                82.0D0/5.0D0*Sbet*Cgam
         T0(3,1,1)= -Sbet
         T0(3,3,1)= +Cbet
         T0(3,4,1)= +7.0D0/10.0D0*Cbet+
     +                82.0D0/5.0D0*Cbet*Cgam-30.0D0*Sbet
         T0(4,4,1)= +1.0D0
C
      Sq1= dsin( q1 )
      Cq1= dcos( q1 )
```

```
      Sq2= dsin( q2 )
      Cq2= dcos( q2 )
C
      A(1,1,1)=-Sq1
      A(1,3,1)= Cq1
      A(2,1,1)= Cq1
      A(2,3,1)= Sq1
      A(3,2,1)= 1.0D0
      A(3,4,1)= Hp
      A(4,4,1)=+1.0D0
C
      A(1,1,2)=-Sq2
      A(1,3,2)=-Cq2
      A(2,1,2)= Cq2
      A(2,3,2)=-Sq2
      A(3,2,2)= 1.0D0
      A(4,4,2)=+1.0D0
C
      call MULT (T0,1,A,1,T,1 )
      call MULT ( T,1,A,2,T,2 )
C*************************************************************
C
      df1q1= ( Cq1*Vc(1,1)+Sq1*Vc(2,1) )/( Cq2*(D+Lp) )
      df1q2= (-Sq1*Sq2*Vc(1,1)+Cq1*Sq2*Vc(2,1)-
Cq2*Vc(3,1))/(D+Lp)
      df1D =   Sq1*Cq2*Vc(1,1)-Cq1*Cq2*Vc(2,1)-Sq2*Vc(3,1)
C*************************************************************
C
         DF1T0(1,4,1)= -82.0D0/5.0D0*Cgam*df1gam
         DF1T0(2,1,1)= +Sbet*df1bet
         DF1T0(2,3,1)= -Cbet*df1bet
         DF1T0(2,4,1)= -82.0D0/5.0D0*Cgam*Cbet*df1bet+
     +                    82.0D0/5.0D0*Sbet*Sgam*df1gam+
```

```
     +                  30.0D0*Sbet*df1bet-
     +                  7.0D0/10.0D0*Cbet*df1bet
        DF1T0(3,1,1)= -Cbet*df1bet
        DF1T0(3,3,1)= -Sbet*df1bet
        DF1T0(3,4,1)= -82.0D0/5.0D0*Cgam*Sbet*df1bet-
     +                  7.0D0/10.0D0*Sbet*df1bet-
     +                  82.0D0/5.0D0*Cbet*Sgam*df1gam-
     +                  30.0D0*Cbet*df1bet
C
      call MULT ( T0,1,TETA,1,Matr1,1 )
      call MULT ( Matr1,1,A,1,B,1 )
      call MULT ( DF1T0,1,A,1,Matr1,1 )
      call MULTS ( B,1,df1q1,Matr2,1 )
      call ZUSAM ( Matr1,1,Matr2,1,DF1T,1,+1 )
C
      call MULT ( T,1,TETA,2,Matr1,1 )
      call MULT ( Matr1,1,A,2,B,2 )
      call MULT ( DF1T,1,A,2,Matr1,1 )
      call MULTS ( B,2,df1q2,Matr2,1 )
      call ZUSAM ( Matr1,1,Matr2,1,DF1T,2,+1 )
C*************************************************************
C
      df2q1= ( 2.0D0*Sq2*(D+Lp)*df1q1*df1q2-
2.0D0*Cq2*df1q1*df1D+
     +             Cq1*Ac(1,1)+Sq1*Ac(2,1) )/( Cq2*(D+Lp) )
      df2q2= (-Sq2*Cq2*(D+Lp)*df1q1**2-2.0D0*df1q2*df1D-
     +      Sq1*Sq2*Ac(1,1)+Cq1*Sq2*Ac(2,1)-Cq2*Ac(3,1)
)/(D+Lp)
C*************************************************************
C
        DF2T0(1,4,1)= -82.0D0/5.0D0*Cgam*df2gam+
     +                  82.0D0/5.0D0*Sgam*df1gam**2
        DF2T0(2,1,1)= +Sbet*df2bet+Cbet*df1bet**2
```

```
      DF2T0(2,3,1)= +Sbet*df1bet**2-Cbet*df2bet
      DF2T0(2,4,1)= +82.0D0/5.0D0*Cgam*Sbet*df1gam**2+
     +                82.0D0/5.0D0*Cgam*Sbet*df1bet**2-
     +                82.0D0/5.0D0*Cgam*Cbet*df2bet+
     +                82.0D0/5.0D0*Sbet*Sgam*df2gam+
     +                7.0D0/10.0D0*Sbet*df1bet**2+
     +                30.0D0*Sbet*df2bet+
     +                164.0D0/5.0D0*Cbet*Sgam*df1gam*df1bet+
     +                30.0D0*Cbet*df1bet**2-
     +                7.0D0/10.0D0*Cbet*df2bet
      DF2T0(3,1,1)= +Sbet*df1bet**2-Cbet*df2bet
      DF2T0(3,3,1)= -Sbet*df2bet-Cbet*df1bet**2
      DF2T0(3,4,1)= -82.0D0/5.0D0*Cgam*Sbet*df2bet-
     +                82.0D0/5.0D0*Cgam*Cbet*df1gam**2-
     +                82.0D0/5.0D0*Cgam*Cbet*df1bet**2+
     +                164.0D0/5.0D0*Sbet*Sgam*df1gam*df1bet+
     +                30.0D0*Sbet*df1bet**2-
     +                7.0D0/10.0D0*Sbet*df2bet-
     +                82.0D0/5.0D0*Cbet*Sgam*df2gam-
     +                7.0D0/10.0D0*Cbet*df1bet**2-
     +                30.0D0*Cbet*df2bet
C
      call MULT ( DF2T0,1,A,1,Matr1,1 )
      call MULTS ( B,1,df2q1,Matr2,1 )
          call MULT ( DF1T0,1,TETA,1,Matr3,1 )
          call MULT ( Matr3,1,A,1,Matr4,1 )
          call MULTS ( Matr4,1, 2.0D0*df1q1 ,Matr3,1 )
            call MULT ( T0,1,TETA,1,Matr4,1 )
            call MULT ( Matr4,1,TETA,1,Matr5,1 )
            call MULT ( Matr5,1,A,1,Matr4,1 )
            call MULTS ( Matr4,1, df1q1**2 ,Matr5,1 )
            call ZUSAM ( Matr3,1,Matr5,1,C,1,+1 )
      call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
```

```
      call ZUSAM ( Matr3,1,C,1,DF2T,1,+1 )
C
      call MULT ( DF2T,1,A,2,Matr1,1 )
      call MULTS ( B,2,df2q2,Matr2,1 )
          call MULT ( DF1T,1,TETA,2,Matr3,1 )
          call MULT ( Matr3,1,A,2,Matr4,1 )
          call MULTS ( Matr4,1, 2.0D0*df1q2 ,Matr3,1 )
            call MULT ( T,1,TETA,2,Matr4,1 )
            call MULT ( Matr4,1,TETA,2,Matr5,1 )
            call MULT ( Matr5,1,A,2,Matr4,1 )
            call MULTS ( Matr4,1, df1q2**2 ,Matr5,1 )
            call ZUSAM ( Matr3,1,Matr5,1,C,2,+1 )
      call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
      call ZUSAM ( Matr3,1,C,2,DF2T,2,+1 )
C*************************************************************
C
      call MULT ( DF2T,2,H,2,Matr1,1 )
      call ZUSAM ( F,2,Matr1,1,L,2,-1 )
C
      call TRNSP ( A,2,Matr1,1 )
      call MULT ( L,2,Matr1,1,Matr2,1 )
          call MULT ( DF2T,1,H,1,Matr1,1 )
              call ZUSAM ( Matr2,1,F,1,Matr3,1,+1 )
           call ZUSAM ( Matr3,1,Matr1,1,L,1,-1 )
C*************************************************************
C
      call TRNSP ( B,1,Matr1,1 )
      call MULT ( L,1,Matr1,1,Matr2,1 )
      call TRACE ( Matr2,1,Parm1 )
      MT(1)= - Parm1
C
      call TRNSP ( B,2,Matr1,1 )
      call MULT ( L,2,Matr1,1,Matr2,1 )
```

```
      call TRACE ( Matr2,1,Parm2 )
      MT(2)= - Parm2
C***************************************************************
      if( df1q1.EQ.0.0d0 ) sign= 0.0d0
      if( df1q1.NE.0.0d0 ) sign= df1q1/DABS( df1q1 )
      Md(1)= MT(1) + sign*Mtr(1)
      if( df1q2.EQ.0.0d0 ) sign= 0.0d0
      if( df1q2.NE.0.0d0 ) sign= df1q2/DABS( df1q2 )
      Md(2)= MT(2) + sign*Mtr(2)
C***************************************************************
      P2(1)= Md(1) * df1q1
      P2(2)= Md(2) * df1q2
C***************************************************************
      PR(1)= Md(1) * df2q1
      PR(2)= Md(2) * df2q2
C***************************************************************
      MT(3)= Jrot3*(-df2gam)
      if( df1gam.EQ.0.0d0 ) sign= 0.0d0
      if( df1gam.NE.0.0d0 ) sign= (-df1gam)/DABS( (-df1gam) )
      Md(3)= MT(3) + sign*Mtr(3) - Mvetr
      P2(3)= Md(3) * (-df1gam)
      PR(3)= Md(3) * (-df2gam)
C***************************************************************
      k= k + 1
      Exit(1,k)= time
      Exit(2,k)= df1q1
      Exit(3,k)= df1q2
      Exit(4,k)= df2q1
      Exit(5,k)= df2q2
      Exit(6,k)= Md(1)
      Exit(7,k)= Md(2)
      Exit(8,k)= Md(3)
      Exit(9,k)=  P2(1)
```

```
      Exit(10,k)= P2(2)
      Exit(11,k)= P2(3)
      Exit(12,k)= PR(1)
      Exit(13,k)= PR(2)
      Exit(14,k)= PR(3)
      WRITE(*,1000)
Pekv,Pekv(1)+Pekv(2)+Pekv(3),Mekv(1),Mekv(2)
      write(1,1001) ((Exit(j,i),j=1,5),i=1,(k-1))
C************************************************************
1001 format(1x,5f12.4)
1002 FORMAT(D12.5)
      END
      SUBROUTINE MULT ( A,I,B,J,C,K )
      INTEGER           I,J,K,L,M,N
      DOUBLE PRECISION A,B,C,SUM
      DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
      DO 3 L=1,4
           DO 2 M=1,4
                SUM=0.0D0
                DO 1 N=1,4
                     SUM=SUM+A(L,N,I)*B(N,M,J)
1               CONTINUE
                C(L,M,K)=SUM
2          CONTINUE
3     CONTINUE
      RETURN
      END
      SUBROUTINE MULTS ( A,I,X,B,J )
      INTEGER           I,J,K,L
      DOUBLE PRECISION A,B,X
      DIMENSION  A(4,4,I),B(4,4,J)
      DO 2 K=1,4
           DO 1 L=1,4
```

```
                B(K,L,J)=A(K,L,I)*X
1           CONTINUE
2       CONTINUE
        RETURN
        END
        SUBROUTINE TRACE ( A,I,X )
        INTEGER           I,J
        DOUBLE PRECISION A,X
        DIMENSION  A(4,4,I)
        X=0.0D0
        DO 1 J=1,4
            X=X+A(J,J,I)
1       CONTINUE
        RETURN
        END
        SUBROUTINE TRNSP ( A,I,B,J )
        INTEGER           I,J,K,L
        DOUBLE PRECISION A,B
        DIMENSION  A(4,4,I),B(4,4,J)
        DO 2 K=1,4
            DO 1 L=1,4
                B(L,K,J)=A(K,L,I)
                B(K,L,J)=A(L,K,I)
1           CONTINUE
2       CONTINUE
        RETURN
        END
        SUBROUTINE ZUSAM ( A,I,B,J,C,K,PAR )
        INTEGER           I,J,K,PAR,L,M
        DOUBLE PRECISION A,B,C
        DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
        IF( PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
        STOP 'ZUSAM: ABEND#1'
```

```
1       DO 3 L=1,4
             DO 2 M=1,4
                  IF( PAR.EQ.+1 ) C(L,M,K)=A(L,M,I)+B(L,M,J)
                  IF( PAR.EQ.-1 ) C(L,M,K)=A(L,M,I)-B(L,M,J)
2            CONTINUE
3       CONTINUE
        RETURN
        END
```

□

```
C     Determination of degeneration zones for 3 degrees
scanning
C     system with taking into account oscillation
      INTEGER   i,j,k
      DOUBLE PRECISION
betgr,bet,df1bet,df2bet,Cbet,Sbet,Abet,Wbet,
     +
gamgr,gam,df1gam,df2gam,Cgam,Sgam,Agam,Wgam,
     +              dz0, dzgr ,dz ,df1dz ,df2dz ,Cdz ,Sdz ,
     +                q1gr ,q1 ,df1q1 ,df2q1 ,Cq1 ,Sq1 ,
     +                q2gr ,q2 ,df1q2 ,df2q2 ,Cq2 ,Sq2 ,
+                q3gr ,q3 ,df1q3 ,df2q3 ,Cq3 ,Sq3 ,
     +                   D  ,df1D  ,
     + radius , hich , Hp , Lp , Parm1 , Parm2 , time , hag ,
timek ,
     + pi , Mvetr , sign , Jrot3 , x1 , y1 , z1
      DOUBLE PRECISION    Rc(4,1),Vc(4,1),Ac(4,1),
     +                    T0(4,4,1),DF1T0(4,4,1),DF2T0(4,4,1),
     +                    T(4,4,3),DF1T(4,4,3),DF2T(4,4,3),
     +                    A(4,4,3),B(4,4,3),C(4,4,3),
     +                    TETA(4,4,2),G(4,4,1),
     +                    H(4,4,2),F(4,4,2),L(4,4,2),
     +                    MT(3),Md(4),Mtr(4),P2(3),PR(3),
     + Matr1(4,4,1),Matr2(4,4,1),Matr3(4,4,1),
     +Matr4(4,4,1),Matr5(4,4,1)
C************************************************************
      REAL Exit(14,3501) , rhag , Pekv(3) , Mekv(3)
      DATA k/0/
C************************************************************
C
      time =  0.00d0
      timek=        4.99d0
      hag  =  0.01d0
```

