Proceedings of the 29th International Conference
2018 Cybernetics & Informatics (K&I)
Jan. 31 - Feb. 3, 2018, Lazy pod Makytou, Slovakia

Control and Positioning of Robotic Arm on CNC Cutting Machines and Their Applications in Industry

Michal Maťuga
Real-time Control System R&D
MicroStep s.r.o.
Bratislava, Slovakia
matuga@microstep.sk

***Abstract*—The paper presents main challenges, problems, and steps involved in software implementation of robotic arm control. The arm is a multi axis robotic arm produced by 3rd party companies. It is embedded into a complex CNC cutting system with multiple motion axes and cutting heads. The control algorithm is implemented into existing complex software of real-time control system of the cutting machine. It has to be compatible with all existing and future modules of the software.**

Keywords—robotic arm; interpolator; control system; CNC; cutting machine; industry; laser cutting; plasma cutting; oxyfuel cutting; water cutting;

I. Introduction

With embedding of robotic arm into existing complex cutting system requirements for the machine control system are increased. Mechanical construction of the robotic arm is not a single module, but it is a part of the multi-module system with different motion axes, cutting technologies and manipulating mechanisms.

Different motion axes of the cutting machine have to be closely synchronized with joint motion axes of the arm. Switch of active modules on the cutting machine (e.g. switch from a plasma torch on the machine frame built by MicroStep, to an oxyfuel torch on robotic arm) has to be smooth, fast and fully automatic. That means that no control system of the arm manufacturer can be used. Control of the robotic arm has to be implemented into the existing control system of the machine.

The control system architecture implements different layers and modules. Every layer consists of several modules. High-level application layers (torch selection, manual/automatic system mode control) controls low-level layers (unloader/loader control, technology 1-N control...). The aim is to develop a module of robotic arm control with compatible interface to the high-level control modules. The module has to provide information about current state of the robotic arm and its associated technology.

II. Arrangement of the Robotic Arm on Cutting Machine

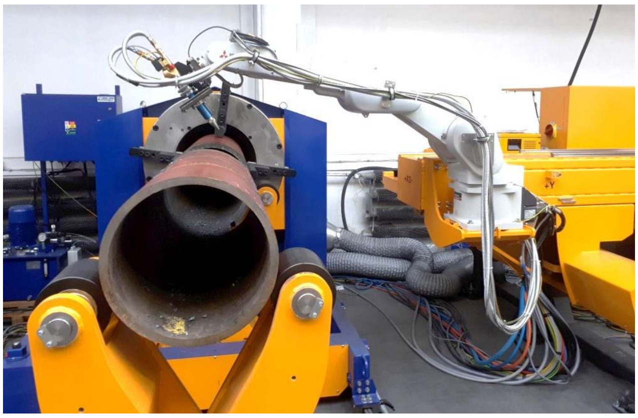

Fig. 1. Example of module with the robotic arm used to cut steel pipe

The mechanical construction of the robotic arm consists of these main parts:

- Robotic arm mount. The mount can be moved with other axes of cutting machine
- Robotic arm fixed to the mount. It can be produced by different manufacturers for example Mitsubishi, ABB, or Comau Robotics
- Electronic control equipment of the arm, for example motors, position sensors, motion controllers, power supply, safety circuits, etc.
- Cutting technology equipment, e.g. oxyfuel or plasma torch, laser or water nozzle with power sources
- Cutting technology mount, the interface between the torch and the robotic arm. It contains safety circuits preventing damage to torch or material in case of any collision.
- Material position sensors for online and offline compensations of material position

978-1-5386-4421-8/18/$31.00 ©2018 European Union

III. Main implementation steps

A. Positioning of the robotic arm on the cutting machine

Every cutting machine is designed to fit closely to customer`s needs. That means that every machine is different and in design phase it requires special attention. To examine correct positioning of the machine, it is needed to perform simulations of the robotic arm.

The input to the simulation is a full 3D model of the cutting machine. It has to contain full model of robotic arm with cables, arm, sensor and torch mounts. It also has to contain other modules of the cutting machine and specification of stock material shape and dimensions.

The simulation examines collision situations between the modules of cutting machines itself and also between a machine and a workpiece of all dimensions.

If some collisions cannot be prevented, simulation result is a definition of tasks for the collision control module in control system of the machine.

The simulation also examines validity of robotic arm position relative to workpiece, that all needed cutting torch positions are accessible.

