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

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

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 a 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

IV. Offline Concepts

A. CAD/graphical based

B. text based

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

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

1) explicit:

2) implicit:

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

E. Simulation

V. Hybrid Concepts

Acknowledgment

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks . . .”. Instead, try “R. B. G. thanks. . .”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

References

Please number citations consecutively within brackets [**?**]. The sentence punctuation follows the bracket [**?**]. Refer simply to the reference number, as in [**?**]—do not use “Ref. [**?**]” or “reference [**?**]” except at the beginning of a sentence: “Reference [**?**] was the first . . .”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [**?**]. Papers that have been accepted for publication should be cited as “in press” [**?**]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [**?**].

References

[1] Wolfgang Weber, Heiko Koch, ”Industrieroboter Methoden der Steuerung und Regelung” Carl Hanser Verlag München, 5. Auflage, 2022