```
      radius= 20000.0D0
      hich  =     0.0D0
      READ(*,1002) hich
      dz0   = 3.0d0*3.1415926d0/2.0d0
      Hp= 0.4D0
      Lp= 0.0D0
      Mtr(1) = 0.830d0
      Mtr(2) = 0.002d0
      Mtr(3) = 23.0d0
      Mvetr  = 110.0d0
         Jrot3 = 180.0d0
           Abet= 3.1415926D0*10.0D0/180.0D0
           Wbet= 2.0D0*3.1415926D0/7.0D0
                Agam= 3.1415926D0*20.0D0/180.0D0
                Wgam= 2.0D0*3.1415926D0/5.0D0
      df1dz= 2.0D0*3.1415926D0
      df2dz= 0.0D0
      do 6 i=1,4,1
           do 5 j=1,4,1
             T0(i,j,1)= 0.0D0
5          continue
6     continue
      do 8 i=1,4,1
           do 7 j=1,4,1
             A(i,j,1)= 0.0D0
             A(i,j,2)= 0.0D0
             A(i,j,3)= 0.0D0
7          continue
8     continue
      do 10 i=1,4,1
            do 9 j=1,4,1
              DF1T0(i,j,1)= 0.0D0
9           continue
```

```
10    continue
      do 12 i=1,4,1
            do 11 j=1,4,1
               TETA(i,j,1)= 0.0D0
               TETA(i,j,2)= 0.0D0
               TETA(i,j,3)= 0.0D0
11          continue
12    continue
      TETA(1,2,1)= -1.0D0
      TETA(2,1,1)= +1.0D0
      TETA(1,2,2)= -1.0D0
      TETA(2,1,2)= +1.0D0
      TETA(1,2,3)= -1.0D0
      TETA(2,1,3)= +1.0D0
      do 14 i=1,4,1
            do 13 j=1,4,1
               DF2T0(i,j,1)= 0.0D0
13          continue
14    continue
      do 16 i=1,4,1
            do 15 j=1,4,1
               H(i,j,1)= 0.0D0
               H(i,j,2)= 0.0D0
               H(i,j,3)= 0.0D0
15          continue
16    continue
C*************************************************************
      H(1,1,1)= 1.4D0/4.0D0
      H(2,2,1)= 5.0D0*0.4D0**2*(1.0D0/12.0D0+1.0D0/4.0D0)
      H(2,4,1)=-5.0D0*0.4D0/2.0D0
      H(3,3,1)= 1.4D0/4.0D0
      H(4,2,1)=-5.0D0*0.4D0/2.0D0
      H(4,4,1)= 5.0D0
```

```
C
      H(1,1,2)= 5.0D0*0.8D0**2/12.0D0
      H(2,2,2)= 5.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(3,3,2)= 5.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(4,4,2)= 5.0D0
C
      H(1,1,3)= 3.0D0*0.8D0**2/12.0D0
      H(2,2,3)= 3.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(3,3,3)= 3.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(4,4,3)= 3.0D0
      G(3,4,1)= -9.81D0
C
      call MULT ( G,1,H,1,F,1 )
      call MULT ( G,1,H,2,F,2 )
      call MULT ( G,1,H,3,F,3 )
C**************************************************************
666   C O N T I N U E
C**************************************************************
C
      bet= Abet*DSIN( Wbet*time )
      gam= Agam*DSIN( Wgam*time )
      betgr= 180.0d0 * bet / 3.1415926d0
      gamgr= 180.0d0 * gam / 3.1415926d0
C
      df1bet= ( Abet*Wbet )*DCOS( Wbet*time )
      df1gam= ( Agam*Wgam )*DCOS( Wgam*time )
C
      df2bet=-( Abet*Wbet*Wbet )*DSIN( Wbet*time )
      df2gam=-( Agam*Wgam*Wgam )*DSIN( Wgam*time )
C**************************************************************
C
         dz= dz0 + df1dz*time
3     if( dz.le.(2.0D0*3.1415926D0) ) goto 4
```

```
         dz= dz-2.0D0*3.1415926D0
     goto 3
4       dzgr= 180.0d0 * dz / 3.1415926d0
C*************************************************************
C
     Cgam= DCOS(gam)
     Sgam= DSIN(gam)
     Cbet= DCOS(bet)
     Sbet= DSIN(bet)
     Cdz = DCOS(dz )
     Sdz = DSIN(dz )
C
      Rc(1,1)= -Sdz*Cbet*radius-Sbet*hich-30.0d0
      Rc(2,1)= +Cdz*radius+82.0d0/5.0d0*Sgam
      Rc(3,1)= -Sdz*Sbet*radius-82.0d0/5.0d0*Cgam+Cbet*hich-
0.7d0
      Rc(4,1)= +1.0d0
C
      Vc(1,1)= +Sdz*Sbet*radius*df1bet-Cdz*Cbet*radius*df1dz-
     :          Cbet*hich*df1bet
      Vc(2,1)= -Sdz*radius*df1dz+82.0d0/5.0d0*Cgam*df1gam
      Vc(3,1)= -Sdz*Cbet*radius*df1bet-Cdz*Sbet*radius*df1dz+
     :          82.0d0/5.0d0*Sgam*df1gam-Sbet*hich*df1bet
      Vc(4,1)=  0.0d0
C
      Ac(1,1)=+Sdz*Sbet*radius*df2bet+Sdz*Cbet*radius*
     +df1bet**2+Sdz*Cbet*radius*df1dz**2+2.0d0*Cdz*Sbet*
     +radius*df1bet*df1dz-Cdz*Cbet*radius*df2dz+Sbet*hich*
     +df1bet**2-Cbet*hich*df2bet
      Ac(2,1)= -Sdz*radius*df2dz-Cdz*radius*df1dz**2+
     :
```

```
     +82.0d0/5.0d0*Cgam*df2gam-82.0d0/5.0d0*Sgam*df1gam**2
      Ac(3,1)= Sdz*Sbet*radius*df1bet**2+Sdz*Sbet*
     +radius*df1dz**2-Sdz*Cbet*radius*df2bet-
     +Cdz*Sbet*radius*df2dz-2.0d0*
     +Cdz*Cbet*radius*df1bet*df1dz+82.0d0/5.0d0*Cgam*
     +df1gam**2+82.0d0/5.0d0*Sgam*df2gam-Sbet*hich*df2bet-
     +Cbet*hich*df1bet**2
      Ac(4,1)=  0.0d0
C***********************************************************
C
 q1=q1+df1q1*hag
 q2= DATAN(-Rc(1,1)*Cq1+Rc(2,1)*Sq1/Rc(3,1))
 q3= DATAN(Sq2*( Rc(1,1)*Cq1+Rc(2,1)*Sq1)+Rc(3,1)*Cq2)/
+(-Rc(1,1)*Sq1+Rc(2,1)*Cq1))
D = DSQRT( Rc(1,1)**2+Rc(2,1)**2+(Rc(3,1)-Hp)**2 )-Lp
C***********************************************************
C
     pi  = 3.1415926D0
     if( (Rc(1,1).eq.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=+pi
     if( (Rc(1,1).gt.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=+pi+q1
     if( (Rc(1,1).lt.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=-pi+q1
       x1=-DSIN(q1)*Rc(1,1)+DCOS(q1)*Rc(2,1)
       y1= Rc(3,1)-Hp
       z1= DCOS(q1)*Rc(1,1)+DSIN(q1)*Rc(2,1)
     if( (y1.eq.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi
     if( (y1.lt.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi+q2
     if( (y1.gt.0.0d0).and.(x1.gt.0.0d0) ) q2=-pi+q2
          q1gr= 180.0d0*q1/pi
          q2gr= 180.0d0*q2/pi
```

```
C*****************************************************************
C
         T0(1,2,1)= +1.0D0
         T0(1,4,1)= -82.0D0/5.0D0*Sgam
         T0(2,1,1)= -Cbet
         T0(2,3,1)= -Sbet
         T0(2,4,1)= -30.0D0*Cbet-7.0D0/10.0D0*Sbet-
      +                82.0D0/5.0D0*Sbet*Cgam
         T0(3,1,1)= -Sbet
         T0(3,3,1)= +Cbet
         T0(3,4,1)= +7.0D0/10.0D0*Cbet+
      +                82.0D0/5.0D0*Cbet*Cgam-30.0D0*Sbet
         T0(4,4,1)= +1.0D0
C
      Sq1= dsin( q1 )
      Cq1= dcos( q1 )
      Sq2= dsin( q2 )
      Cq2= dcos( q2 )
C
      A(1,1,1)=-Sq1
      A(1,3,1)= Cq1
      A(2,1,1)= Cq1
      A(2,3,1)= Sq1
      A(3,2,1)= 1.0D0
      A(3,4,1)= Hp
      A(4,4,1)=+1.0D0
C
      A(1,1,2)=-Sq2
      A(1,3,2)=-Cq2
      A(2,1,2)= Cq2
      A(2,3,2)=-Sq2
      A(3,2,2)= 1.0D0
      A(4,4,2)=+1.0D0
```

```
C
      call MULT (T0,1,A,1,T,1 )
      call MULT ( T,1,A,2,T,2 )
      call MULT ( T,2,A,3,T,3 )
      call MULT ( T,2,A,4,T,4 )
C***************************************************************
C
df2q1=((Cq3(d2*(Cq2)**2+d3*(Sq2*Sq3)**2*(cq1*Vc(1)+Sq1*Vc(2))+
+ d2*Sq2*(sq3)**2*(sq1*Vc(1)-Cq1*Vc(2)+Sq2*Cq2*Sq3(d3(Sq3)**2-
d2)*
+ Vc(3))/(D*(d1*Sq3**2+d2*((Cq2*Cq3)**2+d3*((Sq2*Sq3)**2)-
+ df1q1)/hag
 df1q1=
(Cq3(d2*(Cq2)**2+d3*(Sq2*Sq3)**2*(cq1*Vc(1)+Sq1*Vc(2))+
+ d2*Sq2*(sq3)**2*(sq1*Vc(1)-Cq1*Vc(2)+Sq2*Cq2*Sq3(d3(Sq3)**2-
d2)*
+ Vc(3))/(D*(d1*Sq3**2+d2*((Cq2*Cq3)**2+d3*((Sq2*Sq3)**2)
 df1q2=(-Cq1*Cq2*Vc(1,1)-Sq1*Cq2*Vc(2,1)+Sq2*Vc(3,1)+df1q1*
+Cq2*Cq3*D)/(D*Sq3)
 df1q3=(Vc(1,1)*(-Sq1*Sq3-Cq1*Sq2*Cq3-Cq2*Cq3*Vc(3,1)+df1q1*D*
+Sq2+(Cq1*Sq3-Sq1*Sq2*Cq3)*Vc(2,1))/D
 df1D =    Sq1*Cq2*Vc(1,1)-Cq1*Cq2*Vc(2,1)-Sq2*Vc(3,1)
C***************************************************************
C
        DF1T0(1,4,1)= -82.0D0/5.0D0*Cgam*df1gam
        DF1T0(2,1,1)= +Sbet*df1bet
        DF1T0(2,3,1)= -Cbet*df1bet
        DF1T0(2,4,1)= -82.0D0/5.0D0*Cgam*Cbet*df1bet+
     +                 82.0D0/5.0D0*Sbet*Sgam*df1gam+
     +                 30.0D0*Sbet*df1bet-
     +                 7.0D0/10.0D0*Cbet*df1bet
        DF1T0(3,1,1)= -Cbet*df1bet
        DF1T0(3,3,1)= -Sbet*df1bet
```

```
      DF1T0(3,4,1)= -82.0D0/5.0D0*Cgam*Sbet*df1bet-
     +                7.0D0/10.0D0*Sbet*df1bet-
     +                82.0D0/5.0D0*Cbet*Sgam*df1gam-
     +                30.0D0*Cbet*df1bet
C
      call MULT ( T0,1,TETA,1,Matr1,1 )
      call MULT ( Matr1,1,A,1,B,1 )
      call MULT ( DF1T0,1,A,1,Matr1,1 )
      call MULTS ( B,1,df1q1,Matr2,1 )
      call ZUSAM ( Matr1,1,Matr2,1,DF1T,1,+1 )
C
      call MULT ( T,1,TETA,2,Matr1,1 )
      call MULT ( Matr1,1,A,2,B,2 )
      call MULT ( DF1T,1,A,2,Matr1,1 )
      call MULTS ( B,2,df1q2,Matr2,1 )
      call ZUSAM ( Matr1,1,Matr2,1,DF1T,2,+1 )
C
      call MULT ( T,3,TETA,2,Matr1,1 )
      call MULT ( Matr1,1,A,3,B,2 )
      call MULT ( DF1T,1,A,3,Matr1,1 )
      call MULTS ( B,2,df1q3,Matr2,1 )
      call ZUSAM ( Matr1,1,Matr2,1,DF1T,3,+1 )
C*************************************************************
C
 df2q2= (-Cq1*cq2*Ac(1)-
Sq1*Cq2*Ac(2)+sq2*Ac(3)+Cq2*Cq3*D*df2q1+
+ Cq2*sq2*Sq3*D*df1q1**2+2.0d0*Cq2*q3*df1D*df1q1-
2.0d0*Cq2*Sq3*
+df1q1*df1q3*D-2.0d0Sq3df1D*df1q2-
2.0d0*Cq3*df1q2*df1q3*D)/(D*Sq3)
C
 df2q3=((-Sq1*Sq3-Cq1*Sq2*Cq3)*Ac(1)+(Cq1*Sq3-
Sq1*Sq2*Cq3)*Ac(2)-
```

