

# Analytical Study of Algorithms for Solving Inverse Kinematic Problems in Robot Motion Control Systems

Dr. Osama Ahmad Salim Safarini  
Department of Computer Science  
The University College of Taymaa  
University of Tabuk,  
Tabuk 71491, Saudi Arabia

**Abstract**—The given article covers the general formulations of inverse kinematic problems for robot motion control systems. We have discussed the difficulties how to solve such problems using analytical and numerical methods. We have also analyzed the convergence of iterative algorithms with the regularization on the trajectory with the points outside of the gripper reachability. The example of an iterative calculation of joint trajectories for a 3-link robot using the recursive algorithm for the Jacobi matrix calculation has been presented.

**Keywords**—Robot motion control systems; inverse kinematic problems; iterative methods; algorithm convergence; regularization

## I. INTRODUCTION

When modeling manipulative robot motion control systems (MCS) it is necessary to solve with the help of a computer the inverse kinematic problems (IKP) for their executive mechanisms (EM) using analytical or iterative methods. Algorithms for solving such problems constitute the mathematical basis for development (MCS) of robots.

A robot manipulator is composed of a serial chain of rigid links connected to each other by revolute or prismatic joints. A revolute joint rotates about a motion axis and a prismatic joint slide along a motion axis. Each robot joint location is usually defined relative to neighboring joint. The relation between successive joints is described by  $4 \times 4$  homogeneous transformation matrices that have orientation and position data of robots. The number of those transformation matrices determines the degrees of freedom of robots. The product of these transformation matrices produces final orientation and position data of a  $n$  degrees of freedom robot manipulator. Robot control actions are executed in the joint coordinates while robot motions are specified in the Cartesian coordinates. Conversion of the position and orientation of a robot manipulator end-effector from Cartesian space to joint space is called as inverse kinematics problem, which is of fundamental importance in calculating desired joint angles for robot manipulator design and control [1].

For a manipulator with  $n$  degree of freedom, at any instant of time joint variables is denoted by ( $q_i = q(t)$ ,  $i = 1; 2; 3; \dots; n$ ) and position variables ( $x_j = x(t)$ ,  $j = 1; 2; 3; \dots; m$ ). The relations between the end-effector position  $x(t)$  and joint angle  $q(t)$  can be represented by forward kinematic equation,  $x(t) = f(q(t))$

where  $f$  is a nonlinear, continuous and differentiable function. On the other hand, with the given desired end effector position, the problem of finding the values of the joint variables is inverse kinematics, which can be solved by,  $q(t) = f^{-1}(x(t))$ . Solution of  $q(t)$  is not unique due to nonlinear, uncertain and time varying nature of the governing equations [2].

The different techniques used for solving inverse kinematics can be reviewed with some articles where, Wu et.al. [3], a new analytic inverse kinematics (IK) solver is proposed which is suitable for multiple constrained 12-DOF human limbs. By decomposing human skeleton into five parts one head chain, two arm chains and two leg chains, a multi-constrained human skeleton can be solved analytically.

Drzevitzky [4] introduced Inverse Kinematics problems for anthropomorphic limbs and have shown how to solve those analytically in order to obtain symbolic solutions. The symbolic solutions can be modified and re-computed to match, for example, other input values that serve as constraints when solving the according Inverse Kinematics problem.

In the theoretical robotics solutions for IKP often use algorithms based on analytical expressions that require calculating inverse trigonometric and transcendental functions. Such algorithms are obtained directly from geometric kinematics models of EM, or by vector-matrix models in the representation of Denavita-Hartenberg, describing the kinematics of EM in homogeneous coordinates [7]-[13].

However, for robots with complex kinematics, analytical solutions of the IKP on a given trajectory of grasping may turn out to be erroneous in specific configurations of EM (link positions), as well as at the boundary and outside the reachable zone of the grasp (grripper) because of the degeneracy of the Jacobi matrix due to the lowering of its rank. In such cases, only approximate solutions of IKP can be obtained by iterative methods. But when using such methods, it is necessary to study the convergence of their algorithms.

The article discusses the difficulties of the analytical solution of the IKP using the example of a three-link robot with rotational links, draws attention to the need for regularization of iterative algorithms to ensure their convergence, describes the recursive algorithm for calculating

the Jacobi matrix, and analyzes the operability of iterative algorithms on the trajectories of the gripper containing areas with special configurations of EM.