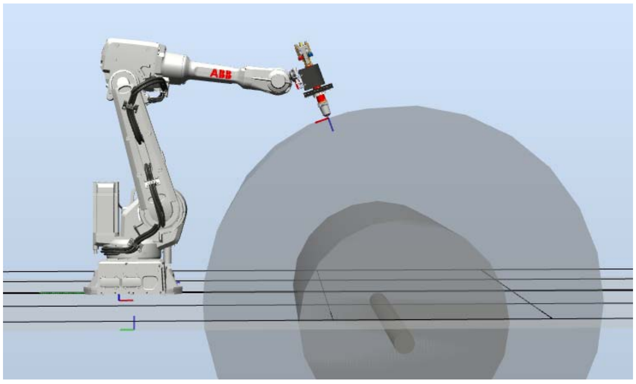

Fig. 2. Simulation of the relative position between the robotic arm and the stock material

B. Machine workflow definition

For correct design of the software architecture of a module it is important to define the workflow. That means switching between different states in the lifetime of the software module.

After machine is powered on, no position of the motion axes is known. Without power supply, the axes can be freely moved. Prior machine operation, initialization of the machine is required (movement of the module while sensor at the known position detects the module). This is not the case of robotic arms, because the arms have batteries powering position sensors, so the position information is not lost. That means, before machine initialization, it is possible (and needed) to move robotic arm into safe position, where collision between arm and modules at unknown positions is prevented.

When other modules of the cutting machine are used, the robotic arm has to be moved to the pose where it frees as much space as possible for movements of active module. This pose may vary between different active modules.

After activation of the robotic arm module other modules are repositioned to free possible space for the arm. Upon activation, machine operator in the manual mode and the control system in the automatic mode needs these main functions:

- Manual movements in XYZ axes - position in Cartesian coordinate system
- Manual movements in AB axes – tilt and rotation in Cartesian coordinate system
- Manual movements of separate joints
- Manual and automatic, offline and online measurement of material position
- Diagnostics, safety functions, error detection and handling, position information, technology state information and others

C. Software implementation

After definition of all requirements on the software, the implementation can be started.

IV. Software implementation

The implementation is divided into less complex steps, which enables better time and human resource planning in the development phase. Significant part of the tasks related to software implementation does not include programming works:

1. A mathematical model of the robotic arm based on forward and inverse kinematics
2. Verification of the mathematical model on the robotic arm hardware. Measurement of positioning accuracy and definition of requirements on calibration of the model
3. Calibration of the mathematical model
4. Design of software architecture of robotic arm control module
5. Final software implementation of designed architecture

A. Design of mathematical model of the robotic arm

For successful control of effector motion it is necessary to identify relation between positions of motion axes and a pose of the kinematic chain with respect to the world coordinates by means of forward and inverse kinematics. In case of cutting machines, the effector is the cutting torch mounted at the end of the arm.

The input to the forward and inverse kinematics calculations is a mathematical model of the arm kinematics. The model should be sufficiently robust and capable of representing all possible arm kinematics.

Manufacturers of the robotic arms provide only ideal dimensions of their products. For example in practice, translatory motion in one axis requires multi-axis compensation. Further calibration can be performed only provided that all transformations between any two joints are properly defined.

The mathematical model is created by generally known concept of coordinate system transformations between joints of the robotic arm. Full 3D translation (1) and rotation (2) of every joint of robotic arm.

$$T = \begin{bmatrix} 1 & 0 & 0 & t_X \\ 0 & 1 & 0 & t_Y \\ 0 & 0 & 1 & t_Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$R= \begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos\beta & 0 & sin\beta \\ 0 & 1 & 0 \\ -sin\beta & 0 & cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\alpha & sin\alpha \\ 0 & -sin\alpha & cos\alpha \end{bmatrix} \quad (2)$$

By serialization (multiplication) of the transformation matrices defining relations between each joint of the arm (1), (2), final transformation matrix can be computed. The variables in the matrix represent angular positions of arm axes. The parameters in the matrix express dimensions of parts of the mechanical construction of the arm and angles between axes of rotation of joints of the robotic arm.

The matrix defines relation between position of arm joints and world position (XYZ position + rotation around axes) of the effector (cutting torch).

B. Verification of the mathematical model on the robotic arm construction

The ideal dimensions of robotic arm mechanical construction have to be entered into created mathematical model to verify if the calibration of the model is needed.

For the verification of forward and inverse kinematics, actual position of the selected point on the mechanical construction have to be measured. This is an ideal application for a 3D tracker measuring device.

The 3D tracker is device consisting of interferometer (distance measurement device) driven in two angular axes with position sensors. The interferometer measures the distance to the reflector (the ball on the right in Fig. 3) while sensors on angular axes measures position of the interferometer.