```
+Cq2*Cq3*Ac(3)+sq2*d*df2q1-q2**2*cq3*sq3*D*df1q1**2+2.0d0Sq2*
+df1D*df1q1+2.0d0*Cq2*Sq3**2*df1q2*df1q1*D-2.0d0*df1D*df1q3+
+2.0d0*df1q3*sq3*D*df1q2*2)/D
C************************************************************
C
        DF2T0(1,4,1)= -82.0D0/5.0D0*Cgam*df2gam+
     +                 82.0D0/5.0D0*Sgam*df1gam**2
        DF2T0(2,1,1)= +Sbet*df2bet+Cbet*df1bet**2
        DF2T0(2,3,1)= +Sbet*df1bet**2-Cbet*df2bet
        DF2T0(2,4,1)= +82.0D0/5.0D0*Cgam*Sbet*df1gam**2+
     +                 82.0D0/5.0D0*Cgam*Sbet*df1bet**2-
     +                 82.0D0/5.0D0*Cgam*Cbet*df2bet+
     +                 82.0D0/5.0D0*Sbet*Sgam*df2gam+
     +                 7.0D0/10.0D0*Sbet*df1bet**2+
     +                 30.0D0*Sbet*df2bet+
     +                 164.0D0/5.0D0*Cbet*Sgam*df1gam*df1bet+
     +                 30.0D0*Cbet*df1bet**2-
     +                 7.0D0/10.0D0*Cbet*df2bet
        DF2T0(3,1,1)= +Sbet*df1bet**2-Cbet*df2bet
        DF2T0(3,3,1)= -Sbet*df2bet-Cbet*df1bet**2
        DF2T0(3,4,1)= -82.0D0/5.0D0*Cgam*Sbet*df2bet-
     +                 82.0D0/5.0D0*Cgam*Cbet*df1gam**2-
     +                 82.0D0/5.0D0*Cgam*Cbet*df1bet**2+
     +                 164.0D0/5.0D0*Sbet*Sgam*df1gam*df1bet+
     +                 30.0D0*Sbet*df1bet**2-
     +                 7.0D0/10.0D0*Sbet*df2bet-
     +                 82.0D0/5.0D0*Cbet*Sgam*df2gam-
     +                 7.0D0/10.0D0*Cbet*df1bet**2-
     +                 30.0D0*Cbet*df2bet
C
     call MULT ( DF2T0,1,A,1,Matr1,1 )
     call MULTS ( B,1,df2q1,Matr2,1 )
         call MULT ( DF1T0,1,TETA,1,Matr3,1 )
```

```
        call MULT ( Matr3,1,A,1,Matr4,1 )
        call MULTS ( Matr4,1, 2.0D0*df1q1 ,Matr3,1 )
          call MULT ( T0,1,TETA,1,Matr4,1 )
          call MULT ( Matr4,1,TETA,1,Matr5,1 )
          call MULT ( Matr5,1,A,1,Matr4,1 )
          call MULTS ( Matr4,1, df1q1**2 ,Matr5,1 )
          call ZUSAM ( Matr3,1,Matr5,1,C,1,+1 )
    call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
    call ZUSAM ( Matr3,1,C,1,DF2T,1,+1 )
C
    call MULT ( DF2T,1,A,2,Matr1,1 )
    call MULTS ( B,2,df2q2,Matr2,1 )
        call MULT ( DF1T,1,TETA,2,Matr3,1 )
        call MULT ( Matr3,1,A,2,Matr4,1 )
        call MULTS ( Matr4,1, 2.0D0*df1q2 ,Matr3,1 )
          call MULT ( T,1,TETA,2,Matr4,1 )
          call MULT ( Matr4,1,TETA,2,Matr5,1 )
          call MULT ( Matr5,1,A,2,Matr4,1 )
          call MULTS ( Matr4,1, df1q2**2 ,Matr5,1 )
          call ZUSAM ( Matr3,1,Matr5,1,C,2,+1 )
    call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
    call ZUSAM ( Matr3,1,C,2,DF2T,2,+1 )
C
    call MULT ( DF2T,1,A,3,Matr1,1 )
    call MULTS ( B,2,df2q3,Matr2,1 )
        call MULT ( DF1T,1,TETA,2,Matr3,1 )
        call MULT ( Matr3,1,A,2,Matr4,1 )
        call MULTS ( Matr4,1, 2.0D0*df1q2 ,Matr3,1 )
          call MULT ( T,1,TETA,2,Matr4,1 )
          call MULT ( Matr4,1,TETA,2,Matr5,1 )
          call MULT ( Matr5,1,A,2,Matr4,1 )
          call MULTS ( Matr4,1, df1q3**2 ,Matr5,1 )
          call ZUSAM ( Matr3,1,Matr5,1,C,2,+1 )
```

```
      call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
      call ZUSAM ( Matr3,1,C,2,DF2T,2,+1 )
C***************************************************************
C
      call MULT ( DF2T,2,H,3,Matr1,1 )
      call ZUSAM ( F,3,Matr1,1,L,3,-1 )
C
      call TRNSP ( A,2,Matr1,1 )
      call MULT ( L,2,Matr1,1,Matr2,1 )
          call MULT ( DF2T,1,H,1,Matr1,1 )
              call ZUSAM ( Matr2,1,F,1,Matr3,1,+1 )
           call ZUSAM ( Matr3,1,Matr1,1,L,1,-1 )
C***************************************************************
C
      call TRNSP ( B,1,Matr1,1 )
      call MULT ( L,1,Matr1,1,Matr2,1 )
      call TRACE ( Matr2,1,Parm1 )
      MT(1)= - Parm1
C
      call TRNSP ( B,2,Matr1,1 )
      call MULT ( L,2,Matr1,1,Matr2,1 )
      call TRACE ( Matr2,1,Parm2 )
      MT(2)= - Parm2
C
      call TRNSP ( B,3,Matr1,1 )
      call MULT ( L,3,Matr1,1,Matr2,1 )
      call TRACE ( Matr2,1,Parm2 )
      MT(2)= - Parm3
C***************************************************************
      if( dflq1.EQ.0.0d0 ) sign= 0.0d0
      if( dflq1.NE.0.0d0 ) sign= dflq1/DABS( dflq1 )
      Md(1)= MT(1) + sign*Mtr(1)
      if( dflq2.EQ.0.0d0 ) sign= 0.0d0
```

```
      if( df1q2.NE.0.0d0 ) sign= df1q2/DABS( df1q2 )
      Md(2)= MT(2) + sign*Mtr(2)
      if( df1q3.EQ.0.0d0 ) sign= 0.0d0
      if( df1q3.NE.0.0d0 ) sign= df1q3/DABS( df1q3 )
      Md(3)= MT(3) + sign*Mtr(3)
C************************************************************
      P2(1)= Md(1) * df1q1
      P2(2)= Md(2) * df1q2
      P2(3)= Md(3) * df1q3
C************************************************************
      PR(1)= Md(1) * df2q1
      PR(2)= Md(2) * df2q2
      PR(3)= Md(3) * df2q3
C************************************************************
      MT(3)= Jrot3*(-df2gam)
      if( df1gam.EQ.0.0d0 ) sign= 0.0d0
      if( df1gam.NE.0.0d0 ) sign= (-df1gam)/DABS( (-df1gam) )
      Md(3)= MT(3) + sign*Mtr(3) - Mvetr
      P2(3)= Md(3) * (-df1gam)
      PR(3)= Md(3) * (-df2gam)
C************************************************************
      k= k + 1
      Exit(1,k)= time
      Exit(2,k)= df1q1
      Exit(3,k)= df1q3
      Exit(4,k)= df2q1
      Exit(5,k)= df2q3
      Exit(6,k)= Md(1)
      Exit(7,k)= Md(2)
      Exit(8,k)= Md(3)
      Exit(9,k)=  P2(1)
      Exit(10,k)= P2(2)
      Exit(11,k)= P2(3)
```

```
      Exit(12,k)= PR(1)
      Exit(13,k)= PR(2)
      Exit(14,k)= PR(3)
      write(1,1001) ((Exit(j,i),j=1,5),i=1,(k-1))
C*************************************************************
1001  format(1x,5f12.4)
1002  FORMAT(D12.5)
      END
      SUBROUTINE MULT ( A,I,B,J,C,K )
      INTEGER            I,J,K,L,M,N
      DOUBLE PRECISION A,B,C,SUM
      DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
      DO 3 L=1,4
           DO 2 M=1,4
                SUM=0.0D0
                DO 1 N=1,4
                     SUM=SUM+A(L,N,I)*B(N,M,J)
1               CONTINUE
                C(L,M,K)=SUM
2          CONTINUE
3     CONTINUE
      RETURN
      END
      SUBROUTINE MULTS ( A,I,X,B,J )
      INTEGER           I,J,K,L
      DOUBLE PRECISION A,B,X
      DIMENSION  A(4,4,I),B(4,4,J)
      DO 2 K=1,4
           DO 1 L=1,4
                B(K,L,J)=A(K,L,I)*X
1          CONTINUE
2     CONTINUE
      RETURN
```

```
      END
      SUBROUTINE TRACE ( A,I,X )
      INTEGER          I,J
      DOUBLE PRECISION A,X
      DIMENSION  A(4,4,I)
      X=0.0D0
      DO 1 J=1,4
           X=X+A(J,J,I)
1     CONTINUE
      RETURN
      END
      SUBROUTINE TRNSP ( A,I,B,J )
      INTEGER          I,J,K,L
      DOUBLE PRECISION A,B
      DIMENSION  A(4,4,I),B(4,4,J)
      DO 2 K=1,4
           DO 1 L=1,4
                B(L,K,J)=A(K,L,I)
                B(K,L,J)=A(L,K,I)
1          CONTINUE
2     CONTINUE
      RETURN
      END
      SUBROUTINE ZUSAM ( A,I,B,J,C,K,PAR )
      INTEGER          I,J,K,PAR,L,M
      DOUBLE PRECISION A,B,C
      DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
      IF( PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
      STOP 'ZUSAM: ABEND#1'
1     DO 3 L=1,4
           DO 2 M=1,4
                IF( PAR.EQ.+1 ) C(L,M,K)=A(L,M,I)+B(L,M,J)
                IF( PAR.EQ.-1 ) C(L,M,K)=A(L,M,I)-B(L,M,J)
```

```
2         CONTINUE
3     CONTINUE
      RETURN
      END
```

```
      subroutine DNM3 ( Mn )
C
C
      INTEGER   i,j,k
        DOUBLE PRECISION   df1q, df2q, A, TETA, T0, DF1T0,
DF2T0, H, F,
     +T(4,4,3), DF1T(4,4,3), DF2T(4,4,3), B(4,4,3), C(4,4,3),
L(4,4,3),
     +Matr(4,4,5), Parm, Mn(3)
*-----------------------
        common /In_DNM3/   df1q(3), df2q(3),      A(4,4,3),
TETA(4,4,3),
     +  T0(4,4,1), DF1T0(4,4,1), DF2T0(4,4,1),   H(4,4,3),
F(4,4,3)
Comment.     Step_1  :
*-----------------------
      call MULT (T0,1,A,1,T,1 )
      do 1 i= 2,3,1
        j= i-1
        call MULT ( T,j,A,i,T,i )
1     continue
*
Comment.     Step_2  :
*-----------------------
      call MULT ( T0,1,TETA,1,Matr,1 )
      call MULT ( Matr,1,A,1,B,1 )
      call MULT ( DF1T0,1,A,1,Matr,1 )
      call MULTS ( B,1,df1q(1),Matr,2 )
      call ZUSAM ( Matr,1,Matr,2,DF1T,1,+1 )
      do 2 i= 2,3,1
        j= i-1
        call MULT ( T,j,TETA,i,Matr,1 )
```

```
      call MULT ( Matr,1,A,i,B,i )
      call MULT ( DF1T,j,A,i,Matr,1 )
      call MULTS ( B,i,df1q(i),Matr,2 )
      call ZUSAM ( Matr,1,Matr,2,DF1T,i,+1 )
2    continue
Comment.     Step_3  :
*-----------------------
     call MULT ( DF2T0,1,A,1,Matr,1 )
     call MULTS ( B,1,df2q(1),Matr,2 )
         call MULT ( DF1T0,1,TETA,1,Matr,3 )
         call MULT ( Matr,3,A,1,Matr,4 )
         call MULTS ( Matr,4, 2.0d0*df1q(1) ,Matr,3 )
           call MULT ( T0,1,TETA,1,Matr,4 )
           call MULT ( Matr,4,TETA,1,Matr,5 )
           call MULT ( Matr,5,A,1,Matr,4 )
           call MULTS ( Matr,4, df1q(1)**2 ,Matr,5 )
           call ZUSAM ( Matr,3,Matr,5,C,1,+1 )
     call ZUSAM ( Matr,1,Matr,2,Matr,3,+1 )
     call ZUSAM ( Matr,3,C,1,DF2T,1,+1 )
     do 3 i= 2,3,1
       j= i-1
       call MULT ( DF2T,j,A,i,Matr,1 )
        call MULTS ( B,i,df2q(i),Matr,2 )
         call MULT ( DF1T,j,TETA,i,Matr,3 )
         call MULT ( Matr,3,A,i,Matr,4 )
         call MULTS ( Matr,4, 2.0d0*df1q(i) ,Matr,3 )
           call MULT ( T,j,TETA,i,Matr,4 )
           call MULT ( Matr,4,TETA,i,Matr,5 )
           call MULT ( Matr,5,A,i,Matr,4 )
           call MULTS ( Matr,4, df1q(i)**2 ,Matr,5 )
           call ZUSAM ( Matr,3,Matr,5,C,i,+1 )
       call ZUSAM ( Matr,1,Matr,2,Matr,3,+1 )
```

```
          call ZUSAM ( Matr,3,C,i,DF2T,i,+1 )
3       continue
*
Comment.      Step_4  :
*----------------------
        call MULT ( DF2T,3,H,3,Matr,1 )
        call ZUSAM ( F,3,Matr,1,L,3,-1 )
        do 4 k= 1,2,1
          i= 3-k
          j= i+1
          call TRNSP ( A,j,Matr,1 )
          call MULT ( L,j,Matr,1,Matr,2 )
            call MULT ( DF2T,i,H,i,Matr,1 )
              call ZUSAM ( Matr,2,F,i,Matr,3,+1 )
             call ZUSAM ( Matr,3,Matr,1,L,i,-1 )
4       continue
*
Comment.      Step_5:
*--------------------------------------------------------------
--------
        do 5 i= 1,3,1
          call TRNSP ( B,i,Matr,1 )
          call MULT ( L,i,Matr,1,Matr,2 )
          call TRACE ( Matr,2,Parm )
          Mn(i)= -Parm
5       continue
        return
        END
        SUBROUTINE MULT ( A,I,B,J,C,K )
        INTEGER          I,J,K,L,M,N
        DOUBLE PRECISION A,B,C,SUM
        DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
```