## II. GENERAL FORMULATIONS OF THE IKP

In systems of positional, high-speed and force-torque control movement of robots, different inverse problems of kinematics are solved.

### A. IKP about Positions of Links

Given a  $6 \times 1$  vector of linear position coordinates and angular coordinates of the orientation of the gripper

$$S_C = (x_C, y_C, z_C, \varphi_C, \theta_C, \psi_C)^T$$

$N \times 1$  vector of generalized coordinates of links is calculated

$$q = (q_1, q_2, \dots, q_N)^T = \Phi^{-1}(S_C), \quad (1)$$

Where  $\Phi^{-1}$  is the  $N \times 1$  vector-valued function inverse to the  $6 \times 1$  vector-valued function  $\Phi(q)$  corresponding to the kinematic scheme of the robot's EM.

The problem (1) is the most complicated from the computational point of view, since it requires the solution of a system of nonlinear algebraic equations of the form.

$$\Phi(q) - S_C = F(q) = 0$$

Where, 0 is the zero  $6 \times 1$  vector.

### IKP about link speeds

For given vectors of linear and angular velocities of grasping (of the gripper):

$$V_C = (v_x, v_y, v_z)^T; \Omega_C = (\omega_\varphi, \omega_\theta, \omega_\psi)^T$$

the  $N \times 1$  vector of generalized link speeds is calculated.

$$\dot{q} = (\dot{q}_1, \dot{q}_2, \dots, \dot{q}_N)^T = J^{-1}(q) \cdot \begin{pmatrix} V_C \\ \Omega_C \end{pmatrix} \quad (2)$$

Where  $J^{-1}(\cdot)$  is the inverse (or pseudoinverse) matrix of the Jacobi matrix  $J(q)$  of the vector-valued function  $\Phi(q)$ .

The problem (2) is a solution of a system of linear algebraic equations of the form:

$$J(q) \cdot \dot{q} = \begin{pmatrix} V_C \\ \Omega_C \end{pmatrix}$$

### B. IKP about the Forces and Moments in the Joints (Hinges) of the Links

By the given vectors of projection of force and moment in the gripper:

$$F_C = (F_x, F_y, F_z)^T; M_C = (M_\varphi, M_\theta, M_\psi)^T$$

The vector of generalized forces in the hinges is determined

$$Q = (Q_1, Q_2, \dots, Q_N)^T = J^T(q) \cdot \begin{pmatrix} F_C \\ M_C \end{pmatrix}. \quad (3)$$

Expression (3), which requires the calculation and transposition of the Jacobi matrix, is valid only for the case of an ideal EM that does not have energy losses in the joints of the links.

The Jacobi matrix in problems (1) - (3), depending on the number of links  $N$  of a robot can be square or have a rectangular form:

$$J(q) = \frac{\partial \Phi(q)}{\partial q} = \frac{\partial F(q)}{\partial q} = \left\{ \frac{\partial F_i(q)}{\partial q_j}, i = \overline{1,6}, j = \overline{1,N} \right\}$$

In a more general formulation, problem (1) can be formulated as the problem of minimizing the square of the norm of the discrepancy vector:

$$\|F(q)\|^2 \rightarrow \min_{q \in R^N}, \quad (4)$$

and solved by iterative methods without computing the Jacobi matrix or gradient methods using its numerical approximations. However, the convergence of algorithms of such methods is slower and when they are used, more steps of the iterative process are required [6].

In [8], an example of an iterative solution of the IKP in the formulation (4) for a 6-link robot is given, on the program trajectory of grasping which the constant orientation of the grasping is given not by the Euler angles, but by the vector of the direction cosines.

If the values of the vector of generalized coordinates are bounded by the admissible domain  $q \in D^N$ , then problems of the form (4) should be solved by methods of conditional minimization. In this case, the exact solution of the IKP may in principle be absent.

