Int J Adv Manuf Technol (2017) 90:2389–2395
DOI 10.1007/s00170-016-9584-2

ORIGINAL ARTICLE

Dynamic performance of industrial robot with CNC controller

Kai Wu[1] · Carsten Krewet[2] · Jobst Bickendorf[2] · Bernd Kuhlenkoetter[1]

Received: 10 July 2016 / Accepted: 7 October 2016 / Published online: 15 October 2016
© Springer-Verlag London 2016

Abstract As the application of industrial robots in machining is constantly increasing, many techniques have been developed to make this use more efficient and accurate. The robot controller is one of the key factors to influence the robot performance. This paper discusses the performance of a CNC kernel which is directly integrated into the industrial robot. The dynamic motion of the robot is thoroughly analyzed. A conventional robot controller is carefully evaluated to make a clear comparison.

Keywords Robot motion · CNC · Robot control

1 Introduction

Nowadays, industrial robots are more and more applied in machining with their advantages of a high flexibility and large workspace. Robot-based machining systems have already been used for chamfering, deburring, and polishing processes, among others [1]. However, the industrial robot exhibits a limited stiffness, which varies significantly in different position configurations and directions that result in vibration/chatter within the machining process [2]. Additionally, when industrial robots are introduced in the process of grinding and milling, the lower accuracy of the robots becomes a problem to attain the manufacturing tolerance. To overcome these obstacles, two aspects are considered. One aspect is to optimize the structure of the industrial robot. There are robots which have been especially designed for particular machining tasks. ABB produces the IRB 6660 for pre-machining operations, which has an additional parallel arm to make the robot stiffer [3]. KUKA offers robots dedicated for milling like the KR 500 R2830 MT (machining tooling) with a payload of up to 500 kg and ±0.08 mm pose repeatability [4]. The Stäubli RX 170 hsm (high speed machining) robot substitutes the sixth axis by a high-speed cutting (HSC) spindle to increase rigidity and precision [5].

Another aspect is to improve the control system. Many control techniques have been proposed to raise the efficiency and accuracy of industrial robots. Mattias Björkman et al. developed a new generation of the ABB robot motion control which includes a model-based trajectory generator and a model-based axis controller. This control concept was implemented in an IRC5 controller. Linear paths and circular paths were tested to compare the normal controller and new concept controllers. A laser measurement system was used to measure the path errors. Experimental results showed that the path accuracy is improved by up to 50 % and the cycle time is reduced by up to 20 % without setting the robot life time at risk [6].

Jae Wook Jeon and Young Youl Ha analyzed existing techniques to control the acceleration and deceleration in the industrial robot and computerized numerical control (CNC) machine tools which were selecting polynomial functions and digital convolution techniques. Both techniques have their

---

✉ Kai Wu
wu@lps.rub.de

Carsten Krewet
carsten.krewet@tu-dortmund.de

Jobst Bickendorf
jobst.bickendorf@tu-dortmund.de

Bernd Kuhlenkoetter
kuhlenkoetter@lps.rub.de

[1] Chair of Production Systems, Ruhr-University of Bochum, Bochum, Germany

[2] Institute of Production Systems, TU Dortmund University, Dortmund, Germany

own limitations. Selecting polynomial functions has the problem of computation load especially when the order of the polynomial becomes higher. Digital convolution techniques are much more efficient than selecting polynomial function techniques and are easily implemented by hardware. But some velocity profiles that are useful for industrial robots and CNC machine tools cannot be generated by these techniques. So a generalized approach was proposed. According to the desired characteristics of acceleration and deceleration, each set of coefficients is calculated and stored. Given a moving distance, acceleration, and deceleration intervals, a velocity profile having the desired characteristics of acceleration and deceleration can be efficiently generated by using these coefficients. Experiments were implemented in a single-axis control system. An arbitrary velocity profile that cannot be generated by digital convolution techniques can be generated efficiently by the proposed technique [7].

Zhaohui Jiang and Taiki Ishita proposed to use neural networks for dynamic trajectory tracking control. A control system which contained both a neural network controller and a linear controller was established. The neural network controller with a three-layer feed forward network was designed and added to the control system in the parallel way to the liner controller. Simulations and experiments were carried out on an industrial manipulator AdeptOne XL robot. Results showed that the neural network controller takes the place of the linear controller to play a main role in the generating process of the actuating force/torque required by the dynamic trajectory. Besides the effectiveness and usefulness of this control method, it showed some limitations of neural network learning and the trajectory tracking accuracy remains unchanged after some specified times of learning [8].