```
      DO 3 L=1,4
           DO 2 M=1,4
                SUM=0.0D0
                DO 1 N=1,4
                     SUM=SUM+A(L,N,I)*B(N,M,J)
1               CONTINUE
                C(L,M,K)=SUM
2          CONTINUE
3     CONTINUE
      RETURN
      END
      SUBROUTINE MULTS ( A,I,X,B,J )
      INTEGER          I,J,K,L
      DOUBLE PRECISION A,B,X
      DIMENSION  A(4,4,I),B(4,4,J)
      DO 2 K=1,4
           DO 1 L=1,4
                B(K,L,J)=A(K,L,I)*X
1          CONTINUE
2     CONTINUE
      RETURN
      END
      SUBROUTINE TRACE ( A,I,X )
      INTEGER          I,J
      DOUBLE PRECISION A,X
      DIMENSION  A(4,4,I)
      X=0.0D0
      DO 1 J=1,4
           X=X+A(J,J,I)
1     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRNSP ( A,I,B,J )
      INTEGER          I,J,K,L
      DOUBLE PRECISION A,B
      DIMENSION  A(4,4,I),B(4,4,J)
      DO 2 K=1,4
           DO 1 L=1,4
                B(L,K,J)=A(K,L,I)
                B(K,L,J)=A(L,K,I)
1          CONTINUE
2     CONTINUE
      RETURN
      END
      SUBROUTINE ZUSAM ( A,I,B,J,C,K,PAR )
      INTEGER          I,J,K,PAR,L,M
      DOUBLE PRECISION A,B,C
      DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
      IF( PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
      STOP 'ZUSAM: ABEND#1'
1     DO 3 L=1,4
           DO 2 M=1,4
                IF( PAR.EQ.+1 ) C(L,M,K)=A(L,M,I)+B(L,M,J)
                IF( PAR.EQ.-1 ) C(L,M,K)=A(L,M,I)-B(L,M,J)
2          CONTINUE
3     CONTINUE
      RETURN
      END
```

```
      subroutine DNM4 ( Mn )
C
C
      INTEGER   i,j,k
        DOUBLE PRECISION   df1q, df2q, A, TETA, T0, DF1T0,
DF2T0, H, F,
      +T(4,4,4), DF1T(4,4,4), DF2T(4,4,4), B(4,4,4), C(4,4,4),
L(4,4,4),
      +Matr(4,4,5), Parm, Mn(4)
*----------------------
        common /In_DNM4/   df1q(4), df2q(4),      A(4,4,4),
TETA(4,4,4),
      +  T0(4,4,1), DF1T0(4,4,1), DF2T0(4,4,1),   H(4,4,4),
F(4,4,4)
*
Comment.     Step_1  :
*----------------------
      call MULT (T0,1,A,1,T,1 )
      do 1 i= 2,4,1
        j= i-1
        call MULT ( T,j,A,i,T,i )
1     continue
*
Comment.     Step_2  :
*----------------------
      call MULT ( T0,1,TETA,1,Matr,1 )
      call MULT ( Matr,1,A,1,B,1 )
      call MULT ( DF1T0,1,A,1,Matr,1 )
      call MULTS ( B,1,df1q(1),Matr,2 )
      call ZUSAM ( Matr,1,Matr,2,DF1T,1,+1 )
      do 2 i= 2,4,1
        j= i-1
```

```
          call MULT  ( T,j,TETA,i,Matr,1 )
          call MULT  ( Matr,1,A,i,B,i )
          call MULT  ( DF1T,j,A,i,Matr,1 )
          call MULTS  ( B,i,df1q(i),Matr,2 )
          call ZUSAM  ( Matr,1,Matr,2,DF1T,i,+1 )
2       continue
*
Comment.      Step_3   :
*-----------------------
        call MULT ( DF2T0,1,A,1,Matr,1 )
        call MULTS ( B,1,df2q(1),Matr,2 )
            call MULT  ( DF1T0,1,TETA,1,Matr,3 )
            call MULT  ( Matr,3,A,1,Matr,4 )
            call MULTS ( Matr,4, 2.0d0*df1q(1) ,Matr,3 )
              call MULT  ( T0,1,TETA,1,Matr,4 )
              call MULT  ( Matr,4,TETA,1,Matr,5 )
              call MULT  ( Matr,5,A,1,Matr,4 )
              call MULTS  ( Matr,4, df1q(1)**2 ,Matr,5 )
              call ZUSAM  ( Matr,3,Matr,5,C,1,+1 )
        call ZUSAM ( Matr,1,Matr,2,Matr,3,+1 )
        call ZUSAM ( Matr,3,C,1,DF2T,1,+1 )
        do 3 i= 2,4,1
          j= i-1
          call MULT  ( DF2T,j,A,i,Matr,1 )
           call MULTS  ( B,i,df2q(i),Matr,2 )
            call MULT  ( DF1T,j,TETA,i,Matr,3 )
            call MULT  ( Matr,3,A,i,Matr,4 )
            call MULTS  ( Matr,4, 2.0d0*df1q(i) ,Matr,3 )
              call MULT  ( T,j,TETA,i,Matr,4 )
              call MULT  ( Matr,4,TETA,i,Matr,5 )
              call MULT  ( Matr,5,A,i,Matr,4 )
              call MULTS  ( Matr,4, df1q(i)**2 ,Matr,5 )
```

```
          call ZUSAM ( Matr,3,Matr,5,C,i,+1 )
        call ZUSAM ( Matr,1,Matr,2,Matr,3,+1 )
        call ZUSAM ( Matr,3,C,i,DF2T,i,+1 )
3     continue
*
Comment.      Step_4   :
*----------------------
      call MULT ( DF2T,4,H,4,Matr,1 )
      call ZUSAM ( F,4,Matr,1,L,4,-1 )
      do 4 k= 1,3,1
        i= 4-k
        j= i+1
        call TRNSP ( A,j,Matr,1 )
        call MULT ( L,j,Matr,1,Matr,2 )
          call MULT ( DF2T,i,H,i,Matr,1 )
            call ZUSAM ( Matr,2,F,i,Matr,3,+1 )
           call ZUSAM ( Matr,3,Matr,1,L,i,-1 )
4     continue
Comment.      Step_5   (  Mn(1)...Mn(4) ) :
*-------------------------------------------------------------
      do 5 i= 1,4,1
        call TRNSP ( B,i,Matr,1 )
        call MULT ( L,i,Matr,1,Matr,2 )
        call TRACE ( Matr,2,Parm )
        Mn(i)= -Parm
5     continue
      return
      END
      SUBROUTINE MULT ( A,I,B,J,C,K )
      INTEGER          I,J,K,L,M,N
      DOUBLE PRECISION A,B,C,SUM
      DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
```

```
      DO 3 L=1,4
           DO 2 M=1,4
                SUM=0.0D0
                DO 1 N=1,4
                     SUM=SUM+A(L,N,I)*B(N,M,J)
1               CONTINUE
                C(L,M,K)=SUM
2          CONTINUE
3     CONTINUE
      RETURN
      END
      SUBROUTINE MULTS ( A,I,X,B,J )
      INTEGER         I,J,K,L
      DOUBLE PRECISION A,B,X
      DIMENSION  A(4,4,I),B(4,4,J)
      DO 2 K=1,4
           DO 1 L=1,4
                B(K,L,J)=A(K,L,I)*X
1          CONTINUE
2     CONTINUE
      RETURN
      END
      SUBROUTINE TRACE ( A,I,X )
      INTEGER          I,J
      DOUBLE PRECISION A,X
      DIMENSION  A(4,4,I)
      X=0.0D0
      DO 1 J=1,4
           X=X+A(J,J,I)
1     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRNSP ( A,I,B,J )
      INTEGER          I,J,K,L
      DOUBLE PRECISION A,B
      DIMENSION  A(4,4,I),B(4,4,J)
      DO 2 K=1,4
           DO 1 L=1,4
                B(L,K,J)=A(K,L,I)
                B(K,L,J)=A(L,K,I)
1          CONTINUE
2     CONTINUE
      RETURN
      END
      SUBROUTINE ZUSAM ( A,I,B,J,C,K,PAR )
      INTEGER          I,J,K,PAR,L,M
      DOUBLE PRECISION A,B,C
      DIMENSION  A(4,4,I),B(4,4,J),C(4,4,K)
      IF( PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
      STOP 'ZUSAM: ABEND#1'
1     DO 3 L=1,4
           DO 2 M=1,4
                IF( PAR.EQ.+1 ) C(L,M,K)=A(L,M,I)+B(L,M,J)
                IF( PAR.EQ.-1 ) C(L,M,K)=A(L,M,I)-B(L,M,J)
2          CONTINUE
3     CONTINUE
      RETURN
      END
```

```
C       Scanning of environment under MRC motion
c            (for two degrees scanning system)
c
*************************************************************
        INTEGER i,j,k
        DOUBLE PRECISION  D,df1D,df2D,
     +  q1,df1q1,df2q1,Cq1,Sq1,q2,df1q2,df2q2,Cq2,Sq2,Hp,Lp,
     +  hag, zam1,zam2,zam3,zam4,time,
     +  basex, basey, Amplit,m,Vmrc, koef1,PAR1,
     +  alf, df1alf, df2alf, Salf, Calf,
     +  bet, df1bet, df2bet, Sbet, Cbet,
     +  x1,y1,z1,pi
        DOUBLE PRECISION TETA1(4,4),
     +  T0(4,4),DF1T0(4,4),DF2T0(4,4),T11(4,4),
     +  Matr1(4,4),Matr2(4,4),Matr3(4,4),MATR4(4,4),
     +  MATR5(4,4),MATR6(4,4),vect1(4),vect2(4),Vect3(4),
     +   A10(4,4),A20(4,4)
        DOUBLE PRECISION
Rc(4),Vc(4),Ac(4),Rcprom(4),Vcprom(4),Acprom(4)
        REAL    ex(7,6001)
        OPEN(1,FILE='POSIT2.DAT')
        OPEN(2,FILE='speed2.DAT')
        OPEN(3,FILE='acccel2.DAT')
        Hp= 0.5d0
        Lp= 0.3d0
        Koef1= 2.0d0
        m= 2.0d0
        Amplit= 0.63d0
        basex= 3.4d0
        basey= 2.5d0
        time= 0.00d0
        Vmrc= 5.0d0
```

```
        hag= 0.01d0
        D=  1000.0d0
        Rc(4)= 1.0D0
        RcPROM(4)= 1.0D0
        Rcprom(1)= -250.0d0
        Rcprom(2)=-1000.0d0
        Rcprom(3)= 1.0d0
        zam1= 2.0d0*3.14159d0*amplit*VMRC/(m*basex*3.0d0)
        zam2= 2.0d0*3.14159d0/(m*basex)
        zam3= 2.0d0*3.14159d0*amplit*VMRC/(m*basey*3.0d0)
        zam4= 2.0d0*3.14159d0/(m*basey)
          do 1 i=1,4,1
        do 2 j=1,4,1
        Tetal(i,j)=0.0d0
        A10(i,j)= 0.0d0
        A20(i,j)= 0.0d0
2       Continue
1       Continue
        do 3 i=1,4,1
        Vcprom(i)= 0.0d0
        Acprom(i)= 0.0d0
3       Continue
C*************************************************************
        Tetal(1,2)=-1.0d0
        Tetal(2,1)= 1.0d0
c*************************************************************
c       Scanning mode
c*************************************************************
        Do 666 k=1,6001,1
        time=time+hag
         par1=1.0d0
         if (k .LT. 11)  Go to 11
```

```
if (k .LT. 91)  Go to 12
if (k .LT. 101) Go to 13
if (k .LT. 201) Go to 17
if (k .LT. 211) Go to 14
if (k .LT. 291) Go to 15
if (k .LT. 301) Go to 16
if (k .LT. 401) Go to 17
if (k .LT. 411)  Go to 11
if (k .LT. 491)  Go to 12
if (k .LT. 501) Go to 13
if (k .LT. 601) Go to 17
par1=-1.0d0
if (k .LT. 611) Go to 14
if (k .LT. 691) Go to 15
if (k .LT. 701) Go to 16
if (k .LT. 801) Go to 17
par1=1.0d0
if (k .LT. 811)  Go to 11
if (k .LT. 891)  Go to 12
if (k .LT. 901) Go to 13
if (k .LT. 1001) Go to 17
if (k .LT. 1011) Go to 14
if (k .LT. 1091) Go to 15
if (k .LT. 1101) Go to 16
if (k .LT. 1201) Go to 17
if (k .LT. 1211)  Go to 11
if (k .LT. 1291)  Go to 12
if (k .LT. 1301) Go to 13
if (k .LT. 1401) Go to 17
par1=-1.0d0
if (k .LT. 1411) Go to 14
if (k .LT. 1491) Go to 15
```

```
if (k .LT. 1501) Go to 16
if (k .LT. 1601) Go to 17
par1=1.0d0
if (k .LT. 1611)  Go to 11
if (k .LT. 1691)  Go to 12
if (k .LT. 1701) Go to 13
if (k .LT. 1801) Go to 17
if (k .LT. 1811) Go to 14
if (k .LT. 1891) Go to 15
if (k .LT. 1901) Go to 16
if (k .LT. 2001) Go to 17
if (k .LT. 2011)  Go to 11
if (k .LT. 2091)  Go to 12
if (k .LT. 2101) Go to 13
if (k .LT. 2201) Go to 17
par1=-1.0d0
if (k .LT. 2211) Go to 14
if (k .LT. 2291) Go to 15
if (k .LT. 2301) Go to 16
if (k .LT. 2401) Go to 17
par1=1.0d0
if (k .LT. 2411)  Go to 11
if (k .LT. 2491)  Go to 12
if (k .LT. 2501) Go to 13
if (k .LT. 2601) Go to 17
if (k .LT. 2611) Go to 14
if (k .LT. 2691) Go to 15
if (k .LT. 2701) Go to 16
if (k .LT. 2801) Go to 17
if (k .LT. 2811)  Go to 11
if (k .LT. 2891)  Go to 12
if (k .LT. 2901) Go to 13
```

```
if (k .LT. 3001) Go to 17
par1=-1.0d0
if (k .LT. 3011) Go to 14
if (k .LT. 3091) Go to 15
if (k .LT. 3101) Go to 16
if (k .LT. 3201) Go to 17
par1= 1.0d0
if (k .LT. 3211)  Go to 11
if (k .LT. 3291)  Go to 12
if (k .LT. 3301) Go to 13
if (k .LT. 3401) Go to 17
if (k .LT. 3411) Go to 14
if (k .LT. 3491) Go to 15
if (k .LT. 3501) Go to 16
if (k .LT. 3601) Go to 17
if (k .LT. 3611) Go to 14
if (k .LT. 3691) Go to 15
if (k .LT. 3701) Go to 16
if (k .LT. 3801) Go to 17
par1=-1.0d0
if (k .LT. 3811)  Go to 11
if (k .LT. 3891)  Go to 12
if (k .LT. 3901) Go to 13
if (k .LT. 4001) Go to 17
if (k .LT. 4011)  Go to 11
if (k .LT. 4091)  Go to 12
if (k .LT. 4101) Go to 13
if (k .LT. 4201) Go to 17
if (k .LT. 4211)  Go to 11
if (k .LT. 4291)  Go to 12
if (k .LT. 4301) Go to 13
if (k .LT. 4401) Go to 17
```

