Overview Of Programming Concepts In Robotics

Belana Roman
Institute of Flight Guidance
German Aerospace Center DLR
Brunswick, Germany
Belana.Roman@dlr.de

Noah Wiederhold
Institute of Flight Systems
German Aerospace Center DLR
Brunswick, Germany
Noah.Wiederhold@dlr.de

***Abstract*—This paper gives an overview of the programming concepts in robotics. It is intended to be used as a general source of information. The paper is structured in a way that the reader get to know a number of concepts one at a time. The paper is concluded with a summary of the programming concepts in robotics.**

***Index Terms*—robotics, programming, concepts, ai, augmented reality, virtual reality**

I. Introduction

II. Overview of Concepts

old-fashioned concepts widely used in the industry

- Online Concepts
  - Playback
  - Master-Slave
  - Teach-in
  - CAD/graphical based
- Offline Concepts
  - CAD/graphical based
  - text based
  - task based
  - Simulation
- Hybrid Concepts

new or theoretical concepts not widely used in the industry, published in scientific papers

- Semantic Robot Programming

III. Online Concepts

Programming with online concepts mean working with the active robot and its controls. This concept is used to give a robot a new set of skills in a fast and easy way, where the programmer has the chance to observe the resulting behavior directly. Commonly used concepts are Teach-in-Programming and Master-Slave-Programming.

A. Teach-in

With Teach-in-Programming the programmer teaches the robot needed sequences of movements. Therefore the programmer moves the robot via control elements or buttons, so the system can save the needed movements parameters like position, joint coordinates or the state of grippers and ”learn”. The movement of the robot can be controlled via consoles or so called ”Teach Pendants”, handheld programming devices. Usually, due to security, the movements are teached with decreased speed. Later on the program paramters like speed or accuracy can be adjusted to meet the needed specifications. Then the programm can be execute automatically, in which the robot moves through all stored positions one after the other and thus executes the planned sequence of movements. Usually there are three forms of movements are distinguished:

- Point-to-Point
- Continous Path
- Muli-Point

Play-back programming is for example a special from of Teach-in-Programming commonly used for Multi-Point. In this the robot is programmed by demonstrating the movement by touch or hand guidance with switched off actuators. Then the robot stores the positions of the joints and interpolates a smooth path with the given points, which can then be traversed as it was shown.

B. Master-Slave

The Master-Slave-Concept gives the chance to program heavy robots via online programming wihtout having to move them manually. To do this, the programmer needs two coupled robots, a small one that is easy to move and the heavy robot whose capabilities are to be programmed. The programmer moves the small robot, the so called Master. These movements are then copied from the so called Slave, the heavy robot. Because of the need of two coupled robots, this programming concept is usually expensive and therefore only used for teleoperations, so for places humans can not visit easily like under water, irradiated areas or in space.

C. Disadvantages of online programming

Even though online programming makes it possible to specify motion sequences very precisely, this type of robot programming is not useful or even possible for all applications. This concept makes it impossible, for example, to control the program flow beyond the movements, to process sensor data or to perform mathematical calculations. In addition, online programming requires time, which is a great disadvantage within a manufacturing process. Within this time, the robot is withdrawn from the process or possibly the whole process has to be stopped for this time. For such problems, concepts of offline programming are used.

IV. Offline Concepts

development doesn't take place on an active robot itsself but on a seperate system, indirect programming the program gets loaded onto the robot later on

(b4 p 186)