Adel Olabi introduced a method of trajectory planning adapted for continuous robot machining. The path strategy was planned in the operational space for the prescribed path. The algorithm of end-effector feedrate planning was discussed in detail. FIR (finite-impulse response) filters were exploited to generate the tool feedrate with limited jerk. Based on the strategies, three trajectories along a logarithmic spiral path were planned and implemented for a six-axis industrial robot. The velocity distributions were presented, and contour errors were calculated referring to the theoretical contour. It was shown that the feedrate planning strategy is an effective solution to control the tool motion for a robot [9].

In addition to the general techniques in controlling the motion, special compensation mechanisms such as elasticity compensation are considered in RC (robot control) according to specific robot types of different manufacturers. Moreover, robot relevant parameters are also included like singularities, reach limits, and joint limits. As the RC is machine related, specialized robot functions can be provided, which have the advantage to quickly program for handling and automation tasks [1]. However, when machining workpieces, completely different movement strategies have to be applied. The CNC system, which has the advantage of high accuracy manufacturing, short production time, and greater manufacturing flexibility, has been widely used in machine tools. Different control systems have their own algorithms to operate the motion. In order to find out whether the robot motion under the control of a CNC kernel has a better performance than conventional controllers for machining tasks, these two control systems are discussed in this paper.

2 Control system

For machining tasks, the contour and the surface quality are of special importance. Sophisticated algorithms have to ensure a suitable movement. Functionalities such as the correction of tool geometries, the compensation of discordance, path-planning methods (B-Splines, Akima-Splines, etc.), the planning of path dynamics, or configurable contour deviations have to be provided. According to these requirements, CNC is supposed to be a workpiece-oriented technology which is applied in machining tools. There are some CAD/CAM-based systems already existing for the offline programming of robots [1]. But the generated CNC programs often need to be compiled by a corresponding post processor to directly run on the robot control. This would involve a loss of information, as functions from the CNC programs are not always supported in the widely different robot languages. A few years ago, KUKA integrated the CNC kernel in its robot controller which offers the possibility to execute CNC programs directly. Besides, it offers the function to switch CNC operations and conventional robot operations for different applications [10, 11]. The CNC kernel integrated in KUKA robots has the name KUKA.CNC which was developed by ISG (Industrielle Steuerungstechnik GmbH) [1, 12]. Meanwhile, the newest conventional controller for a KUKA robot is KR C4. KUKA.CNC can be installed under the same conditions of KR C4. It runs parallel to the KR C4 and has its own user interface. These two controllers have their own program languages. KUKA Robot Language (KRL) running on the KR C4 offers two types of programming forms, which are the user group (inline forms) and the expert group (KRL syntax). KUKA.CNC executes DIN 66025-compliant CNC programs [10]. The complete standard code can be interpreted and implemented by the robot (G functions, M/H/T functions, local and global subprograms, etc.). However, the CNC mode should be activated by the KRL program before running the CNC program. The KRL function DEF gCodeExecute (GCodeFileName:IN) can be used for this purpose [13]. As not all differences between these two control systems are discussed, typical motions are chosen to be analyzed to get the performance. The linear path is mostly used for the

Table 1 Program comparison between KRL and CNC in linear path

Parameters	KRL	CNC
Acceleration	Acceleration weighting	Acceleration weighting Weighting the ramp times Acceleration profile
Velocity	Set specific value	Set specific value Feed calculation based on the weighted maximum axis feeds (modal)
Position control	Exact stop CP approximation	Exact stop Default (approximation)

machining tasks, so it is discussed in detail in the following chapters [12, 14].