```
if (k .LT. 4411)  Go to 11
if (k .LT. 4491)  Go to 12
if (k .LT. 4501) Go to 13
if (k .LT. 4601) Go to 17
if (k .LT. 4611)  Go to 11
if (k .LT. 4691)  Go to 12
if (k .LT. 4701) Go to 13
if (k .LT. 4801) Go to 17
if (k .LT. 4811)  Go to 11
if (k .LT. 4891)  Go to 12
if (k .LT. 4901) Go to 13
if (k .LT. 5001) Go to 17
if (k .LT. 5011)  Go to 11
if (k .LT. 5091)  Go to 12
if (k .LT. 5101) Go to 13
if (k .LT. 5201) Go to 17
if (k .LT. 5211)  Go to 11
if (k .LT. 5291)  Go to 12
if (k .LT. 5301) Go to 13
if (k .LT. 5401) Go to 17
if (k .LT. 5411)  Go to 11
if (k .LT. 5491)  Go to 12
if (k .LT. 5501) Go to 13
if (k .LT. 5601) Go to 17
if (k .LT. 5611)  Go to 14
if (k .LT. 5691)  Go to 15
if (k .LT. 5701) Go to 16
if (k .LT. 5801) Go to 17
if (k .LT. 5811)  Go to 14
if (k .LT. 5891)  Go to 15
if (k .LT. 5901) Go to 16
 Go to 17
```

```
11        call move1(
Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
12          call move2
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
13          call move3
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
14        call move4
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
15        call move5
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
16       call move6
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
17        call stop (Rcprom,Vcprom,Acprom,vect1,vect2,vect3)
          go to 6
6         do 5 j=1,4,1
          Rcprom(j)= Vect1(J)
          Vcprom(j)= Vect2(J)
          Acprom(j)= Vect3(J)
5         Continue
c***********************************************************
c            Introduction of base rocking
c***********************************************************
32        alf=zam1*DCOS(zam2*time)
          df1alf=-zam1*zam2*Dsin(zam2*time)
          df2alf=-zam1*zam2*zam2*DCOS(zam2*time)
          Salf= DSIN(alf)
```

```
      Calf= DCOS(alf)
      bet= zam3*DCOS(zam4*time)
      df1bet=-zam3*zam4*Dsin(zam4*time)
      df2bet=-zam3*zam4*zam4*DCOS(zam4*time)
      Sbet= DSIN(bet)
      Cbet= DCOS(bet)
c**************************************************************
********
c     definition motion with taking into consideration base
rocking
c**************************************************************
********
      T11(1,1)= Cbet
      T11(1,2)= Salf*Sbet
      T11(1,3)=-Calf*Sbet
      T11(1,4)= 0.0d0
      T11(2,1)= 0.0d0
      T11(2,2)= Calf
      T11(2,3)= Salf
      T11(2,4)= Lp
      T11(3,1)= Sbet
      T11(3,2)=-Salf*Cbet
      T11(3,3)= Calf*Cbet
      T11(3,4)=-Hp
      T11(4,1)= 0.0d0
      T11(4,2)= 0.0d0
      T11(4,3)= 0.0d0
      T11(4,4)= 1.0d0
C
      T0(1,1)= Cbet
      T0(1,2)= 0.0d0
      T0(1,3)= Sbet
```

```
      T0(1,4)= Hp*Sbet
      T0(2,1)= Salf*Sbet
      T0(2,2)= Calf
      T0(2,3)=-Salf*Cbet
      T0(2,4)=-Lp*Calf-Hp*Salf*Cbet
      T0(3,1)=-Calf*Salf
      T0(3,2)= Salf
      T0(3,3)= Calf*Cbet
      T0(3,4)=-Lp*Salf+Hp*Calf*Cbet
      T0(4,1)= 0.0d0
      T0(4,2)= 0.0d0
      T0(4,3)= 0.0d0
      T0(4,4)= 1.0d0
C
      A10(1,1)= 1.0d0
      A10(2,2)= Calf
      A10(2,3)=-Salf
      A10(3,2)= Salf
      A10(3,3)= Calf
      A10(4,4)= 1.0d0
c
      A20(1,1)= Cbet
      A20(1,3)= Sbet
      A20(1,4)= Hp*Sbet
      A20(2,2)= 1.0d0
      A20(2,4)=-Lp
      A20(3,1)=-Sbet
      A20(3,3)= Cbet
      A20(3,4)= Hp*Cbet
      A20(4,4)= 1.0d0
c*****************************************************************
```

```
c       Definition the target vector with taking into acount
rocking
c*****************************************************************
        call MULTS1(T11,Rcprom,Rc)
c
        call MULT(Teta1,T0,Matr1)
        call MULTS(Matr1,df1alf,Matr2)
        call MULT(A10,TETA1,MATR1)
        call MULT(MATR1,A20,MATR3)
        call MULTS(MATR3,DF1bet,MATR4)
        call ZUSAM(MATR2,MATR4,DF1T0,1)
C
        call MULTS1(DF1T0,Rcprom,vect1)
        call MULTS2(Vcprom,vect1,vect2,-1)
        call MULTS1(T11,vect2,Vc)
C
        call MULT(TETA1,T0,MATR1)
        call MULTS(MATR1,DF2alf,MATR2)
        call MULT(A10,TETA1,MATR1)
        call MULT(MATR1,A20,MATR3)
        call MULTS(MATR3,DF2bet,MATR1)
        call MULTS(TETA1,KOEF1,MATR3)
        call MULT(MATR3,A10,MATR4)
        call MULT(MATR4,teta1,MATR3)
        call MULT(MATR3,A20,MATR4)
        call MULTS(MATR4,DF1alf,MATR5)
        call MULTS(MATR3,DF1bet,MATR3)
        call MULT(TETA1,TETA1,MATR4)
        call MULT(MATR4,T0,MATR5)
        call MULTS(MATR5,DF1alf,MATR6)
        call MULTS(MATR6,DF1alf,MATR4)
        call MULT(teta1,teta1,MATR5)
```

```
      call MULT(A10,MATR5,MATR6)
      call MULT(MATR6,A20,MATR5)
      call MULTS(MATR5,DF1bet,MATR6)
      call MULTS(MATR6,DF1bet,MATR5)
      call ZUSAM(MATR1,MATR2,MATR6,1)
      call ZUSAM(MATR6,MATR3,MATR1,1)
      call ZUSAM(MATR1,MATR4,MATR2,1)
      call ZUSAM(MATR2,MATR5,DF2T0,1)
c
      call MULTS(DF1T0,KOEF1,MATR1)
      call MULTS1(MATR1,Vcprom,vect1)
      call MULTS1(DF2T0,RcPROM,vect2)
      call MULTS2(vect1,vect2,vect3,1)
      call MULTS2(Acprom,vect3,vect1,-1)
      call MULTS1(T11,vect1,Ac)
c*****************************************************************
c     Decision of inverse kinematic task in positions
c*****************************************************************
       q1= DATAN(-Rc(1)/Rc(2))
    q2= DATAN(-(Rc(3)-Hp)/DSQRT(Rc(1)**2+Rc(2)**2))
    D = DSQRT(Rc(1)**2+Rc(2)**2+(Rc(3)-Hp)**2)-Lp
c***************************************************************
c****************************************************************
    pi  = 3.1415926D0
    if( (Rc(1).eq.0.0d0).and.(Rc(2).gt.0.0d0) ) q1=+pi
    if( (Rc(1).gt.0.0d0).and.(Rc(2).gt.0.0d0) ) q1=+pi+q1
    if( (Rc(1).lt.0.0d0).and.(Rc(2).gt.0.0d0) ) q1=-pi+q1
      x1=-DSIN(q1)*Rc(1)+DCOS(q1)*Rc(2)
      y1= Rc(3)-Hp
      z1= DCOS(q1)*Rc(1)+DSIN(q1)*Rc(2)
    if( (y1.eq.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi
    if( (y1.lt.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi+q2
```

```
      if( (y1.gt.0.0d0).and.(x1.gt.0.0d0) ) q2=-pi+q2
         Cq1= DCOS(q1)
         Sq1= Dsin(q1)
         Cq2= DCOS(q2)
         Sq2= Dsin(q2)
C************************************************************
C
      df1q1= ( Cq1*Vc(1)+Sq1*Vc(2) )/( Cq2*(D+Lp) )
      df1q2= (-Sq1*Sq2*Vc(1)+Cq1*Sq2*Vc(2)-Cq2*Vc(3))/(D+Lp)
      df1D =   Sq1*Cq2*Vc(1)-Cq1*Cq2*Vc(2)-Sq2*Vc(3)
C************************************************************
C
      df2q1= ( 2.0D0*Sq2*(D+Lp)*df1q1*df1q2-
2.0D0*Cq2*df1q1*df1D+
     +           Cq1*Ac(1)+Sq1*Ac(2) )/( Cq2*(D+Lp) )
      df2q2= (-Sq2*Cq2*(D+Lp)*df1q1**2-2.0D0*df1q2*df1D-
     +           Sq1*Sq2*Ac(1)+Cq1*Sq2*Ac(2)-Cq2*Ac(3))/(D+Lp)
C************************************************************
C        Output results
C**************************************************************
*****
         ex(1,k)= time
         ex(2,k)= q1
         ex(3,k)= q2
         ex(4,k)= df1q1
         ex(5,k)= df1q2
         ex(6,k)= df2q1
         ex(7,k)= df2q2
         write(1,1001) ex(1,k),ex(2,k),ex(3,k)
         write(2,1001) ex(1,k),ex(4,k),ex(5,k)
         write(3,1001) ex(1,k),ex(6,k),ex(7,k)
666      Continue
```

```
1001    format(1x,3e12.4)
        close(1)
        close(2)
        close(3)
        stop
        END
      Subroutine Move1 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
      Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
     + Vmrc,hag,Arazg,PAR1
      Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
        hag= 0.01
        VMRC= 5.0d0
        Arazg= 500.0d0
        Acpro(1)= Arazg*par1
        Acpro(2)= 0.0d0
        Acpro(3)= 0.0d0
        Acpro(4)= 0.0d0
        Vcpro(1)= Vcprom(1)+Acpro(1)*hag
        Vcpro(2)= VMRC
        Vcpro(3)= 0.0d0
        Vcpro(4)= 0.0d0
        Rcpro(1)= Rcprom(1)+Vcpro(1)*hag
        Rcpro(2)= Rcprom(2)+Vmrc*hag
        Rcpro(3)= Rcprom(3)
        Rcpro(4)= 1.0d0
        RETURN
        END
      Subroutine Move2 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
      Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
```

```
     + Vmrc,hag,vhor,PAR1
       Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
         hag= 0.01
         VMRC= 5.0d0
         vhor= 50.0d0
         Acpro(1)= 0.0d0
         Acpro(2)= 0.0d0
         Acpro(3)= 0.0d0
         Acpro(4)= 0.0d0
         Vcpro(1)= Vhor*par1
         Vcpro(2)= VMRC
         Vcpro(3)= 0.0d0
         Vcpro(4)= 0.0d0
         Rcpro(1)= Rcprom(1)+Vhor*hag*par1
         Rcpro(2)= Rcprom(2)+Vmrc*hag
         Rcpro(3)= Rcprom(3)
         Rcpro(4)= 1.0d0
         RETURN
         END
       Subroutine Move3 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
       Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
     + Vmrc,hag,Atorm,PAR1
       Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
         hag= 0.01
         VMRC= 5.0d0
         Atorm= 500.0d0
         Acpro(1)=-Atorm
         Acpro(2)= 0.0d0
         Acpro(3)= 0.0d0
```

```
          Acpro(4)= 0.0d0
          Vcpro(1)= Vcprom(1)-Atorm*hag*par1
          Vcpro(2)= VMRC
          Vcpro(3)= 0.0d0
          Vcpro(4)= 0.0d0
          Rcpro(1)= Rcprom(1)+Vcpro(1)*hag
          Rcpro(2)= Rcprom(2)+Vmrc*hag
          Rcpro(3)= Rcprom(3)
          Rcpro(4)= 1.0d0
          RETURN
          END
       Subroutine Move4 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
       Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
     + Vmrc,hag,Arazg,PAR1
       Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
          hag= 0.01
          VMRC= 5.0d0
          Arazg= 1200.0d0
          Acpro(1)= 0.0d0
          Acpro(2)= 0.0d0
          Acpro(3)= Arazg*par1
          Acpro(4)= 0.0d0
          Vcpro(1)= 0.0d0
          Vcpro(2)= VMRC
          Vcpro(3)= Vcprom(3)+Arazg*hag*par1
          Vcpro(4)= 0.0d0
          Rcpro(1)= Rcprom(1)
          Rcpro(2)= Rcprom(2)+Vmrc*hag
          Rcpro(3)= Rcprom(3)+Vcpro(3)*hag
          Rcpro(4)= 1.0d0
```

```
          RETURN
          END
        Subroutine Move5 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
        Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
       + Vmrc,hag,vver,PAR1
        Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
          hag= 0.01
          VMRC= 5.0d0
          vver= 120.0d0
          Acpro(1)= 0.0d0
          Acpro(2)= 0.0d0
          Acpro(3)= 0.0d0
          Acpro(4)= 0.0d0
          Vcpro(1)= 0.0d0
          Vcpro(2)= VMRC
          Vcpro(3)= Vver*par1
          Vcpro(4)= 0.0d0
          Rcpro(1)= Rcprom(1)
          Rcpro(2)= Rcprom(2)+Vmrc*hag
          Rcpro(3)= Rcprom(3)+vver*hag*par1
          Rcpro(4)= 1.0d0
          RETURN
          END
        Subroutine Move6 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
        Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
       + Vmrc,hag,Atorm,PAR1
        Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
          hag= 0.01
```