### C. Analytical Solution of the IKP

Consider a 3-link robot of the BBB type operating in an angular coordinate system for which problem (1) is solved ambiguously and depends on the sign of the angular position of the third link (the lower or upper arm configuration):

$$\begin{aligned} q_1 &= \text{arctg}(y_C/x_C); \\ q_3 &= \pm \arccos(C); \\ q_2 &= \text{arctg} \left( \frac{(z_C - L_1)}{R_{xy}} \right) \pm \arccos(D), \end{aligned} \quad (5a)$$

Where  $L_1, L_2, L_3$  are the lengths of the links;

$$\begin{aligned} R_{xyz} &= \sqrt{x_C^2 + y_C^2 + (z_C - L_1)^2}; \\ C &= (x_C^2 + y_C^2 + (z_C - L_1)^2 - L_2^2 - L_3^2) / (2L_2L_3); \\ D &= (x_C^2 + y_C^2 + (z_C - L_1)^2 + L_2^2 - L_3^2) / (2L_2R_{xyz}); \end{aligned}$$

The solution of the IKP can also be obtained from other expressions [9]:

$$\begin{aligned} q_1 &= \arctg \left( \frac{y_c}{x_c} \right); \quad E = \sqrt{1 - C^2}; \\ q_3 &= \pm \arctg (E/C); \\ q_2 &= \arctg \left( \frac{(z_c - L_1)/R_{xy}}{\left( \frac{L_3 \sin q_3}{L_2 + L_3 \cos q_3} \right)} \right) - \arctg \left( \frac{L_3 \sin q_3}{L_2 + L_3 \cos q_3} \right) \end{aligned} \quad (5b)$$

However, the solutions of the IKP with MATLAB functions by the expressions (5a) on the trajectory of grasping with points outside its reachability zone turn out to be complex numbers that have no practical meaning. To obtain real solutions of the IKP, the values of the variables C and D, which are cosines, should be limited to one. The solutions of the IKP by the expressions (5b) are obtained real at points both inside and outside the reach zone of the gripper, so there is no need to limit the values of the variables C and E.

### III. ALGORITHMS OF ITERATIVE METHODS

We consider algorithms for the numerical solution of problem (1), in which the Jacobi matrix is calculated in explicit form.

#### A. Algorithm of Newton's Method

Theoretically, in the case of a square Jacobi matrix, for example, when  $N = 3$  or  $6$ , the following algorithm can be used:

$$q^{k+1} = q^k - a_k J^{-1}(q^k) \cdot F(q^k), \quad (6)$$

Where  $a_k \leq 1$  is the scalar value of a constant or variable step.

Wherein At each step of the iterative process, it is required to calculate the inverse matrix corresponding to the Jacobi matrix. However, in special configurations of EM, if the Jacobi matrix is poorly conditioned or completely degenerate due to a decrease in its rank, the operability of algorithm (6) is lost,

#### Simplified algorithm of Newton's method

If we replace the inverse matrix  $J^{-1}$  in the algorithm (6) with the transposed Jacobi matrix

$$q^{k+1} = q^k - a_k J^T(q^k) \cdot F(q^k), \quad (7)$$

Coarse (Rough) solutions of problem (1) can be obtained, but in special configurations of EM, the convergence of algorithm (7) is also not guaranteed.

#### B. Algorithm of the Gauss-Newton Method

When solving problems with a rectangular Jacobi matrix, when the number of links is redundant ( $N > 6$ ) or insufficient ( $N < 6$ ) in the algorithm (6), instead of the inverse matrix  $J^{-1}$ , it is necessary to use the left

$J^+ = (J^T \cdot J)^{-1} J^T$  or the right  $J^+ = J^T (J \cdot J^T)^{-1}$  pseudoinverse matrices [5]. We obtain the following algorithm:

$$q^{k+1} = q^k - a_k J^+(q^k) \cdot F(q^k) \quad (8)$$

#### C. Algorithms of the Levenberg-Marquardt Method

In the algorithm (8), in calculating pseudo inverse matrices, because the matrices  $(J^T \cdot J)$  and  $(J \cdot J^T)$  can turn out to be poorly conditioned or degenerate, to ensure the convergence of the iterative processes, it is necessary to carry out factorization based on the matrix decomposition or to use simple regularization:

$$q^{k+1} = q^k - a_k [(J^T J + a_{2k} E)^{-1} J^T] \cdot F(q^k) \quad (9)$$

$$q^{k+1} = q^k - a_k [J^T (J J^T + a_{2k} E)^{-1}] \cdot F(q^k) \quad (10)$$

Where  $E$  is the identity matrix;  $a_{2k}$  is a regularizing scalar parameter.