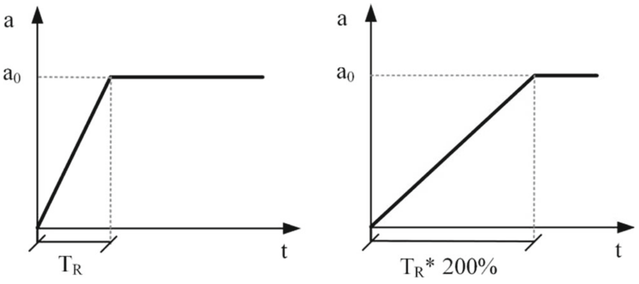
The functions to program a linear path are stated in Table 1 [14]. It shows that the big difference in these two control systems is the control of the acceleration. In CNC, there are three kinds of acceleration profiles to run a linear path that are step-shaped, trapezoidal, and square-sinusoidal profiles the corresponding names of which in CNC programs are profile 0, profile 1, and profile 2 (Fig. 1) [12].

In addition to the control of the acceleration profile, CNC has the function to regulate the ramp time which controls the time to reach the defined acceleration value (Fig. 2). There are four phases in the acceleration profile (Fig. 3): (i) increase in accelerating, (ii) decrease in accelerating, (iii) increase in braking, and (iv) decrease in braking. In each phase, the weighting of ramping time can be defined separately. The weighting is available only in respect of trapezoidal or square-sinusoidal acceleration profiles.

Referring to the KRL programming system, there is no command to define the acceleration profile. One parameter, which can be used for regulating the acceleration, is acceleration weighting [14].

3 Experiment

In order to analyze the performance of the industrial robot under two control systems, a group of paths has been planned based on the analysis of the two programming systems. A metrology system, which is a stereo high-speed camera system, is adopted for tracking the robot's motion. Based on the acquired motion data, the velocity, acceleration, and path accuracy are investigated. The planned paths and experimental setup are introduced as follows.



(a) Weighting value 100% (b) Weighting value 200%

Fig. 2 **a**, **b** Example for weighting of ramp time with G133 [14]

3.1 Experimental design

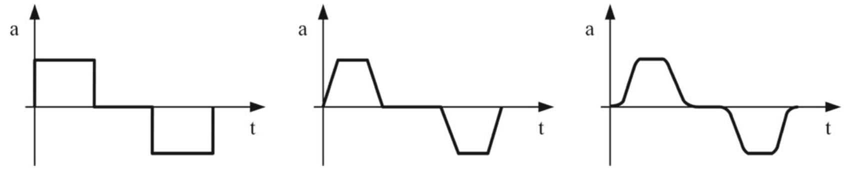
As discussed in Section 2, there are clear differences in the two control systems. Not all possible paths are programmed for testing. In the two control systems, there is no information about the definite acceleration value. In addition, KRL has no information on the acceleration profile. The regulation of the acceleration value is realized by the parameter acceleration weighting in percentage value. A preliminary test has revealed that the respective acceleration value differs a lot among the different types of paths. In order to get the performance of the robot in full acceleration, a short distance with high velocity is programmed. Moreover, this work considers the paths operated in same accelerations in the two control systems. The specific parameters are stated in Table 2.

3.2 Experiment setup

The experiment setup includes a KUKA robot, the metrology system, the testing tool, and reference object which are labeled in Fig. 4.

The KUKA robot (KR 210 R2700) with six axes to be tested has a total weight about 1111 kg and a maximum total load of 260 kg. The position repeatability can reach ±0.06 mm and the maximum reach is 2700 mm. The robot runs with the controller of KR C4. Under the conditions of KUKA System Software 8.2, KUKA.CNC 2.0 is installed. With the KUKA

Fig. 1 **a**–**c** Acceleration profiles in CNC [14]



(a) step-shaped (b) trapezoidal (c) square-sinusoidal

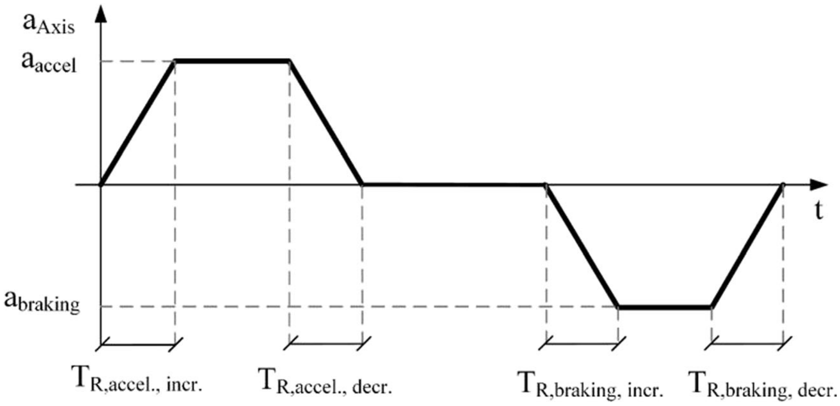


Fig. 3 Parameters of acceleration profile [14]

smartPAD (teach pendant for the industrial robot), it has been possible to conveniently program test paths directly on the smartPAD.