```
      VMRC= 5.0d0
      Atorm= 1200.0d0
      Acpro(1)= 0.0d0
      Acpro(2)= 0.0d0
      Acpro(3)= -Atorm*par1
      Acpro(4)= 0.0d0
      Vcpro(1)= 0.0d0
      Vcpro(2)= VMRC
      Vcpro(3)= Vcprom(3)-Atorm*hag*par1
      Vcpro(4)= 0.0d0
      Rcpro(1)= Rcprom(1)
      Rcpro(2)= Rcprom(2)+Vmrc*hag
      Rcpro(3)= Rcprom(3)+Vcpro(3)*hag
      Rcpro(4)= 1.0d0
      RETURN
      END
    Subroutine Stop (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro)
    Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
   + Vmrc,hag
    Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
      hag= 0.01
      VMRC= 5.0d0
      Acpro(1)= 0.0d0
      Acpro(2)= 0.0d0
      Acpro(3)= 0.0d0
      Acpro(4)= 0.0d0
      Vcpro(1)= 0.0d0
      Vcpro(2)= VMRC
      Vcpro(3)= 0.0d0
      Vcpro(4)= 0.0d0
```

```
      Rcpro(1)= Rcprom(1)
      Rcpro(2)= Rcprom(2)+Vmrc*hag
      Rcpro(3)= Rcprom(3)
      Rcpro(4)= 1.0d0
      RETURN
      END
      SUBROUTINE MULT ( A,B,C)
      INTEGER          L,M,N
      DOUBLE PRECISION A,B,C,SUM
      DIMENSION  A(4,4),B(4,4),C(4,4)
      DO 3 L=1,4
          DO 2 M=1,4
              SUM=0.0D0
              DO 1 N=1,4
                  SUM=SUM+A(L,N)*B(N,M)
1             CONTINUE
              C(L,M)=SUM
2         CONTINUE
3     CONTINUE
      RETURN
      END
      SUBROUTINE MULTS1 ( A,X,y )
      INTEGER          K,L
      DOUBLE PRECISION A,X,y
      DIMENSION  A(4,4),x(4),y(4)
      DO 2 K=1,4
         y(k)= 0.0d0
          DO 1 L=1,4
              Y(k)=A(K,L)*X(L)+Y(k)
1         CONTINUE
2     CONTINUE
      RETURN
```

```
      END
      SUBROUTINE TRACE ( A,X )
      INTEGER          J
      DOUBLE PRECISION A,X
      DIMENSION  A(4,4)
      X=0.0D0
      DO 1 J=1,4
           X=X+A(J,J)
1     CONTINUE
      RETURN
      END
      SUBROUTINE TRNSP ( A,B)
      INTEGER          K,L
      DOUBLE PRECISION A,B
      DIMENSION  A(4,4),B(4,4)
      DO 2 K=1,4
           DO 1 L=1,4
                B(L,K)=A(K,L)
                B(K,L)=A(L,K)
1          CONTINUE
2     CONTINUE
      RETURN
      END
      SUBROUTINE ZUSAM ( A,B,C,PAR )
      INTEGER          PAR,L,M
      DOUBLE PRECISION A,B,C
      DIMENSION  A(4,4),B(4,4),C(4,4)
      IF( PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
      STOP 'ZUSAM: ABEND#1'
1     DO 3 L=1,4
           DO 2 M=1,4
                IF( PAR.EQ.+1 ) C(L,M)=A(L,M)+B(L,M)
```

```
                  IF( PAR.EQ.-1 ) C(L,M)=A(L,M)-B(L,M)
2            CONTINUE
3       CONTINUE
        RETURN
        END
        SUBROUTINE MULTS2 (A,B,C,PAR)
        INTEGER PAR,L
        DOUBLE PRECISION A,B,C
        DIMENSION A(4),B(4),C(4)
        IF (PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
        STOP 'MULTS2: ABEND#1'
1       DO 3 L=1,4,1
        C(L)=A(L)+B(L)*PAR
3       CONTINUE
        RETURN
        END
        SUBROUTINE MULTS ( A,X,B)
        INTEGER          K,L
        DOUBLE PRECISION A,B,X
        DIMENSION  A(4,4),B(4,4)
        DO 2 K=1,4
             DO 1 L=1,4
                  B(K,L)=A(K,L)*X
1            CONTINUE
2       CONTINUE
        RETURN
        END
```

```
C       Scanning of environment under MRC motion
c            (for three degree scanning system)
c
*****************************************************************
        INTEGER i,j,k,limit1,limit2,count3,count4
        DOUBLE PRECISION  D,df1D,df2D,
     +  q1,df1q1,df2q1,Cq1,Sq1,q2,df1q2,df2q2,Cq2,Sq2,Hp,Lp,
     +  hag, zam1,zam2,zam3,zam4,PAR1,X1,Y1,Z1,
     +  time,time1,time2,timepr,
     +  basex, basey, Amplit,m, VMRC,
     + time11,time12,time31,time32,time33,Count1,count2,KOEF1,
     +  Vhor, Vver, Ahrazg, Avrazg, Ahtorm, Avtorm,
     +  alf, df1alf, df2alf, Salf, Calf,
     +  bet, df1bet, df2bet, Sbet, Cbet,
     +  Md1,Md2,p1,p2,parm1,parm2
        DOUBLE PRECISION TETA1(4,4),H1(4,4),H2(4,4),G1(4,4),
     +  L1(4,4),L2(4,4),A1(4,4),A2(4,4),T1(4,4),T2(4,4),
     +  df1T1(4,4),df1T2(4,4),df2T1(4,4),df2T2(4,4),
     +  T0(4,4),DF1T0(4,4),DF2T0(4,4),T11(4,4),
     +  Matr1(4,4),Matr2(4,4),Matr3(4,4),MATR4(4,4),
     +  MATR5(4,4),MATR6(4,4),vect1(4),vect2(4),Vect3(4),
     +  A10(4,4),A20(4,4),F1(4,4),F2(4,4),
     +  B1(4,4),C1(4,4),B2(4,4),C2(4,4)
        DOUBLE PRECISION
Rc(4),Vc(4),Ac(4),Rcprom(4),Vcprom(4),Acprom(4)
        REAL     ex(7,6001)
        OPEN(1,FILE='POSIT2.DAT')
        OPEN(2,FILE='speed2.DAT')
        OPEN(3,FILE='acccel2.DAT')
        Hp= 0.5d0
        Lp= 0.3d0
        Koef1= 2.0d0
```

```
      m= 2.0d0
      Amplit= 0.63d0
      basex= 3.4d0
      basey= 2.5d0
      time= 0.00d0
      Vmrc= 5.0d0
      hag= 0.01d0
      D=  1000.0d0
      Rc(4)= 1.0D0
      RcPROM(4)= 1.0D0
      Rcprom(1)= -250.0d0
      Rcprom(2)=-1000.0d0
      Rcprom(3)= 1.0d0
      zam1= 2.0d0*3.14159d0*amplit*VMRC/(m*basex*3.0d0)
      zam2= 2.0d0*3.14159d0/(m*basex)
      zam3= 2.0d0*3.14159d0*amplit*VMRC/(m*basey*3.0d0)
      zam4= 2.0d0*3.14159d0/(m*basey)
      do 3 i=1,4,1
      Vcprom(i)= 0.0d0
      Acprom(i)= 0.0d0
3     Continue
C*************************************************************
       do 9 i=1,4,1
       do 8 j=1,4,1
      Teta1(i,j)=0.0d0
      A10(i,j)= 0.0d0
      A20(i,j)= 0.0d0
      A1(i,j)= 0.0d0
      A2(i,j)= 0.0d0
      H1(i,j)= 0.0d0
      H2(i,j)= 0.0d0
      G1(i,j)= 0.0d0
```

```
8         Continue
9         Continue
C***************************************************************
C
      H1(1,1)= 1.4D0/4.0D0
      H(1,1,1)= 1.4D0/4.0D0
      H(2,2,1)= 5.0D0*0.4D0**2*(1.0D0/12.0D0+1.0D0/4.0D0)
      H(2,4,1)=-5.0D0*0.4D0/2.0D0
      H(3,3,1)= 1.4D0/4.0D0
      H(4,2,1)=-5.0D0*0.4D0/2.0D0
      H(4,4,1)= 5.0D0
C
      H(1,1,2)= 5.0D0*0.8D0**2/12.0D0
      H(2,2,2)= 5.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(3,3,2)= 5.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(4,4,2)= 5.0D0
C
      H(1,1,3)= 3.0D0*0.8D0**2/12.0D0
      H(2,2,3)= 3.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(3,3,3)= 3.0D0*(0.21D0/5.0D0-0.8D0**2/12.0D0)
      H(4,4,3)= 3.0D0
      G(3,4,1)= -9.81D0
C
      call MULT ( G,1,H,1,F,1 )
      call MULT ( G,1,H,2,F,2 )
      call MULT ( G,1,H,3,F,3)

C***************************************************************
         Teta1(1,2)=-1.0d0
         Teta1(2,1)= 1.0d0
c***************************************************************
c        Scanning mode
```

```
C*************************************************************
      Do 666 k=1,6001,1
       time=time+hag
       par1=1.0d0
       if (k .LT. 11)  Go to 11
       if (k .LT. 91)  Go to 12
       if (k .LT. 101) Go to 13
       if (k .LT. 201) Go to 17
       if (k .LT. 211) Go to 14
       if (k .LT. 291) Go to 15
       if (k .LT. 301) Go to 16
       if (k .LT. 401) Go to 17
       if (k .LT. 411)  Go to 11
       if (k .LT. 491)  Go to 12
       if (k .LT. 501) Go to 13
       if (k .LT. 601) Go to 17
       par1=-1.0d0
       if (k .LT. 611) Go to 14
       if (k .LT. 691) Go to 15
       if (k .LT. 701) Go to 16
       if (k .LT. 801) Go to 17
       par1=1.0d0
       if (k .LT. 811)  Go to 11
       if (k .LT. 891)  Go to 12
       if (k .LT. 901) Go to 13
       if (k .LT. 1001) Go to 17
       if (k .LT. 1011) Go to 14
       if (k .LT. 1091) Go to 15
       if (k .LT. 1101) Go to 16
       if (k .LT. 1201) Go to 17
       if (k .LT. 1211)  Go to 11
       if (k .LT. 1291)  Go to 12
```

```
if (k .LT. 1301) Go to 13
if (k .LT. 1401) Go to 17
par1=-1.0d0
if (k .LT. 1411) Go to 14
if (k .LT. 1491) Go to 15
if (k .LT. 1501) Go to 16
if (k .LT. 1601) Go to 17
par1=1.0d0
if (k .LT. 1611)  Go to 11
if (k .LT. 1691)  Go to 12
if (k .LT. 1701) Go to 13
if (k .LT. 1801) Go to 17
if (k .LT. 1811) Go to 14
if (k .LT. 1891) Go to 15
if (k .LT. 1901) Go to 16
if (k .LT. 2001) Go to 17
if (k .LT. 2011)  Go to 11
if (k .LT. 2091)  Go to 12
if (k .LT. 2101) Go to 13
if (k .LT. 2201) Go to 17
par1=-1.0d0
if (k .LT. 2211) Go to 14
if (k .LT. 2291) Go to 15
if (k .LT. 2301) Go to 16
if (k .LT. 2401) Go to 17
par1=1.0d0
if (k .LT. 2411)  Go to 11
if (k .LT. 2491)  Go to 12
if (k .LT. 2501) Go to 13
if (k .LT. 2601) Go to 17
if (k .LT. 2611) Go to 14
if (k .LT. 2691) Go to 15
```

```
if (k .LT. 2701) Go to 16
if (k .LT. 2801) Go to 17
if (k .LT. 2811)  Go to 11
if (k .LT. 2891)  Go to 12
if (k .LT. 2901) Go to 13
if (k .LT. 3001) Go to 17
par1=-1.0d0
if (k .LT. 3011) Go to 14
if (k .LT. 3091) Go to 15
if (k .LT. 3101) Go to 16
if (k .LT. 3201) Go to 17
par1= 1.0d0
if (k .LT. 3211)  Go to 11
if (k .LT. 3291)  Go to 12
if (k .LT. 3301) Go to 13
if (k .LT. 3401) Go to 17
if (k .LT. 3411) Go to 14
if (k .LT. 3491) Go to 15
if (k .LT. 3501) Go to 16
if (k .LT. 3601) Go to 17
if (k .LT. 3611) Go to 14
if (k .LT. 3691) Go to 15
if (k .LT. 3701) Go to 16
if (k .LT. 3801) Go to 17
par1=-1.0d0
if (k .LT. 3811)  Go to 11
if (k .LT. 3891)  Go to 12
if (k .LT. 3901) Go to 13
if (k .LT. 4001) Go to 17
if (k .LT. 4011)  Go to 11
if (k .LT. 4091)  Go to 12
if (k .LT. 4101) Go to 13
```

```
if (k .LT. 4201) Go to 17
if (k .LT. 4211)  Go to 11
if (k .LT. 4291)  Go to 12
if (k .LT. 4301) Go to 13
if (k .LT. 4401) Go to 17
if (k .LT. 4411)  Go to 11
if (k .LT. 4491)  Go to 12
if (k .LT. 4501) Go to 13
if (k .LT. 4601) Go to 17
if (k .LT. 4611)  Go to 11
if (k .LT. 4691)  Go to 12
if (k .LT. 4701) Go to 13
if (k .LT. 4801) Go to 17
if (k .LT. 4811)  Go to 11
if (k .LT. 4891)  Go to 12
if (k .LT. 4901) Go to 13
if (k .LT. 5001) Go to 17
if (k .LT. 5011)  Go to 11
if (k .LT. 5091)  Go to 12
if (k .LT. 5101) Go to 13
if (k .LT. 5201) Go to 17
if (k .LT. 5211)  Go to 11
if (k .LT. 5291)  Go to 12
if (k .LT. 5301) Go to 13
if (k .LT. 5401) Go to 17
if (k .LT. 5411)  Go to 11
if (k .LT. 5491)  Go to 12
if (k .LT. 5501) Go to 13
if (k .LT. 5601) Go to 17
if (k .LT. 5611)  Go to 14
if (k .LT. 5691)  Go to 15
if (k .LT. 5701) Go to 16
```

```
          if (k .LT. 5801) Go to 17
          if (k .LT. 5811)  Go to 14
          if (k .LT. 5891)  Go to 15
          if (k .LT. 5901) Go to 16
           Go to 17
11        call move1(
Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
12          call move2
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
13          call move3
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
14        call move4
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
15        call move5
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
16       call move6
(Rcprom,Vcprom,Acprom,vect1,vect2,vect3,PAR1)
          go to 6
17        call stop (Rcprom,Vcprom,Acprom,vect1,vect2,vect3)
          go to 6
6         do 5 j=1,4,1
          Rcprom(j)= Vect1(J)
          Vcprom(j)= Vect2(J)
          Acprom(j)= Vect3(J)
5         Continue
c************************************************************
c              Introduction of base rocking
```