When applying any of the algorithms (6) - (10), the matrix  $J(q)$  can be calculated either by the direct derivation of the vector function  $\Phi(q)$  by analytic expressions, which is rather cumbersome for  $N \geq 3$ , or by a more efficient recursive algorithm [13], using the transformation matrices of homogeneous coordinates and the intersection operations (vector products) of the columns of the rotation matrix of grasping ( $n, o, a$ ) - normal vectors, orientation and approach.

#### A recursive algorithm for computing the Jacobi matrix

In the base coordinate system of the robot, the Jacobi matrix performs the transformation of the vector of generalized link speeds:

$$\begin{pmatrix} V_{C,0} \\ \Omega_{c,0} \end{pmatrix} = J_0(q) \cdot \dot{q}$$

Where  $V_{C,0}$  and  $\Omega_{c,0}$  are the vectors of the projections of the linear and angular velocity of grasping.

Here the Jacobi matrix is defined as the product

$$J_0(q) = J_V(q) \cdot J_N(q)$$

Where  $J_N(q) - 6 \times N$  Jacobi matrix of velocity transformation in the coordinate system of the gripper

$$\begin{pmatrix} V_{C,N} \\ \Omega_{C,N} \end{pmatrix} = J_N(q) \cdot \dot{q}$$

$J_V(q) - 6 \times 6$  matrix transformation of linear and angular velocities of grasping from the  $N$ -th to the base coordinate system.

$$\begin{pmatrix} V_{C,0} \\ \Omega_{C,0} \end{pmatrix} = J_V(q) \cdot \begin{pmatrix} V_{C,N} \\ \Omega_{C,N} \end{pmatrix}$$

To calculate the matrix  $J_N(q)$ ,  $4 \times 4$  matrices are required that determine the position and orientation of the gripper in the  $j$ -th coordinate system, i.e., with respect to the  $(j-1)$ -th link:

$$\begin{aligned} U_j(q) &= A_j(q_1) A_{j+1}(q_2) \dots A_N(q_N) = \\ &= \begin{pmatrix} R_{N,j} & P_{N,j} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \end{aligned}$$

Where  $R_{N,j} = (n_{N,j}, o_{N,j}, a_{N,j})^T$  - rotation matrix with orientation vectors of the grip coordinate system;  $P_{N,j} = (X_{N,j}, Y_{N,j}, Z_{N,j})^T$  vector of the position of the center of the coordinate system of the grasp.

Columns of the matrix  $J_N(q) = (J_{1,N} J_{2,N} \dots J_{6,N})$  are calculated by the reverse recursion ( $j = 6, 5, \dots, 1$ ):

$$J_{j,N} = \begin{cases} \begin{pmatrix} d_j \\ \delta_j \end{pmatrix} & \text{For } j \text{ link B-type} \\ \begin{pmatrix} \delta_j \\ 0_{3 \times 1} \end{pmatrix} & \text{For } j \text{ link p-type,} \end{cases}$$

Where  $d_j$  -  $3 \times 1$  vector, composed of Z-components of vector products.

$$d_j = \begin{bmatrix} (P_{N,j} \times n_{N,j})_\tau \\ (P_{N,j} \times o_{N,j})_\tau \\ (P_{N,j} \times a_{N,j})_\tau \end{bmatrix} = \begin{bmatrix} P_{x,j} n_{y,j} - P_{y,j} n_{x,j} \\ P_{x,j} o_{y,j} - P_{y,j} o_{x,j} \\ P_{x,j} a_{y,j} - P_{y,j} a_{x,j} \end{bmatrix};$$

$\delta_j$  - a  $3 \times 1$  unit vector directed along the axis of motion of the  $j$ -th link

$$\delta_j = R_{N,j}^T \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} n_{\tau,j} \\ o_{\tau,j} \\ a_{\tau,j} \end{bmatrix};$$

$0_{3 \times 1}$  is the zero  $3 \times 1$  vector.