The metrology system is a high-speed camera measuring system named PONTOS. This system mainly consists of two high-speed cameras, illuminant, and PONTOS software (Fig. 5). The maximum camera resolution can achieve 2048× 2048 and the corresponding frame rate is up to 1080 Hz. Depending on the measuring area, its accuracy can reach 0.001 mm. According to different measuring areas and measuring distances, there are corresponding lenses to fit. In the measuring area, a multitude of positions could be measured which are indicated with markers (Fig. 6). According to the frame rate set in the system, a certain number of pictures would be captured which are imported to the PONTOS software for post-processing. Then the 3D coordinate values can be extracted. Some information can be directly obtained in the software such as the velocity, deviation, and acceleration.

The testing tool is installed on the flange of the robot, whose center is marked with a marker for being tracked by the metrology system. The designed paths are also programmed according to this center.

The reference object is used for establishing the robot's work coordinate system, which is built by teaching it three points marked on the reference object. The designed paths are consequently programmed according to the work coordinate system. Both control systems have operated under this work coordinated system. Each path has been repeated and measured three times. The specific measuring procedure has been introduced in the previous article [15].

Table 2 Parameters in linear path program

Type	Acceleration weighting	Velocity (mm/s)	Path length (mm)
KRL	25 %, 100 %	1000	200
Profile 0	100 %	1000	200
Profile 1	100 %	1000	200
Profile 2	100 %	1000	200