```
c*************************************************************
32      alf=zam1*DCOS(zam2*time)
        df1alf=-zam1*zam2*Dsin(zam2*time)
        df2alf=-zam1*zam2*zam2*DCOS(zam2*time)
        Salf= DSIN(alf)
        Calf= DCOS(alf)
        bet= zam3*DCOS(zam4*time)
        df1bet=-zam3*zam4*Dsin(zam4*time)
        df2bet=-zam3*zam4*zam4*DCOS(zam4*time)
        Sbet= DSIN(bet)
        Cbet= DCOS(bet)
c**************************************************************
********
c       definition motion with taking into consideration base
rocking
c**************************************************************
********
       T11(1,1)= Cbet
       T11(1,2)= Salf*Sbet
       T11(1,3)=-Calf*Sbet
       T11(1,4)= 0.0d0
       T11(2,1)= 0.0d0
       T11(2,2)= Calf
       T11(2,3)= Salf
       T11(2,4)= Lp
       T11(3,1)= Sbet
       T11(3,2)=-Salf*Cbet
       T11(3,3)= Calf*Cbet
       T11(3,4)=-Hp
       T11(4,1)= 0.0d0
       T11(4,2)= 0.0d0
       T11(4,3)= 0.0d0
```

```
      T11(4,4)= 1.0d0
C
      T0(1,1)= Cbet
      T0(1,2)= 0.0d0
      T0(1,3)= Sbet
      T0(1,4)= Hp*Sbet
      T0(2,1)= Salf*Sbet
      T0(2,2)= Calf
      T0(2,3)=-Salf*Cbet
      T0(2,4)=-Lp*Calf-Hp*Salf*Cbet
      T0(3,1)=-Calf*Salf
      T0(3,2)= Salf
      T0(3,3)= Calf*Cbet
      T0(3,4)=-Lp*Salf+Hp*Calf*Cbet
      T0(4,1)= 0.0d0
      T0(4,2)= 0.0d0
      T0(4,3)= 0.0d0
      T0(4,4)= 1.0d0
C
      A10(1,1)= 1.0d0
      A10(2,2)= Calf
      A10(2,3)=-Salf
      A10(3,2)= Salf
      A10(3,3)= Calf
      A10(4,4)= 1.0d0
c
      A20(1,1)= Cbet
      A20(1,3)= Sbet
      A20(1,4)= Hp*Sbet
      A20(2,2)= 1.0d0
      A20(2,4)=-Lp
      A20(3,1)=-Sbet
```

```
      A20(3,3)= Cbet
      A20(3,4)= Hp*Cbet
      A20(4,4)= 1.0d0
c*****************************************************************
c      Definition the target vector with taking into acount
rocking
c*****************************************************************
      call MULTS1(T11,Rcprom,Rc)
c
      call MULT(Teta1,T0,Matr1)
      call MULTS(Matr1,df1alf,Matr2)
      call MULT(A10,TETA1,MATR1)
      call MULT(MATR1,A20,MATR3)
      call MULTS(MATR3,DF1bet,MATR4)
      call ZUSAM(MATR2,MATR4,DF1T0,1)
c
      call MULTS1(DF1T0,Rcprom,vect1)
      call MULTS2(Vcprom,vect1,vect2,-1)
      call MULTS1(T11,vect2,Vc)
c
      call MULT(TETA1,T0,MATR1)
      call MULTS(MATR1,DF2alf,MATR2)
      call MULT(A10,TETA1,MATR1)
      call MULT(MATR1,A20,MATR3)
      call MULTS(MATR3,DF2bet,MATR1)
      call MULTS(TETA1,KOEF1,MATR3)
      call MULT(MATR3,A10,MATR4)
      call MULT(MATR4,teta1,MATR3)
      call MULT(MATR3,A20,MATR4)
      call MULTS(MATR4,DF1alf,MATR5)
      call MULTS(MATR3,DF1bet,MATR3)
      call MULT(TETA1,TETA1,MATR4)
```

```
        call MULT(MATR4,T0,MATR5)
        call MULTS(MATR5,DF1alf,MATR6)
        call MULTS(MATR6,DF1alf,MATR4)
        call MULT(teta1,teta1,MATR5)
        call MULT(A10,MATR5,MATR6)
        call MULT(MATR6,A20,MATR5)
        call MULTS(MATR5,DF1bet,MATR6)
        call MULTS(MATR6,DF1bet,MATR5)
        call ZUSAM(MATR1,MATR2,MATR6,1)
        call ZUSAM(MATR6,MATR3,MATR1,1)
        call ZUSAM(MATR1,MATR4,MATR2,1)
        call ZUSAM(MATR2,MATR5,DF2T0,1)
C
        call MULTS(DF1T0,KOEF1,MATR1)
        call MULTS1(MATR1,Vcprom,vect1)
        call MULTS1(DF2T0,RcPROM,vect2)
        call MULTS2(vect1,vect2,vect3,1)
        call MULTS2(Acprom,vect3,vect1,-1)
        call MULTS1(T11,vect1,Ac)
q1=q1+df1q1*hag
□
 q2= DATAN(-Rc(1,1)*Cq1+Rc(2,1)*Sq1/Rc(3,1))
□
 q3= DATAN(Sq2*( Rc(1,1)*Cq1+Rc(2,1)*Sq1)+Rc(3,1)*Cq2)/
+(-Rc(1,1)*Sq1+Rc(2,1)*Cq1))
□
D = DSQRT( Rc(1,1)**2+Rc(2,1)**2+(Rc(3,1)-Hp)**2 )-Lp
□
C*************************************************************
C
     pi  = 3.1415926D0
     if( (Rc(1,1).eq.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=+pi
```

```
      if( (Rc(1,1).gt.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=+pi+q1
      if( (Rc(1,1).lt.0.0d0).and.(Rc(2,1).gt.0.0d0) ) q1=-pi+q1
        x1=-DSIN(q1)*Rc(1,1)+DCOS(q1)*Rc(2,1)
        y1= Rc(3,1)-Hp
        z1= DCOS(q1)*Rc(1,1)+DSIN(q1)*Rc(2,1)
      if( (y1.eq.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi
      if( (y1.lt.0.0d0).and.(x1.gt.0.0d0) ) q2=+pi+q2
      if( (y1.gt.0.0d0).and.(x1.gt.0.0d0) ) q2=-pi+q2
           q1gr= 180.0d0*q1/pi
           q2gr= 180.0d0*q2/pi
C***************************************************************
C
         T0(1,2,1)= +1.0D0
         T0(1,4,1)= -82.0D0/5.0D0*Sgam
         T0(2,1,1)= -Cbet
         T0(2,3,1)= -Sbet
         T0(2,4,1)= -30.0D0*Cbet-7.0D0/10.0D0*Sbet-
     +                82.0D0/5.0D0*Sbet*Cgam
         T0(3,1,1)= -Sbet
         T0(3,3,1)= +Cbet
         T0(3,4,1)= +7.0D0/10.0D0*Cbet+
     +                82.0D0/5.0D0*Cbet*Cgam-30.0D0*Sbet
         T0(4,4,1)= +1.0D0
C
      Sq1= dsin( q1 )
      Cq1= dcos( q1 )
      Sq2= dsin( q2 )
      Cq2= dcos( q2 )
C
      A(1,1,1)=-Sq1
      A(1,3,1)= Cq1
      A(2,1,1)= Cq1
```

```
      A(2,3,1)= Sq1
      A(3,2,1)= 1.0D0
      A(3,4,1)= Hp
      A(4,4,1)=+1.0D0
C
      A(1,1,2)=-Sq2
      A(1,3,2)=-Cq2
      A(2,1,2)= Cq2
      A(2,3,2)=-Sq2
      A(3,2,2)= 1.0D0
      A(4,4,2)=+1.0D0
C
      call MULT (T0,1,A,1,T,1 )
      call MULT ( T,1,A,2,T,2 )
      call MULT ( T,2,A,3,T,3 )
      call MULT ( T,2,A,4,T,4 )
C*************************************************************
C
df2q1=((Cq3(d2*(Cq2)**2+d3*(Sq2*Sq3)**2*(cq1*Vc(1)+Sq1*Vc(2))+
+ d2*Sq2*(sq3)**2*(sq1*Vc(1)-Cq1*Vc(2)+Sq2*Cq2*Sq3(d3(Sq3)**2-
d2)*
+ Vc(3))/(D*(d1*Sq3**2+d2*((Cq2*Cq3)**2+d3*((Sq2*Sq3)**2)-
+ df1q1)/hag
 df1q1=
(Cq3(d2*(Cq2)**2+d3*(Sq2*Sq3)**2*(cq1*Vc(1)+Sq1*Vc(2))+
+ d2*Sq2*(sq3)**2*(sq1*Vc(1)-Cq1*Vc(2)+Sq2*Cq2*Sq3(d3(Sq3)**2-
d2)*
+ Vc(3))/(D*(d1*Sq3**2+d2*((Cq2*Cq3)**2+d3*((Sq2*Sq3)**2)
 df1q2=(-Cq1*Cq2*Vc(1,1)-Sq1*Cq2*Vc(2,1)+Sq2*Vc(3,1)+df1q1*
+Cq2*Cq3*D)/(D*Sq3)
 df1q3=(Vc(1,1)*(-Sq1*Sq3-Cq1*Sq2*Cq3-Cq2*Cq3*Vc(3,1)+df1q1*D*
+Sq2+(Cq1*Sq3-Sq1*Sq2*Cq3)*Vc(2,1))/D
```

```
 df1D =   Sq1*Cq2*Vc(1,1)-Cq1*Cq2*Vc(2,1)-Sq2*Vc(3,1)
        DF1T0(1,4,1)= -82.0D0/5.0D0*Cgam*df1gam
        DF1T0(2,1,1)= +Sbet*df1bet
        DF1T0(2,3,1)= -Cbet*df1bet
        DF1T0(2,4,1)= -82.0D0/5.0D0*Cgam*Cbet*df1bet+
     +                 82.0D0/5.0D0*Sbet*Sgam*df1gam+
     +                 30.0D0*Sbet*df1bet-
     +                 7.0D0/10.0D0*Cbet*df1bet
        DF1T0(3,1,1)= -Cbet*df1bet
        DF1T0(3,3,1)= -Sbet*df1bet
        DF1T0(3,4,1)= -82.0D0/5.0D0*Cgam*Sbet*df1bet-
     +                 7.0D0/10.0D0*Sbet*df1bet-
     +                 82.0D0/5.0D0*Cbet*Sgam*df1gam-
     +                 30.0D0*Cbet*df1bet
C
     call MULT ( T0,1,TETA,1,Matr1,1 )
     call MULT ( Matr1,1,A,1,B,1 )
     call MULT ( DF1T0,1,A,1,Matr1,1 )
     call MULTS ( B,1,df1q1,Matr2,1 )
     call ZUSAM ( Matr1,1,Matr2,1,DF1T,1,+1 )
C
     call MULT ( T,1,TETA,2,Matr1,1 )
     call MULT ( Matr1,1,A,2,B,2 )
     call MULT ( DF1T,1,A,2,Matr1,1 )
     call MULTS ( B,2,df1q2,Matr2,1 )
     call ZUSAM ( Matr1,1,Matr2,1,DF1T,2,+1 )
C
     call MULT ( T,3,TETA,2,Matr1,1 )
     call MULT ( Matr1,1,A,3,B,2 )
     call MULT ( DF1T,1,A,3,Matr1,1 )
     call MULTS ( B,2,df1q3,Matr2,1 )
     call ZUSAM ( Matr1,1,Matr2,1,DF1T,3,+1 )
```

```
C*************************************************************
C
 df2q2= (-Cq1*cq2*Ac(1)-
Sq1*Cq2*Ac(2)+sq2*Ac(3)+Cq2*Cq3*D*df2q1+
+ Cq2*sq2*Sq3*D*df1q1**2+2.0d0*Cq2*q3*df1D*df1q1-
2.0d0*Cq2*Sq3*
+df1q1*df1q3*D-2.0d0Sq3df1D*df1q2-
2.0d0*Cq3*df1q2*df1q3*D)/(D*Sq3)
C
 df2q3=((-Sq1*Sq3-Cq1*Sq2*Cq3)*Ac(1)+(Cq1*Sq3-
Sq1*Sq2*Cq3)*Ac(2)-
+Cq2*Cq3*Ac(3)+sq2*d*df2q1-q2**2*cq3*sq3*D*df1q1**2+2.0d0Sq2*
+df1D*df1q1+2.0d0*Cq2*Sq3**2*df1q2*df1q1*D-2.0d0*df1D*df1q3+
+2.0d0*df1q3*sq3*D*df1q2*2)/D
C*************************************************************
C
        DF2T0(1,4,1)= -82.0D0/5.0D0*Cgam*df2gam+
     +                 82.0D0/5.0D0*Sgam*df1gam**2
        DF2T0(2,1,1)= +Sbet*df2bet+Cbet*df1bet**2
        DF2T0(2,3,1)= +Sbet*df1bet**2-Cbet*df2bet
        DF2T0(2,4,1)= +82.0D0/5.0D0*Cgam*Sbet*df1gam**2+
     +                 82.0D0/5.0D0*Cgam*Sbet*df1bet**2-
     +                 82.0D0/5.0D0*Cgam*Cbet*df2bet+
     +                 82.0D0/5.0D0*Sbet*Sgam*df2gam+
     +                 7.0D0/10.0D0*Sbet*df1bet**2+
     +                 30.0D0*Sbet*df2bet+
     +                 164.0D0/5.0D0*Cbet*Sgam*df1gam*df1bet+
     +                 30.0D0*Cbet*df1bet**2-
     +                 7.0D0/10.0D0*Cbet*df2bet
        DF2T0(3,1,1)= +Sbet*df1bet**2-Cbet*df2bet
        DF2T0(3,3,1)= -Sbet*df2bet-Cbet*df1bet**2
        DF2T0(3,4,1)= -82.0D0/5.0D0*Cgam*Sbet*df2bet-
```