A. CAD/graphical based

B. text based

text based = explicit you tell how to move to accomplish a certain task

uses problem solving programming languages gives access to commands for movements with specific parameters for the specific movement

commands mostly define movements from on point to another and how to interpolate between them

programming enviroments for text based from the most common robot manufacturers

examples abb robot studio uses rapid language

kuka officelite uses krl language

there are many programming languages to choose from developing using the textual concept based on the 2019 market share of todays biggest robot manufacturers the most commong languages used are KAREL, RAPID, KRL (s1)

in addition to the textual programming features most of the enviroments also offer a graphical programming interface

the main concepts used in these enviroments are controlling the robot by moving its arms on different axis and rotating it on those or to simulate a controller (representation of the ones used in the online concepts)

kuka mainly uses the second concept where as abb uses both

(https://new.abb.com/products/robotics/robotstudio)

some enviroments even make use of ar technologie to show the roboter and its movements in the real world

example robot studio

C. task based

task based = implicit you tell what task to do to achieve a certain goal more abstract than text based, adds a layer of abstraction to the programming workflow

description of the tasks gets translated into lines of code by a task transformer (b1 p 116)

uses sensor information to add dynamic to the program while running

example task based programming enviroment: RoboGuide from FANUC defining by drag and drop Visual Components by flowchart Robotmaster

D. CAD/graphical based

uses a 3d scenery viewer which shows the simulated production enviroment

can simulate multiple robots and multiple tasks apart from robot movements

example visual components

E. Simulation

V. Hybrid Concepts

mixing of both offline and online concepts can be done by creating a general program offine and later fine tune this programm with online concepts on the real robot or the other way around by first programming the robot online and then optimizing the program offline (b4 p 186)

VI. New Concepts

A. Semantic Robot Programming

= programming by demonstrating

in a paper ... the authors propose a new concept of programming robots by demonstrating the desired behavior to the robot by giving an initial state and a goal state for which the robot has to find the best way to get from the initial state to the goal state

they use a new scene estimation method called DIGEST which splits the scene into a scene graph representing the scene structure of the initial state only requirement is the information about the number of objects present in the scene

given now the structure of the initial state and the goal, a task planer to find the best way to get from the initial state to the goal state is used

the task planer generates a series of actions to accomplish the task (p1 p 2)

B. AR

(p3)

C. VR

(p2)

References

[1] Wolfgang Weber and Heiko Koch. *Industrieroboter: Methoden der Steuerung und Regelung*. Hanser eLibrary. Hanser, München, 5., aktualisierte und erweiterte auflage edition, 2022.

[2] Flexible produktionsplanung dank einfacher roboterprogrammierung. *JOT Journal für Oberflächentechnik*, 60(5-6):32–33, 2020.

[3] Alexander Japs, M.Sc. (Produktmarketing Control Systems) und Stefan Kuhnert (Produktmarketing I/O-Systems), beide Phoenix Contact Electronics GmbH, Bad Pyrmont. Offenes ecosystem für grenzenlose automatisierung. *Konsturktion & Entwicklung*, 2021.

[4] Helmut Maier. *Grundlagen der Robotik: Helmut Maier*. VDE VERLAG GMBH, Berlin, 3., neu bearbeitete und erweiterte auflage edition, 2022.

[5] *Towards a robotic society: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems : October, 1-5, 2018, Madrid, Spain, Madrid Municipal Conference Centre*, Piscataway, NJ, 2018. IEEE.

[6] *2020 17th International Conference on Ubiquitous Robots (UR)*, Piscataway, NJ, 2020. IEEE.

[7] Zhen Zeng, Zheming Zhou, Zhiqiang Sui, and Odest Chadwicke Jenkins. Semantic robot programming for goal-directed manipulation in cluttered scenes.

[8] Gabriele Bolano, Arne Roennau, Ruediger Dillmann, and Albert Groz. Virtual reality for offline programming of robotic applications with online teaching methods. In *2020 17th International Conference on Ubiquitous Robots (UR)*, pages 625–630, Piscataway, NJ, 2020. IEEE.

[9] Camilo Perez Quintero, Sarah Li, Matthew KXJ Pan, Wesley P. Chan, H. F. van der Machiel Loos, and Elizabeth Croft. Robot programming through augmented trajectories in augmented reality. In *Towards a robotic society*, pages 1838–1844, Piscataway, NJ, 2018. IEEE.

[10] Statista. Industrial robots worldwide: market share 2019 — statista, 19.02.2023.