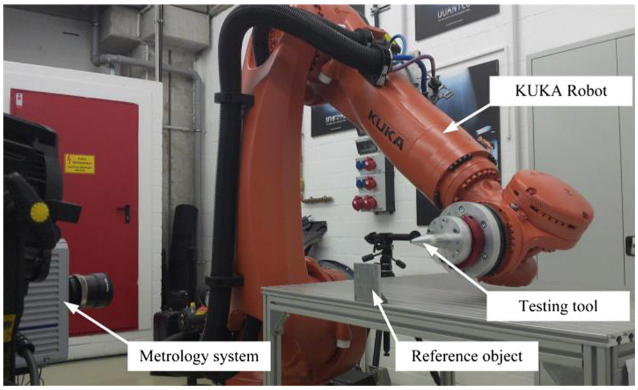


Fig. 4 Experiment configuration

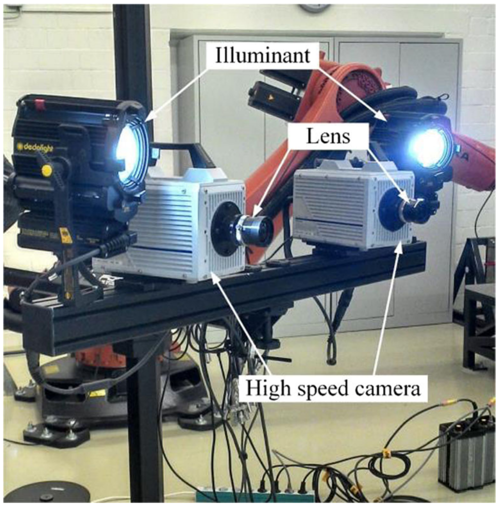


Fig. 5 PONTOS metrology system

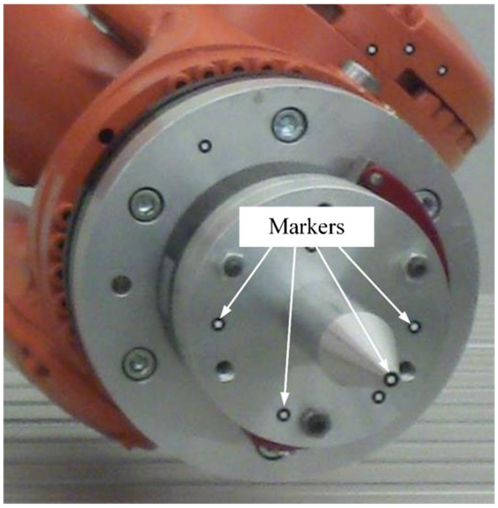


Fig. 6 Measuring markers

4 Experiment results and analysis

According to the experiment plan, it introduces paths with a running distance of 200 mm and a programmed velocity of 1000 mm/s. Based on the measuring data, the distribution of velocity, the acceleration values are obtained (Fig. 7). The direct calculation of the measuring data would involve a high noise level. So an adaptive window filter similar to the end-fit filter is used for the velocity and acceleration analysis [15, 16]. Moreover, the deviation to the programmed path is calculated to find the difference among the paths. As presented in Table 3, all paths have not reached the programmed velocity. The table shows that the KRL with 100 % acceleration weighting has the maximum acceleration value of 2334.77 mm/s^2 and a maximum running velocity of 554.35 mm/s. Different acceleration values and velocities ap-

Fig. 7 **a**–**f** Velocity and acceleration distribution of linear path with programmed velocity of 1000 mm/s

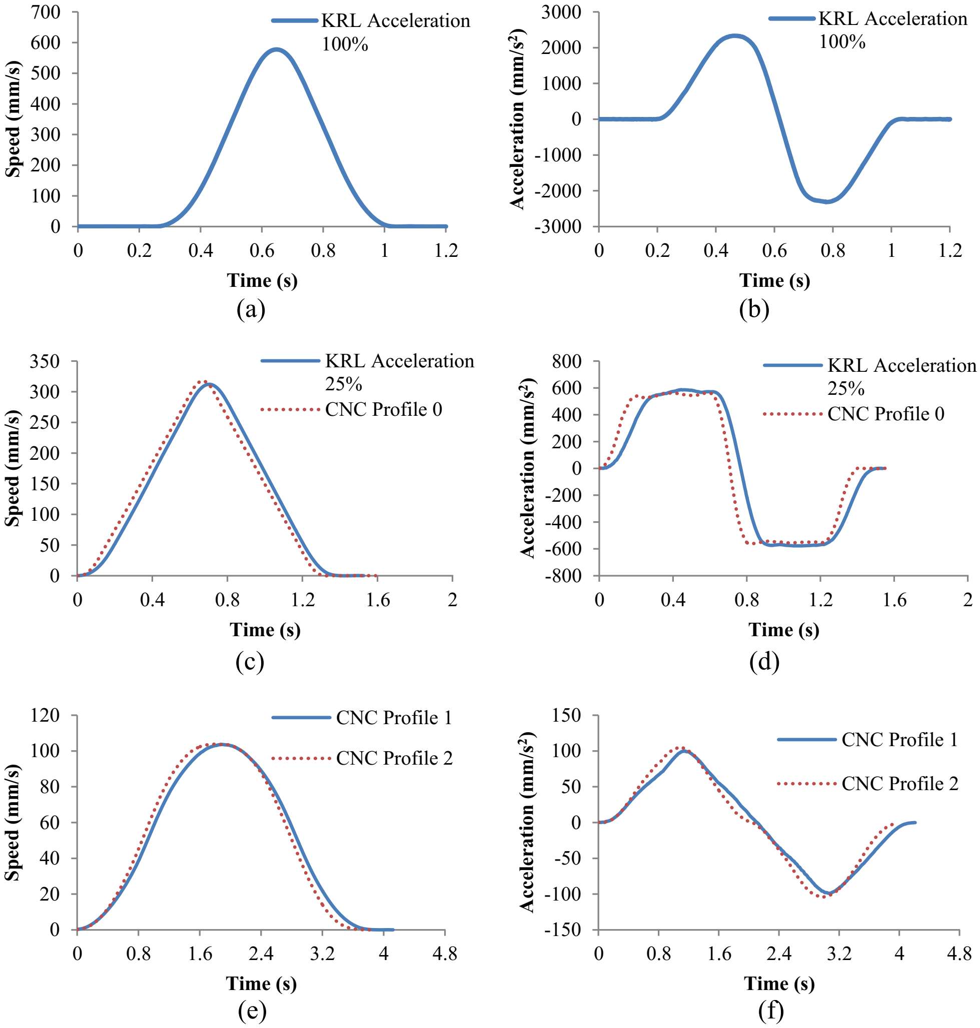


Table 3 Maximum velocity and acceleration of linear path with programmed velocity of 1000 mm/s

