Robotica (1987) volume 5, pp 265–266

# Introduction to the special "CAD in Robotics" issue

Andras Siegler

*Computer and Automation Institute of the Hungarian Academy of Science, H-1111 Budapest, Kende u. 13.–17. (Hungary)*

(Received: October 30, 1986)

The intensive development of computer-based design systems used in the manufacturing industry has major impact on the design of robot control systems, and, especially, on the development of robot programming methods. It is well known that by merely putting robots into the manufacturing environment doesn't result in considerable improvement in productivity and usage of manpower. In order to achieve this requirement a considerable number of measures are needed concerning the organization of the design and manufacturing process. One of the organizational and design problems is the method of teaching-in and programming industrial robots. Significant increase of productivity can be expected from the application of robots in that case when the teach-in process making the robots able to perform a given task or a variety of tasks doesn't slow down the production i.e. it doesn't require to divert manufacturing equipment from production.

Making the teach-in phase and the productive use of robots independent in space and time, designing robot operations in advance in the course of man-computer communication, automating the process of designing robot operations are some of those technical development topics which can be characterized with the term "Application of CAD methods in robotics".

This subject incorporates very divers fields; here I mention the most typical ones:

- Kinematical and dynamical description and design of robots by CAD tools;
- Graphical-geometrical modelling of robots and their production environment;
- Designing off-line programming languages for robots;
- Designing motion paths of robots (collision detection, optimization of the usage of production environment).

From among the above enumerated topics this special issue of *ROBOTICA* concentrates mainly on several aspects of applying CAD tools in designing the path of robots and in their off-line programming. The first three contributions to the "CAD tools in robotics" issue deal with geometric modelling tools used in robot programming and computer visualization of robot activity.

The list of contributions starts with an application research paper "CAD-based off-line programming of painting robots reporting on simulation tools used for designing optimal paths for paint spraying robots. The algorithms described in the paper are used primarily for calculating and visualizing the paint distribution over an arbitrary surface. The paper presents a three-dimensional modeller which is the tool for designing the surfaces to be painted and to represent the robot itself. I would like to draw the reader's attention to the path optimizer of this package which minimizes paint loss and ensures good coating evenness.

Considering that in the course of the off-line programming procedure the robot and its working environment are not available they have to be modelled graphically in detail and displayed to the programmer. A significant amount of geometrical calculations must be performed in order to detect collisions already in the simulation phase. The second paper having the title "Geometric tools for the off-line programming of robots" describes the application of the *BUILD* modeller to robot off-line programming and gives a survey of the geometric problems encountered in this kind of robot programming.

Besides the three-dimensional geometric modeller the other tool needed for robot simulation is a specific software environment which enables the programmer to describe the structure of moving mechanisms, to design the scheduling and path of the motion itself as well as to program visualization of the simulated process. The next paper under the title "A 3-dimensional computer animation system with robotic applications" presents a computer animation programming language which features built-in robot programming tools as well. This animation package has been designed for multiple purposes – it is a general tool for creating movie-like scenes and describing the structure and motion of actors playing a role in them as well as it is a robot simulation system, including the programs of world-to-joint transformation for different kinds of mechanisms as well as tools for playing back and visualizing robot programs.

Robot simulation packages of the kind described in the first three papers are often used as experimental tools for world modelling and path planning algorithms. The paper "Computer simultation of sensor-based robot collision avoidance in an unknown environment" deals with the problem of sensor driven motion planning. The topic of this paper is a robot path simulation system used for testing robot collision avoidance in an environment with unknown obstacles. Real-time graphics animation is used to study the behaviour of an autonomous vehicle or a manipulator arm.

One of the main problems of general purpose robot programming and simulation systems is that the available programming tools must be independent of any given robot and possibly they should be portable to different microprocessor configurations. The topic of the paper

https://doi.org/10.1017/S0263574700016271 Published online by Cambridge University Press

"ARC: A robot programming and control system" is a development environment aimed at creating, simulating and executing robot programs. Robots are here considered as part of an FMS system, and their programming and control is performed in a 4-level hierarchy incorporating manipulator-, object-, task- and FMS-levels.

The series of guest papers on different aspects of CAD tools used in robotics ends with an analysis of currently available robot languages having built-in facilities for sensor communication. Under the title "Towards sensor-based general purpose robot programming languages" the author surveys the requirements of sensor signal processing at the language level. The paper also presents a framework of a high-level, block-structure robot programming language and outlines its implementation.

In closing this Introduction I would like to express my gratitude to Professor J. Rose, the editor of *ROBOTICA*, for inviting me to become the guest editor of this special issue on CAD tools in robotics. I also would like to thank all contributors of this issue for their efforts to prepare informative and valuable papers reporting on their recent research work in the field of computer-aided design and robotics software.

https://doi.org/10.1017/S0263574700016271 Published online by Cambridge University Press