The interferometer and its angular axes have highly precise sensors, leading to final accuracy of 3D position measurement better than 0.1 mm in the measuring range of 10+ meters.

Fig. 3. 3D tracker`s angular axes and interferometer (source faro.com)

With the help of the 3D tracker, positioning performance of the cutting torch tip can be measured and positioning accuracy of the torch tip can be examined.

The designed experiment consists of change of desired world position of torch tip and measurement of actual change. First measurements were carried out only for translation of the torch (no rotation around world axes).

TABLE I. REFERENCE VS. MEASURED TRANSLATION

Translation [X,Y,Z]	
Reference [mm]	***Measured [mm]***
[0, 20, 0]	[0, 19.9, 0.1]
[0, 200, 0]	[1.6, 198.7, -0.9]
[0, 0, 20]	[0, -0.5, 19.8]
[0, 0, 200]	[0.1, 2.5, 202]

The accuracy of torch positioning based on ideal parameters is not suitable for cutting applications which demands positioning precision better than 0.1mm. Positioning error changes in non-linear fashion because different dimensional errors play different role in different reference torch poses.

The effect of dimensional errors is also not acceptable when testing reference rotation of the cutting torch around one world axis. The cutting torch tip should be in the same pose when the rotation around world axis is executed (at the beginning of the coordinate system). Measured translation represents positioning error.

TABLE II. TRANSLATION ERROR OF CUTTING TORCH TIP ON DIFFERENT REFERENCE ROTATIONS OF THE TORCH AROUND WORLD AXIS

Reference rotation [deg]	***Measured translation of torch tip [X,Y,Z] [mm]***
10	[0, 0, 0.1]
20	[0.2, 0.2, 0.5]
30	[0, 0.4, 2.5]
40	[0.1, 0.7, 4.9]

Fig. 4. Measuring the oxyfuel cutting torch position on the Mitsubishi robotic arm with the Leica 3D tracker

C. Calibration of the mathematical model

In order to increase positioning accuracy, precise measurement of the robotic arm mechanical construction is required. With the help of 3D tracker, the identification of axes of rotation is very precise.

Motion axes of the robotic arm are rotational. Changing reference position of separate joint leads to circular movement of torch tip (or any physical point behind the joint). The axis of rotation of the circular trajectory is normal to the plane defined by the circle and runs through the circle center.

To increase measuring accuracy, it is possible to measure more positions (points) and approximate circle over the measured points, for example using a least-square method.

When the 3D position of the axes of rotation is known, the position difference can be measured directly, so the parameters of the mathematical model can be identified. This includes translation parameters (3D distance between axes), but as well as rotation parameters (3D angle between rotation axes).

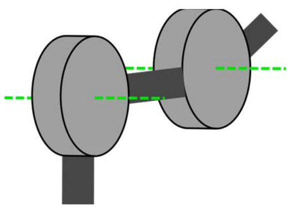

Fig. 5. Ideal axes of rotation of two joints are parallel, but in real world the angle is non zero and have to be measured

Dedicated 3rd party software *Polyworks* is designed for such measurements, and the positions can be directly measured into *Polyworks* workspace with the 3D tracker.

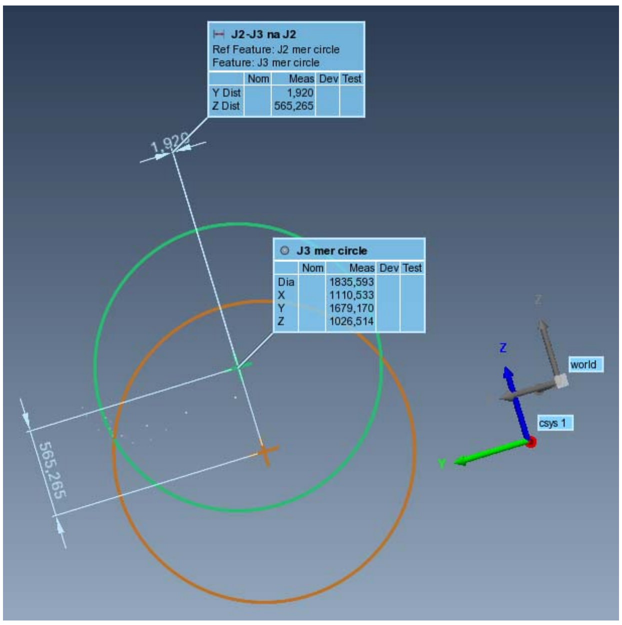

Fig. 6. Measuring distances between rotation axes of two joints in the software PolyWorks

The ideal dimensions provided by robotic arm manufacturer compared to measured ones were different in the orders of 0.1 mm or 0.1 degrees. After incorporating calibrated dimensions into mathematical model, positioning accuracy of the cutting torch as the effector of the robotic arm is acceptable.