Type	Maximum velocity (mm/s)	Acceleration (mm/s^2)
Profile 0	317.79	554.35
Profile 1	103.63	99.91
Profile 2	103.79	105.33
KRL acceleration 25 %	312.17	571.33
KRL acceleration 100 %	577.63	2334.77

Table 4 Average deviation to the fitted straight line [mm]

Type	Average deviation	Maximum deviation
Profile 0 acceleration 100 %	0.094	0.355
Profile 1 acceleration 100 %	0.079	0.367
Profile 2 acceleration 100 %	0.085	0.371
KRL acceleration 25 %	0.110	0.385
KRL acceleration 100 %	0.175	0.446

pear in the CNC path. Profile 0 has a higher acceleration and velocity than the other two types. The difference between profile 1 and profile 2 is very small.

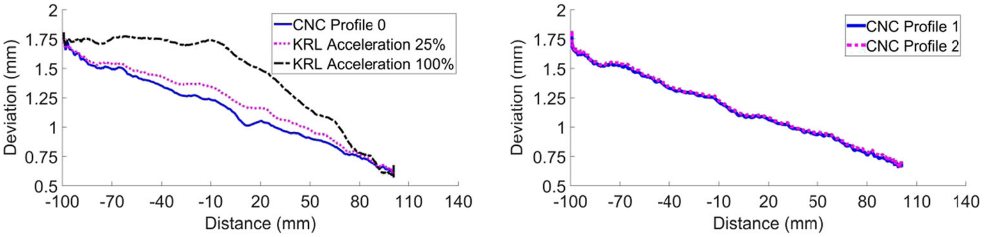
Comparing the KRL path with an acceleration weighting of 25 % with profile 0 with an acceleration weighting of 100 %, both paths have nearly the same acceleration distribution except that the acceleration of profile 0 is quicker to reach the highest acceleration value (Fig. 7d). Accordingly, its velocity is a little higher (Table 3). Profile 1 and profile 2 have the same maximum velocity (Table 3). Comparing the acceleration shape, profile 2 has a smoother change to reach the value zero (Fig. 7f), which indicates a coherence with the profiles in Fig. 1b and Fig. 1c.

The distribution of the deviations to the programmed path is shown in Fig. 8. The paths are furthermore fitted by a straight line with the principle of the least squares to assess the path's linearity (Fig. 9). The average deviation and the maximum deviation to the fitted line are obtained (Table 4). It shows that the path with 100 % acceleration weighting in the KRL has the biggest average deviation and maximum deviation. When the weighting value of 25 % is taken, the deviation becomes smaller. This illustrates that the lower the acceleration, the lower the fluctuation. The CNC paths show lower deviations than the KRL paths. It is reasonable to compare profile 0 and the KRL with 25 % weighting value. These two types of paths have similar accelerations and velocity profiles. However, both the average deviation and maximum deviation of the KRL path are larger than that of the profile 0 path (Fig. 9a and Table 4).

5 Conclusion

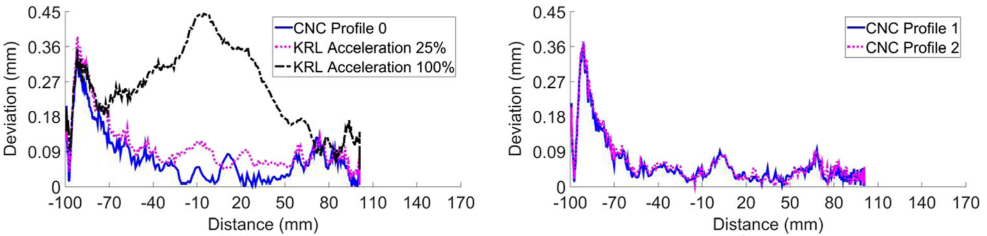
In this paper, linear paths have been compared in a CNC controller and a KR C4 from the aspects of programming and motion. With regard to programming, CNC shows more commands than KRL to control the acceleration profile. Through tracking the robot motion, the results illustrate that the KRL can achieve higher acceleration values than the CNC

