MODELLING AND DESIGN OF ROBOTIC WORKSTATIONS FOR CONTACT TASKS

KRASSIMIR E. GEORGIEV
Bulgarian Academy of Sciences – Institute of Mechanics, 1113 Sofia
E-mail: kris@imbm.bas.bg

ABSTRACT. This article deals with the design and modeling of robotic workstations (RWS) for contact tasks – assembly, grinding and polishing. We have shown that there are specific problems at the whole process of development of robotic workstations: micro and macromotion devices, easy reconfiguration of the system, ad-hoc integration of peripherals, rapidly adaptable manufacturing systems and etc.

KEY WORDS: robotic tasks, contact operations, CAD design

1. Introduction

The overall mission of computer simulation and virtual development is to build a fully functional virtual products and systems used for understanding of simulated behavior in order to achieve optimal system performance at minimum weight and cost. [1,2,3,4,5,6]. At physical description and modeling robotic system is represented by physical models, for example as a multibody system, containing rigid bodies, joints and coupling force elements. The mathematical description and modeling is a representation of a system by mathematical equations which can be derived from the physical model description, e.g. the equations of motion of a multibody system. The simulation results of the mathematical model description are considered as the behavioral model description - the trajectories of position and velocity of the bodies. Then coupling of robotic models in the behavioral model description is referred to as simulator coupling, modular modeling and simulation or virtual assembly of them. The simulation of the global system is realized by time discrete linker and scheduler which combines the inputs and outputs of the corresponding subsystems and establishes communication between the subsystems to discrete time instants. Therefore it is possible to use different software packages for each subsystem and then to link the solvers together. Now the general modeling and design may be presented in form, shown in Fig.1, with respective levels of task simulation and planning. The modular description of systems allows for independent and parallel modeling of the internal dynamics of each subsystem (system decomposition). The inputs and outputs of the physical model are also physical quantities such as forces or motion of the bodies. It is assumed that the conditions for
decoupling of movement are present and the parameters of the regional macro motion are defined by standards of joint positions, velocities and accelerations, which are limited functions of time. Sensors for position and velocity of joint displacements are present, assuming that the elasticity of links and the frictional forces in the joints are considered as zero.

**2. Mechatronical approach to systematic design of RWS**

The key idea of Mechatronics is a system-thinking approach to intelligent–machine design by synergistically combining mechanics, electronics and control during the development. The ultimate purpose is to achieve a system performance superior to what can be achieved by traditional development and design cycle. The author’s idea follows this approach and mechatronics principles to close the open kinematics chains, using control and information loops. Then it is possible to estimate different parameters, to compliance them and to achieve complex properties: Adaptivity, reconfigurable structure, energy efficiency and high performance of robotic workstations for contact tasks.

The goal is to achieve efficiency of the technological macro movements as well as high speed, productivity and energy efficiency. The results from this research are very important when designing complex methodology and creating new strategies for high performance and human scale intelligent mechatronic systems. A method
Modelling and design of robotic workstations for contact tasks

for kinematic and dynamic research of 4R assembly robots and adaptive assembly sockets with local dynamic compliance was suggested. The essence of the method is in appropriate applying of 3D modeling in Solid Works environment, combined with software for simulation of kinematic and dynamic processes: Matlab. The Robotics Toolbox module for Matlab contains many useful tools for research of robotic manipulators like: kinematics, dynamics and trajectory path planning. The module is useful for simulations and analysis of data, obtained from real robots. The software module Cosmos Motion 2005 for Solid Works simulates mechanic motorized mechanisms and the physical forces created by them. For creation of the mathematical model of an assembly 4R manipulator, it is necessary first to determine its kinematic structure. Then coordinate frames are assigned to each link according to Denavit – Hartenberg standard convention and the joint variables are determined. Then we have for the global transformation matrix:

\[
A_n = \begin{bmatrix}
C\theta_nC\beta_n - S\theta_nS\alpha_nS\beta_n & -S\theta_nC\alpha_n & S\theta_nC\beta_n + S\theta_nS\alpha_nC\beta_n & a_nC\theta_n \\
S\theta_nC\beta_n + C\theta_nS\alpha_nS\beta_n & C\theta_nC\alpha_n & S\theta_nS\beta_n - C\theta_nS\alpha_nC\beta_n & a_nS\theta_n \\
-C\alpha_nS\beta_n & S\alpha_n & C\alpha_nC\beta_n & d_n \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

where the following parameters are valid: the distance \(d_n\), length of the link \(a_n\), joint angle deflection \(\alpha_n\), angle \(\beta_n\), joint variable \(\theta_n\). After that we have developed the full kinematic model for the 4R kinematical robotic structure.