D. Design of software architecture and final implementation

Designing of the software architecture is often neglected, but it is essential in complex, long developed software. This step defines co-operating software modules, their topology (layers) and interfaces. The architecture is essential also at the time and human resource planning.

Complete software architecture is out of the scope of this paper, but this chapter presents the most important modules to implement control of robotic arm.

1) Path planner

The path planner receives reference positions of the movements. The module controls limits of axes or collisions between machine physical parts. With the implementation of the robotic arm control, the path has to be customized for needs of arm kinematics.

The linear movement of the effector in the world coordinate system does not guarantee linear movement of joint axes. If the start and end positions of movement are in the permissible range of the axes of the robotic arm, there is no guarantee that the position in the middle of the movement is not outside of the permissible range of the joints.

Also the constant speed of the linear movement in world coordinate system does not guarantee constant speeds of the joints. The speeds may also change directions, so the stop in the middle of the movement is needed.

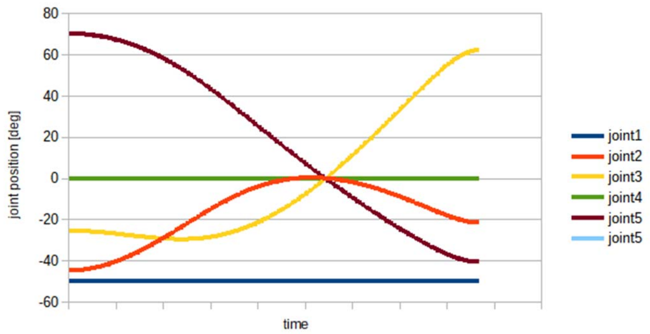

Fig. 7. Linear movement of the cutting torch in world coordinate system at constant speed requires reversal of speed of the joint 2 and increased requirements on the speed of the joint 3

Because of these facts, the movement has to be split into smaller steps. The planner computes end position for each step, checks if the position is inside range of the arm motion axes and computes dynamic requirements.

2) Interpolator

The precomputed movements are loaded into the interpolator that generates real-time reference trajectories for motion controllers of the cutting machine.

Because long movements are in the trajectory planner split down to small parts, the interpolator interpolates world and joint position of the movement linearly. The joint position is than sent to the motion controllers which creates negligible positioning error, because joint coordinates do not exactly match world coordinates. This approach significantly increases computation speed, because inverse kinematics does not have to be computed for every position sample generated in 2 kHz frequency.

3) Robotic arm control

The module of the robotic arm control communicates with the motion controllers of the arm motors. It controls the state of the controllers (switch to on/off...) and reads the state, actual position or errors.

The module also controls cutting technology of the robotic arm and its safety sensors. It is also responsible for measurements of the workpiece position requested from upper layers of the control system.

E. Life cycle of the desired movement of the robotic arm

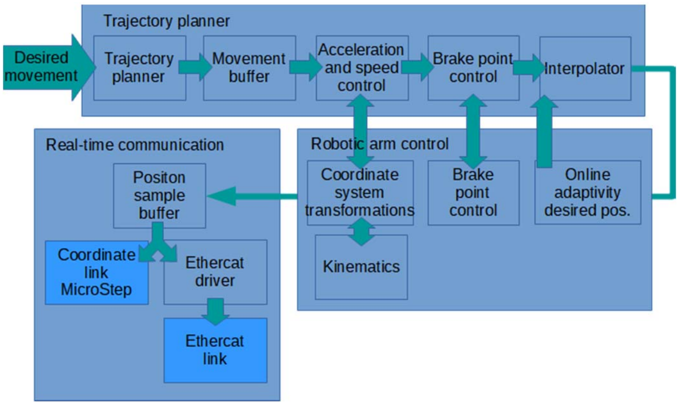

Fig. 8. Life cycle of the desired movement and software modules working with it

1) Desired movement

Desired movement contains information about desired position of the movement, speeds and movement type. There is a difference between fast movement to a destination and linear translation.

The movement is created in the upper layers of the control system.

2) Trajectory planner

Based on information in desired movement, the trajectory planner cuts the movement into smaller parts, 1 mm for translation axes and 1 degree for rotation axes. Partial movements are put into the movement buffer.

3) Movement buffer

Movement buffer is needed, because movements are processed in chunks. Bigger picture about the character of the planned trajectory is needed later, look ahead algorithms requires information about previous and following movements when processing the movement.

4) Acceleration and speed control

In order to control accelerations and speed of the joint motion axes of the robotic arm, inverse kinematics of the desired movement is required.