The matrix  $J_V(q)$  in the block representation has the following form:

$$J_V(q) = \begin{bmatrix} R_{N,0} & 0_{3 \times 3} \\ 0_{3 \times 3} & R_{N,0} \end{bmatrix}$$

Where  $0_{3 \times 3}$  is the zero matrix;  $R_{N,0}(q)$  -  $3 \times 3$  rotation matrix with the column vectors of the normal, orientation and approach, calculated in the base coordinate system.

A more detailed recursive algorithm for computing the Jacobi matrix is described in [9].

### Analysis of the convergence of iterative algorithms

Let us analyze the processes of the iterative IKP solution using algorithms (6) - (10) with the Jacobi matrix calculation using the above recursive algorithm on two BBB type robot gripper trajectories (Fig. 1).

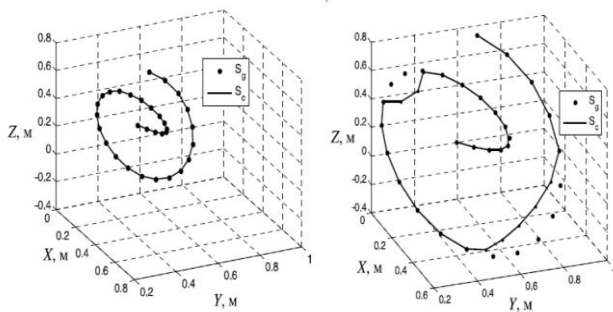


Fig. 1. Given trajectories of grasp.

The trajectories are given by the base points  $S_g = \{X, Y, Z\}$ , the first is in the reachable zone of grasp, the second - with some points outside its boundary.

Fig. 2 and 3 show the trajectory of the links obtained for the upper configuration of the robot by analytical methods (5a) and (5b), and calculated by these trajectories by solving the direct problem of kinematics corresponding trajectories of the gripper  $S_c = \{X_c, Y_c, Z_c\}$  are shown in Fig. 1 by solid lines.

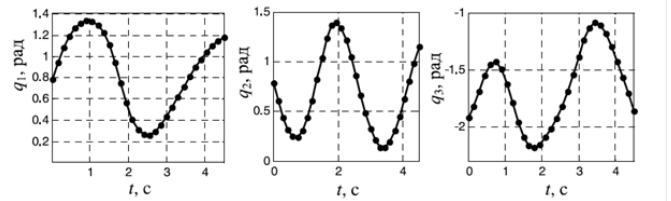


Fig. 2. Results of the solution of the IKP on the first trajectory of grasp.

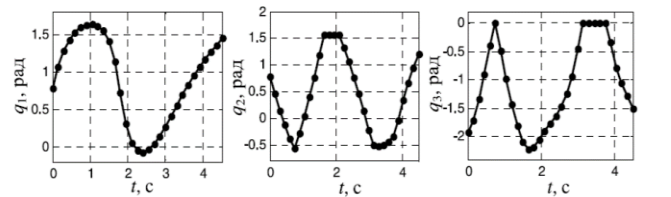


Fig. 3. Results of the solution of the IKP on the second trajectory of grasp.

In Fig. 4 are presented the graphs of the modules of the Jacobi matrix determinant of the BBB-type robot, calculated from the recursive algorithm for the two variants of the link trajectories (see Fig. 2 and 3).

Exactly the same graphs are obtained when calculating the elements of the Jacobi matrix by analytic expressions. It is seen from the graphs that for the second variant of the trajectories at the points on the boundary of the reachability zone of the grasp the determinant of the degenerate Jacobi matrix takes the value zero.

On the first trajectory of the grasp, the solutions obtained by all the compared algorithms (6) - (10) coincide with the trajectories of the links (see Fig. 2), calculated by analytical methods. At the same time, for each of the algorithms, the values of the constant parameters  $a_k$  and  $a_{2k}$  were established, which ensure the convergence of the IKP solution processes for the number of iterations  $k_{max} \leq 100$ .

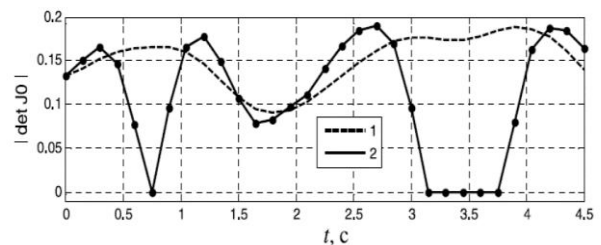


Fig. 4. Graphs of modules of the Jacobi matrix determinant.