3. Robotic workstations for contact tasks—modeling and control

Main aspects of this research could be formulated as follow: How to combine micro and macromotion and orienting of details; How to create hybrid system for different contact operations—assembly, grinding and polishing; Suggest new possibilities for building of Rapidly adaptable manufacturing cells; Discussion on easy reconfiguration of the robot workstation (cell) and ad-hoc integration of peripherals; Problems of cooperating robots and etc. The system state can be described as:

\[
S = \langle e, r, c \rangle
\]

Where \(s, r, c\) are usually expressed as vectors of the state of the effectors, sensors and control. The raw data obtained from real sensors usually cannot be utilized directly to control the system, so it has to be transformed into a useful form and this transformation is called data aggregation. As a result of this a virtual sensor reading \(v\) is obtained:

\[
V = F_v(r)
\]

Here \(F_v\) is called aggregating function. The software should be structured as a library of concurrent procedures and functions, which will be used as software blocks for construction of the control system. These software blocks must have the ability of reading and influencing each of the components of the system state—\(e, r, c\), so whether the mechanical and electronical part are modular or not the
Krassimir E. Georgiev

software part can be created as a set of modules and in such case at least the software component of the system can be tailored to the needs of the executed task. The proposed methodology of constructing controllers takes care of 2 problems: one is aggregation of data obtained from real sensors and the other is: synchronization between sensor data processing and the effector motions.

\[ V = F^*(r, e, c) \]

Now the system state can be decomposed by taking into account that the system can have several effectors and rather aggregated sensor readings \( v \) than real sensor readings \( r \) are used by the researcher.

\[ s = \langle e_1, \ldots, e_k, v_1, \ldots, v_n, c \rangle \]

where : \( k \) is the number of effectors in the system ; \( n \) is the number of virtual sensors ;Treating the system as a discrete time system, the next state of each of the effectors can be computed using a transfer function \( F_e \):

\[ E_{j+1} = F_e(e_j, v_1, \ldots, v_n, c) \quad j = 1, \ldots, k \]

Each effector control process creates or kills virtual sensor processes according to the needs of motion control. The effector control processes in each step obtain data from the virtual sensor processes. The both kind of processes can be treated as device dependent drivers. Both kinds of processes change the state \( c \) of the control subsystem. The virtual sensor process reads the real sensors, aggregates the obtained data and sends the result through data pipelines to the effector control process. For further speed-up, each effector control process can be portioned into 3 concurrent processes: future trajectory position generation taking into account virtual sensor readings, solving inverse kinematics problem and executing the joint control algorithms (reaching the generated position).

\[ T_{min} = f(N, m^I, l^I, v^I, t_k) \]

where \( N \) is the number of details, \( m^I \) - mass of the details, \( L^I \) - details dimensions, \( V^I \) - velocity, \( T_k \) - time of contact operations.

\[ T_{min} = 2 S^I N^I (H^I/\sqrt{v^I} + t_1) \]

where \( S \) denotes the sum of respective calculations on given \( I \)-group of details; \( H_i \) - an optimal distance to the placement of group of details.

For consistency of the robot’s model it is necessary to initialize the dynamic model too. The inertial parameters for each link are determined. This is a difficult and sometimes nearly impossible task. In this study the data is obtained from a solid body 3D model of the robot in Solid works environment (Fig.3). Building a virtual prototype and a dynamic model using Solid works and Cosmos Motion Software Virtual model of an adaptive assembly socket: the model is built upon real prototype in order to make dynamic research and to compare the data from the two objects. Virtual model of a 4R manipulators and a positioning robotized table: solid-body models are created for all components. Then they are joined together to form an assembly. This is done by creating mates and joints between the links. Coordinate frames are assigned for each joint according to Denavit – Hartenberg convention.
Modelling and design of robotic workstations for contact tasks

Fig. 2 Design, redesign and reconfiguration process

3.1 Computer simulations

Trajectory planning for the manipulator: the research is done for the last but one stage of the assembly process – vertical descend of the end-effector towards the assembled detail. The appropriate joint angles are determined in radians. Computing of the joint angles for straight line motion: joint angles of the manipulator are determined for applying motion to the end effector so it moves along a straight line trajectory, while maintaining constant orientation – vertical. For computing of the joint coordinates \( q_i \), the inverse kinematics is solved and \( q_i \) is determined for five points from the trajectory.

3.2 Modular design of robotic structure and peripherals

The modular component structure of environment supports a rapid exchangeability of models and allows to spread out modeling tasks and skills to different researchers in order to achieve sophisticated integration of capable models, reduce developing time and costs, create new innovative solutions. The calculated parameters are involved in the 3D kinematic model of the robotic system, using Solid Works 2005 - Cosmos Motion 2005. The mechatronic environment consists of different number modular components and mechatronics procedures: Each procedure consists of 5 steps: Initial synthesis of 3D kinematical mode, Preliminary metrical synthesis, Preliminary synthesis of control functions, direct and inverse dynamic tasks, Using multivariant analysis and varying the important characteristics to obtain optimal design of building modules, Final synthesis of the control functions (optimization).

Then for open and closed planar mechanisms we obtain integrated task of design, applying so called synthesis by using analysis and a feedback Df, which is described...
Krassimir E. Georgiev

the estimation of obtained parameters and a goal correction of the input mechanism parameters. This process is running as automatic iteration procedure. Using the developed virtual environment we can obtain many graphics of the current model parameters (force and velocity - Fig.4), which are subject to further improvement by the author.

Fig.4. Velocity and force calculations on 3D space

4. Conclusion

One promising approach to solve the prototyping issue is to apply virtual reality techniques to engineering design. This approach is used to 3D CAD model design ideas, simulate a mechanical part or assembly and evaluate ergonomics in a virtual environment. The virtual engineering refers to a number of virtual developing methodologies and means that design and validation activities occur collaboratively in order to optimize and prove product and system— for example robotic systems for contact operations— assembly, grinding and polishing.

REFERENCES