```
     +                 82.0D0/5.0D0*Cgam*Cbet*df1gam**2-
     +                 82.0D0/5.0D0*Cgam*Cbet*df1bet**2+
     +                 164.0D0/5.0D0*Sbet*Sgam*df1gam*df1bet+
     +                 30.0D0*Sbet*df1bet**2-
     +                 7.0D0/10.0D0*Sbet*df2bet-
     +                 82.0D0/5.0D0*Cbet*Sgam*df2gam-
     +                 7.0D0/10.0D0*Cbet*df1bet**2-
     +                 30.0D0*Cbet*df2bet
C
      call MULT ( DF2T0,1,A,1,Matr1,1 )
      call MULTS ( B,1,df2q1,Matr2,1 )
          call MULT ( DF1T0,1,TETA,1,Matr3,1 )
          call MULT ( Matr3,1,A,1,Matr4,1 )
          call MULTS ( Matr4,1, 2.0D0*df1q1 ,Matr3,1 )
            call MULT ( T0,1,TETA,1,Matr4,1 )
            call MULT ( Matr4,1,TETA,1,Matr5,1 )
            call MULT ( Matr5,1,A,1,Matr4,1 )
            call MULTS ( Matr4,1, df1q1**2 ,Matr5,1 )
            call ZUSAM ( Matr3,1,Matr5,1,C,1,+1 )
      call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
      call ZUSAM ( Matr3,1,C,1,DF2T,1,+1 )
C
      call MULT ( DF2T,1,A,2,Matr1,1 )
      call MULTS ( B,2,df2q2,Matr2,1 )
          call MULT ( DF1T,1,TETA,2,Matr3,1 )
          call MULT ( Matr3,1,A,2,Matr4,1 )
          call MULTS ( Matr4,1, 2.0D0*df1q2 ,Matr3,1 )
            call MULT ( T,1,TETA,2,Matr4,1 )
            call MULT ( Matr4,1,TETA,2,Matr5,1 )
            call MULT ( Matr5,1,A,2,Matr4,1 )
            call MULTS ( Matr4,1, df1q2**2 ,Matr5,1 )
            call ZUSAM ( Matr3,1,Matr5,1,C,2,+1 )
```

```
      call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
      call ZUSAM ( Matr3,1,C,2,DF2T,2,+1 )
C
      call MULT ( DF2T,1,A,3,Matr1,1 )
      call MULTS ( B,2,df2q3,Matr2,1 )
          call MULT ( DF1T,1,TETA,2,Matr3,1 )
          call MULT ( Matr3,1,A,2,Matr4,1 )
          call MULTS ( Matr4,1, 2.0D0*df1q2 ,Matr3,1 )
            call MULT ( T,1,TETA,2,Matr4,1 )
            call MULT ( Matr4,1,TETA,2,Matr5,1 )
            call MULT ( Matr5,1,A,2,Matr4,1 )
            call MULTS ( Matr4,1, df1q3**2 ,Matr5,1 )
            call ZUSAM ( Matr3,1,Matr5,1,C,2,+1 )
      call ZUSAM ( Matr1,1,Matr2,1,Matr3,1,+1 )
      call ZUSAM ( Matr3,1,C,2,DF2T,2,+1 )
C*************************************************************
C
      call MULT ( DF2T,2,H,3,Matr1,1 )
      call ZUSAM ( F,3,Matr1,1,L,3,-1 )
C
      call TRNSP ( A,2,Matr1,1 )
      call MULT ( L,2,Matr1,1,Matr2,1 )
          call MULT ( DF2T,1,H,1,Matr1,1 )
              call ZUSAM ( Matr2,1,F,1,Matr3,1,+1 )
           call ZUSAM ( Matr3,1,Matr1,1,L,1,-1 )
C*************************************************************
C
      call TRNSP ( B,1,Matr1,1 )
      call MULT ( L,1,Matr1,1,Matr2,1 )
      call TRACE ( Matr2,1,Parm1 )
      MT(1)= - Parm1
C
```

```
      call TRNSP ( B,2,Matr1,1 )
      call MULT ( L,2,Matr1,1,Matr2,1 )
      call TRACE ( Matr2,1,Parm2 )
      MT(2)= - Parm2
C
      call TRNSP ( B,3,Matr1,1 )
      call MULT ( L,3,Matr1,1,Matr2,1 )
      call TRACE ( Matr2,1,Parm2 )
      MT(2)= - Parm3
C*************************************************************
      if( df1q1.EQ.0.0d0 ) sign= 0.0d0
      if( df1q1.NE.0.0d0 ) sign= df1q1/DABS( df1q1 )
      Md(1)= MT(1) + sign*Mtr(1)
      if( df1q2.EQ.0.0d0 ) sign= 0.0d0
      if( df1q2.NE.0.0d0 ) sign= df1q2/DABS( df1q2 )
      Md(2)= MT(2) + sign*Mtr(2)
      if( df1q3.EQ.0.0d0 ) sign= 0.0d0
      if( df1q3.NE.0.0d0 ) sign= df1q3/DABS( df1q3 )
      Md(3)= MT(3) + sign*Mtr(3)
C*************************************************************
      P2(1)= Md(1) * df1q1
      P2(2)= Md(2) * df1q2
      P2(3)= Md(3) * df1q3
C*************************************************************
      PR(1)= Md(1) * df2q1
      PR(2)= Md(2) * df2q2
      PR(3)= Md(3) * df2q3
C*************************************************************
C         Output results
C***************************************************************
*****
         ex(1,k)= q1
```

```
      ex(2,k)= q2
      ex(3,k)= q3
      ex(4,k)= df1q1
      ex(5,k)= df1q2
      ex(6,k)= df1q3
      ex(7,k)= df2q1
      ex(8,k)= df2q2
      ex(9,k)= df2q3
      ex(10,k)= time
      write(1,1001) ex(1,k),ex(2,k),ex(3,k)
      write(2,1001) ex(1,k),ex(4,k),ex(5,k)
      write(3,1001) ex(1,k),ex(6,k),ex(7,k)
666    Continue
1001  format(1x,3e12.4)
      close(1)
      close(2)
      close(3)
      stop
      END
    Subroutine Movel (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
    Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
   + Vmrc,hag,Arazg,PAR1
    Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
      hag= 0.01
      VMRC= 5.0d0
      Arazg= 500.0d0
      Acpro(1)= Arazg*par1
      Acpro(2)= 0.0d0
      Acpro(3)= 0.0d0
      Acpro(4)= 0.0d0
```

```
        Vcpro(1)= Vcprom(1)+Acpro(1)*hag
        Vcpro(2)= VMRC
        Vcpro(3)= 0.0d0
        Vcpro(4)= 0.0d0
        Rcpro(1)= Rcprom(1)+Vcpro(1)*hag
        Rcpro(2)= Rcprom(2)+Vmrc*hag
        Rcpro(3)= Rcprom(3)
        Rcpro(4)= 1.0d0
        RETURN
        END
      Subroutine Move2 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
      Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
     + Vmrc,hag,vhor,PAR1
      Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
        hag= 0.01
        VMRC= 5.0d0
        vhor= 50.0d0
        Acpro(1)= 0.0d0
        Acpro(2)= 0.0d0
        Acpro(3)= 0.0d0
        Acpro(4)= 0.0d0
        Vcpro(1)= Vhor*par1
        Vcpro(2)= VMRC
        Vcpro(3)= 0.0d0
        Vcpro(4)= 0.0d0
        Rcpro(1)= Rcprom(1)+Vhor*hag*par1
        Rcpro(2)= Rcprom(2)+Vmrc*hag
        Rcpro(3)= Rcprom(3)
        Rcpro(4)= 1.0d0
        RETURN
```

```
      END
    Subroutine Move3 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
    Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
   + Vmrc,hag,Atorm,PAR1
    Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
      hag= 0.01
      VMRC= 5.0d0
      Atorm= 500.0d0
      Acpro(1)=-Atorm
      Acpro(2)= 0.0d0
      Acpro(3)= 0.0d0
      Acpro(4)= 0.0d0
      Vcpro(1)= Vcprom(1)-Atorm*hag*par1
      Vcpro(2)= VMRC
      Vcpro(3)= 0.0d0
      Vcpro(4)= 0.0d0
      Rcpro(1)= Rcprom(1)+Vcpro(1)*hag
      Rcpro(2)= Rcprom(2)+Vmrc*hag
      Rcpro(3)= Rcprom(3)
      Rcpro(4)= 1.0d0
      RETURN
      END
    Subroutine Move4 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
    Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
   + Vmrc,hag,Arazg,PAR1
    Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
      hag= 0.01
      VMRC= 5.0d0
```

```
          Arazg= 1200.0d0
          Acpro(1)= 0.0d0
          Acpro(2)= 0.0d0
          Acpro(3)= Arazg*parl
          Acpro(4)= 0.0d0
          Vcpro(1)= 0.0d0
          Vcpro(2)= VMRC
          Vcpro(3)= Vcprom(3)+Arazg*hag*parl
          Vcpro(4)= 0.0d0
          Rcpro(1)= Rcprom(1)
          Rcpro(2)= Rcprom(2)+Vmrc*hag
          Rcpro(3)= Rcprom(3)+Vcpro(3)*hag
          Rcpro(4)= 1.0d0
          RETURN
          END
        Subroutine Move5 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
        Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
       + Vmrc,hag,vver,PAR1
        Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
          hag= 0.01
          VMRC= 5.0d0
          vver= 120.0d0
          Acpro(1)= 0.0d0
          Acpro(2)= 0.0d0
          Acpro(3)= 0.0d0
          Acpro(4)= 0.0d0
          Vcpro(1)= 0.0d0
          Vcpro(2)= VMRC
          Vcpro(3)= Vver*parl
          Vcpro(4)= 0.0d0
```

```
        Rcpro(1)= Rcprom(1)
        Rcpro(2)= Rcprom(2)+Vmrc*hag
        Rcpro(3)= Rcprom(3)+vver*hag*par1
        Rcpro(4)= 1.0d0
        RETURN
        END
      Subroutine Move6 (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro,PAR1)
      Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
     + Vmrc,hag,Atorm,PAR1
      Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
        hag= 0.01
        VMRC= 5.0d0
        Atorm= 1200.0d0
        Acpro(1)= 0.0d0
        Acpro(2)= 0.0d0
        Acpro(3)= -Atorm*par1
        Acpro(4)= 0.0d0
        Vcpro(1)= 0.0d0
        Vcpro(2)= VMRC
        Vcpro(3)= Vcprom(3)-Atorm*hag*par1
        Vcpro(4)= 0.0d0
        Rcpro(1)= Rcprom(1)
        Rcpro(2)= Rcprom(2)+Vmrc*hag
        Rcpro(3)= Rcprom(3)+Vcpro(3)*hag
        Rcpro(4)= 1.0d0
        RETURN
        END
      Subroutine Stop (Rcprom,Vcprom,Acprom,
Rcpro,Vcpro,Acpro)
      Double Precision Rcprom,Vcprom,Acprom,Rcpro,Vcpro,Acpro,
```

```
     + Vmrc,hag
      Dimension
Rcprom(4),Vcprom(4),Acprom(4),Rcpro(4),Vcpro(4),Acpro(4)
        hag= 0.01
        VMRC= 5.0d0
        Acpro(1)= 0.0d0
        Acpro(2)= 0.0d0
        Acpro(3)= 0.0d0
        Acpro(4)= 0.0d0
        Vcpro(1)= 0.0d0
        Vcpro(2)= VMRC
        Vcpro(3)= 0.0d0
        Vcpro(4)= 0.0d0
        Rcpro(1)= Rcprom(1)
        Rcpro(2)= Rcprom(2)+Vmrc*hag
        Rcpro(3)= Rcprom(3)
        Rcpro(4)= 1.0d0
        RETURN
        END
        SUBROUTINE MULT ( A,B,C)
        INTEGER          L,M,N
        DOUBLE PRECISION A,B,C,SUM
        DIMENSION  A(4,4),B(4,4),C(4,4)
        DO 3 L=1,4
             DO 2 M=1,4
                  SUM=0.0D0
                  DO 1 N=1,4
                       SUM=SUM+A(L,N)*B(N,M)
1                 CONTINUE
                  C(L,M)=SUM
2            CONTINUE
3       CONTINUE
```

```
      RETURN
      END
      SUBROUTINE MULTS1 ( A,X,y )
      INTEGER          K,L
      DOUBLE PRECISION A,X,y
      DIMENSION  A(4,4),x(4),y(4)
      DO 2 K=1,4
          y(k)= 0.0d0
           DO 1 L=1,4
                Y(k)=A(K,L)*X(L)+Y(k)
1          CONTINUE
2     CONTINUE
      RETURN
      END
      SUBROUTINE TRACE ( A,X )
      INTEGER          J
      DOUBLE PRECISION A,X
      DIMENSION  A(4,4)
      X=0.0D0
      DO 1 J=1,4
           X=X+A(J,J)
1     CONTINUE
      RETURN
      END
      SUBROUTINE TRNSP ( A,B)
      INTEGER          K,L
      DOUBLE PRECISION A,B
      DIMENSION  A(4,4),B(4,4)
      DO 2 K=1,4
           DO 1 L=1,4
                B(L,K)=A(K,L)
                B(K,L)=A(L,K)
```

```
1             CONTINUE
2         CONTINUE
          RETURN
          END
          SUBROUTINE ZUSAM ( A,B,C,PAR )
          INTEGER          PAR,L,M
          DOUBLE PRECISION A,B,C
          DIMENSION  A(4,4),B(4,4),C(4,4)
          IF( PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
          STOP 'ZUSAM: ABEND#1'
1         DO 3 L=1,4
               DO 2 M=1,4
                    IF( PAR.EQ.+1 ) C(L,M)=A(L,M)+B(L,M)
                    IF( PAR.EQ.-1 ) C(L,M)=A(L,M)-B(L,M)
2              CONTINUE
3         CONTINUE
          RETURN
          END
          SUBROUTINE MULTS2 (A,B,C,PAR)
          INTEGER PAR,L
          DOUBLE PRECISION A,B,C
          DIMENSION A(4),B(4),C(4)
          IF (PAR.EQ.-1.OR.PAR.EQ.+1 ) GOTO 1
          STOP 'MULTS2: ABEND#1'
1         DO 3 L=1,4,1
          C(L)=A(L)+B(L)*PAR
3         CONTINUE
          RETURN
          END
          SUBROUTINE MULTS ( A,X,B)
          INTEGER          K,L
          DOUBLE PRECISION A,B,X
```

```
      DIMENSION  A(4,4),B(4,4)
      DO 2 K=1,4
           DO 1 L=1,4
                B(K,L)=A(K,L)*X
1          CONTINUE
2     CONTINUE
      RETURN
      END
```

APPENDIX D

PHOTOGRAPH OF SCANNING SYSTEM