Before this computation there is a great opportunity to process final world coordinate transformations. As an example could be mentioned rotating of world coordinate system around the rotation axis of the cut pipe to increase range of robotic arm (Fig. 2).

When the solution of inverse kinematic task exists, desired speeds of all joints can be computed, and acceleration and speed limits can be applied to prevent exceeding limits of the joints by reference speeds.

5) Brake point control

Every movement can have non zero reference speed at the destination. Desired destination speeds should be recomputed with the regard to full trajectory. In case of series of short linear movement in the axis, with decreasing distance to final destination of the trajectory, the destination speed of the partial movements should also decrease, and the last movement should have zero destination speed.

6) Interpolator

Before interpolation, all movements were processed ahead of the time and stored in the buffer. The interpolator loads movement instructions incrementally and interpolates them in real-time while generating reference position samples for motion controllers.

This is the only step where real-time information about the state of the machine can be processed. Online adaptive control of cutting torch height (further referred to as adaptivity) uses live values from the corresponding sensors to measure the position of the material surface relative to the cutting torch and compensate for differences. It can add reference position changes only in this step. That means that the dynamics of the changed destination cannot be processed because a movement have to be interpolated in time. Adaptivity generates small changes of reference position so that the dynamics after this changes can be neglected.

Difference between two position samples correspond to current speed information for motion controllers because the samples are generated at constant and known frequency. The speed information for controllers has to be sufficiently smooth, so the update frequency has to be correspondingly high (2 kHz).

Because of operating system latency, the samples have to be generated little bit ahead of the time and stored in the small buffer (5 ms). The buffer has to be as short as possible, because it adds undesirable delay in emergency situations.

7) Real-time communication

Communication drivers are driven by synchronization interrupts from the communication links. These interrupts are hard real-time electrical impulses with zero latency and define time in all motion controllers. In the operating system they are processed with the highest priority.

In the interrupts are generated at 2 kHz, new reference position sample is loaded from the buffer and is simultaneously sent to the motion controllers on all relevant communication links.

V. Conclusions

Because of its flexibility, multi joint robotic arm is an interesting mechanical equipment of cutting machine. It introduces unique requirements for the control system designed to handle motion axes exactly in the same way as handling world coordinate axes.

The most significant challenge in implementation of the robotic arm control is the trajectory planning, acceleration and speed control. Further challenges are calibration of the mathematical model and implementation of online, real-time adaptivity.

The needs can be successfully satisfied with minor limitations, and the robotic arm can be considered to be an interesting and beneficial equipment that has potential to increase value of standard cutting machines.

Acknowledgment

The author would like to express his acknowledgement to MicroStep for financial support and encouragement. The work was partially supported by Ministry of Education, Science, Research and Sport of the Slovak Republic as a part of applied research project "Research of technological nodes of CNC machines for spatial cutting of materials using energy beam technologies", applied research project 2015-10962/19795:1-15AA according to Act No 185/2009 on incentives for research and development and to completion of Act No 595/2003 about Income Tax as amended.

References

[1] Mark W. Spong, Seth Hutchinson, M. Vidyasagar, "Robot Modeling and Control", Wiley, 2005, pp. 61-94, pp. 175-192

[2] Crane, Carl D.; Joseph Duffy, "Kinematic Analysis of Robot Manipulators", Cambridge University Press, 1998

[3] Denavit, J., Hartenberg, "A kinematic notation for low-pair mechanisms based on matrices", ASME J. Appl. Mechanics 77 (1955), pp. 215–221.

[4] "KDL Api documentation", [Online], cited 4. 11. 2017
Available: http://www.orocos.org/kdl

[5] "IKFast: The Robot Kinematics Compiler documentation", [Online], cited 2.12.2017
Available: http://openrave.org/docs/0.8.2/openravepy/ikfast/

[6] "LinuxCNC Cubic interpolator", [Online], cited 20.12.2017

[7] Chen, J. and Chao, L. M., Positioning error analysis for robot manipulators with all rotary joints, IEEE Journal on Robotics and Automation, 1987, pp. 539–545.
Available: http://wiki.linuxcnc.org/cgi-bin/wiki.pl?CubicInterpolator

[8] Curtis S. Wilson, "Precision contouring error analysis", [Online], cited 15.1.2018
Available: http://static.mah.priv.at/public/ASPE_Contouring_Paper.pdf

[9] Kuu-young Young, Jin-Jou Chen, "Implementation of a Variable D-H Parameter Model for Robot Calibration Using an FCMAC Learning Algorithm", April 1999, Vol. 24, pp 313–346