Fig. 8 **a**, **b** Deviation to the programmed path



(a) (b)

Fig. 9 **a**, **b** Deviation to the fitted straight line



(a) (b)

controller. The linear movement has obvious deviations when the robot runs at high velocity with full acceleration in KRL. When the acceleration and running speed are similar under two control systems, the experimental results show that the CNC path has, to some extent, a better linearity than the KRL path. However, the corner path and circular paths etc. are not included. Besides, the industrial robot is moved without load. Therefore, further studies shall focus on the dynamic performance of robotic machining. Different types of trajectories could be implemented with industrial robots under the CNC control system. As the machining process comprises continuous contact forces and excitations, the industrial robots would have deformations and vibrations correspondingly. This would influence the surface quality of the workpieces. For these reasons, it is advisable to investigate the machining dynamics [17] and the control strategies for a better understanding and to improve the robotic machining in the long run.

References

1. Karim A, Verl A (2013) Challenges and obstacles in robot-machining, in: IEEE ISR. IEEE, Seoul 2013:1–4. doi:10.1109/ISR.2013.6695731
2. Chen Y, Dong F (2013) Robot machining: recent development and future research issues. Int J Adv Manuf Technol 66:1489–1497. doi:10.1007/s00170-012-4433-4
3. ABB, High Performance Robot, (n.d.). http://www.abb.com/product/seitp327/63bc9f7c7410ca74c125744200314d16.aspx. Accessed 2 Feb 2016
4. KUKA Roboter GmbH, Heavy duty robot in MT-version, (n.d.). http://www.kuka-robotics.com/germany/en/products/industrial_robots/heavy/kr500_fortec/kr480_r3330_mt/start.htm. Accessed 2 Feb 2016
5. Stäubli, High speed machning (hsm), (n.d.). http://www.staubli.com/en/robotics/6-axis-scara-industrial-robot/specialized-robot/high-speed-machining-robots/. Accessed 2 Feb 2016
6. Björkman M, Brogårdh T, Hanssen S, Lindström S-E, Moberg S, Norrlöf M (2008) A new concept for motion control of industrial robots. http://www.control.isy.liu.se/publications/
7. Jeon JW, Ha YY (2000) A generalized approach for the acceleration and deceleration of industrial robots and CNC machine tools. IEEE Trans Ind Electron 47:133–139. doi:10.1109/41.824135
8. Jiang Z, Ishita T (2008) A neural network controller for trajectory control of industrial robot manipulators. J Comput 3:1–9. doi:10.4304/jcp.3.8.1-8
9. Olabi A, Béarée R, Gibaru O, Damak M (2010) Feedrate planning for machining with industrial six-axis robots. Control Eng Pract 18: 471–482. doi:10.1016/j.conengprac.2010.01.004
10. KUKA Roboter GmbH, KUKA.CNC for maximum robot performance in machining processes, 2011. http://www.kuka-robotics.com/res/sps/a737ee03-5832-4c95-9d91-84e0de80c664_kuka_cnc_englisch.pdf
11. Bay A (2012) KUKA.CNC—a direct approach. Auto Tech Rev 1: 54–55. doi:10.1365/s40112-012-0194-6
12. Industrielle Steuerungstechnik GmbH, Programming manual, Stuttgart, 2010
13. KUKA Roboter GmbH, KUKA.CNC 2.0 for KUKA System Software 8.2, 2012
14. KUKA Roboter GmbH, KUKA System Software 8.2 operating and programming instructions for system integrators, 2011
15. Wu K, Brueninghaus J, Johnen B, Kuhlenkoetter B (2015) Applicability of stereo high speed camera systems for robot dynamics analysis. Int Conf Control Autom Robot, IEEE, Singapore 2015:44–48. doi:10.1109/ICCAR.2015.7165999
16. Janabi-Sharifi F, Hayward V, Chen CJ (2000) Discrete-time adaptive windowing for velocity estimation. IEEE Trans Control Syst Technol 8:1003–1009. doi:10.1109/87.880606
17. Cheng K (Ed) (2009) Machining dynamics: theory, applications and practices. Springer-Verlag London. doi:10.1007/978-1-84628-368-0