#### REFERENCES

On the second trajectory of grasping, because of the degeneracy of the Jacobi matrix at special points, the convergence of algorithms (6) and (8) is lost. Algorithms (9) and (10), with regularized left and right pseudo-inverse matrices, are the most stable. At values of  $a_k = 2$ ,  $a_{2k} = 0,1$  they give the same solutions as in Fig. 3, for the number of iterations  $k_{max} \leq 30$ . The algorithm (7), which uses the transposed Jacobi matrix, converges, but gives very rough solutions.

#### IV. CONCLUSION

Thus, based on the results of the study, the following practical recommendations can be made.

When modeling and developing control systems for the movement (motion) of manipulative robots, it is advisable to use iterative algorithms for solving the IKP. Analytical solutions by expressions containing inverse trigonometric functions in special configurations of EM can turn out to be incorrect.

The most accurate solutions of the IKP multi-link robots can be obtained by algorithms (9) and (10) with regularization, in which the Jacobi matrix is recommended to be calculated without numerical differentiation using a recursive algorithm that uses kinematic models in homogeneous coordinates.

When programming (planning) the grasp paths (trajectories) in robot motion control systems, care should be taken to avoid the situations shown in Fig. 1 and 3.

The base points should be set in the working area of the robot, determined by the overall dimensions of the links and the Permissible variation ranges of the generalized coordinates.

- [1] P. Baerlocher, "Inverse Kinematic Techniques of the Interactive Posture Control of Articulated Figures". Thesis2001.
- [2] S Alavandar, M.J. Nigam. "Inverse Kinematics Solution of 3DOF Planar Robot using ANFIS", Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844. Vol. III (2008), Suppl. issue: Proceedings of ICCCC 2008, pp. 150-155.
- [3] X. Wu, L. Ma, Z. Chen, Y. Gao, "A 12-DOF Analytic Inverse Kinematics Solver for Human Motion Control ", Journal of Information & Computational Science, PP 137-141, 30-August 2004.
- [4] S. Drzevitzky, "Symbolic Solutions for Inverse Kinematics Problems for Anthropomorphic Limbs", Diplomarbeit Thesis, Germany, March, 2008.
- [5] BEKLEMISHEV D.V. DOPOLNITELNYE GLAVY LINEYNOY ALGEBRY. MOSCOW: NAUKA PUBL., 1983, 336 P. (RUS).
- [6] Dennis D., Shnabel R. Chislennyye metody bezuslovnoy optimizatsii i resheniya nelineynykh uravneniy. Moscow: Mir Publ., 1988, 440 p. (rus)
- [7] ZENKEVICH S.L., YUSHCHENKO A.S. UPRAVLENIYE ROBOTAMI. OSNOVY UPRAVLENIYA MANIPULYATSIONNYMI ROBOTAMI. MOSCOW: MGTU IM. N.E. BAUMANA PUBL., 2000, 400 P. (RUS).
- [8] Ignatova Ye.I., Rostov N.V. Kinematicheskiye Zadachi v Matematicheskom Modelirovanii Robotov, Nauchno-tekhnicheskiye Vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravleniye. St. Petersburg: SPbGPU Publ, 2010, No. 2(97), Pp. 159–164. (RUS).
- [9] Ignatova Ye.I., Lopota A.V., Rostov N.V. Sistemy Upravleniya Dvizheniyem Robotov. Kompyuternoye Proyektirovaniye. St. Petersburg: SPbGPU Publ, 2014, 302 p. (RUS).
- [10] Shakhinpur M. Kurs Robototekhniki, Per. S Angl. Pod Red. S.L. Zenkevicha. Moscow: Mir Publ., 1990, 527 p. (RUS).
- [11] Fu K., Gonsales R., Li K. Robototekhnika, Per. S Angl. Pod Red. V.G. Gradetskogo. Moscow: Mir Publ., 1989, 624 p. (RUS).
- [12] Yurevich Ye.I. Osnovy Robototekhniki. St. Petersburg: BKhV-Petersburg Publ., 2005, 416 p. (RUS).
- [13] Corke P.I. Robotics Toolbox 9.9 for MATLAB R9. 2014, 286 p.