IMPROVED METHOD FOR HIGHLY ACCURATE INTEGRATION OF TRACK MOTIONS

Michael Kleinkes, Werner Neddermeyer and Michael Schnell
University of Applied Science in Gelsenkirchen
Neidenburger Str. 43, 45897 Gelsenkirchen

Keywords: Industrial robots, 7^{th} axis, linear track, offline programming, motion control, flexible automation.

Abstract: Modern Robotics today deals with increasing requirements on the flexible automation. One of this is the usage of linear tracks or even called 7^{th} axis to extend the robots workspace. The inaccuracies of the linear track deteriorate the accuracy, which is in constrast to highly accurate robot systems needed for modern applications. To enhance the accuracy of the system consisting of robot and linear track, an identification of the non-linearities of the linear track is necessary. This article introduces an optimisation of a method for highly accurate integration of track motions where the profile of the linear track is identified by single coordinate systems along the track, combined by a cubic spline interpolation. Resulting there is a continous description of the track profile, depending on the current position of the robot on the linear track.

1 INTRODUCTION

Modern industrial robot applications become more and more complex. The increasing demand on flexibility and automation effects the need of high precision robot systems. One part of this robot systems are external linear tracks, even called 7^{th} axis, on which the robot can be linear moved to extend its workspace. In most of the cases this extension of the workspace via linear track deteriorates the accuracy of the system robot and track in significant number about several millimeters or more. This alarming fact is not compatible to the demands on the flexible automation at all and additionally is neglected by scientists and robot manufacturers at well.
In (6) there is one method presented, which makes it possible to identify the inaccuracies of the linear track and correct offline robot programs. For this, the linear track is measured at n positions where a corresponding track coordinate system is calculated. The number of positions on the linear track where the single track coordinate systems are calculated - following called sampling points - are determined by a frequency scan of the linear track. To get a continous description of the linear track these track systems are combined by a cubic spline interpolation. By this it is possible to get a correction frame for each arbitrary position on the linear track. This paper sets up on the first article about highly accurate integration of track motions and presents optimisations in the fields of frequency check and spline correction and fulfils parts of the future prospects of this article.

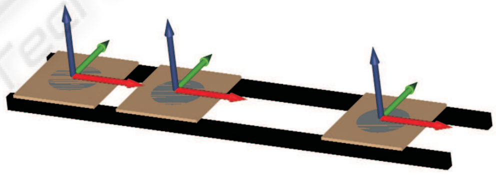

Figure 1: Track coordinate systems.

2 TRACK PROFILE FREQUENCY CHECK

For the identification of the track motion one rigid object is needed, which is to measure at different positions on the linear track, by an 3D-coordinate measurement device. Basing on the fact that the rigid object is not deformed after moving from one

Kleinkes M., Neddermeyer W. and Schnell M. (2006).
IMPROVED METHOD FOR HIGHLY ACCURATE INTEGRATION OF TRACK MOTIONS.
In *Proceedings of the Third International Conference on Informatics in Control, Automation and Robotics*, pages 469-473
DOI: 10.5220/0001203504690473

position to the next position on the linear track, the positional changings of some measurement points on the rigid object are only caused by the linear track.
The very high reapeatability of modern industrial robots makes it possible to use the robot as such a rigid object. The inaccuracy in moving to one point repeatedly is in most of the cases more then 10 times less than the inaccuracies caused by the linear track. In (7) there is an overview about the absolute and repeat accuracy of the used robot.
One important step in the process of identifying the linear track is the determination of the needed number of sampling points on the track. This value is depending on the maximum error which has to be identified and varies with each individual linear track. A special procedure developed before is here used again and is in certain ways optimised. For this, the robot is moved along the linear track with its Tool-Center-Point (TCP) in one constant position. During the robots movement it is measured by the external measurement system in a continious measurement mode. This creates measurement data from a theoretical straight line movement disturbed by the inaccuracies of the linear track on the one hand, the robots own vibrations and the measurement inaccuracies on the other hand.
So if we sperate the theoretical path from the measurement results we get:

$$h_k = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} - s_k \tag{1}$$

with

$$s_k = k \cdot v \cdot dt \; , \; k = 0, 1, \ldots, N-1 \tag{2}$$

Taking this function h_k we make the discrete fourier transform on it and get the spectrum of the nonlinearities of the track. One special aspect not considered before is the spectral leakage effect of time delimited functions. Considering the finite number of measurement values taken in a finite time intervall, the number of the discrete function values of h_k is $N-1$ as can be seen in (1). In respect to this, there is a convolution with the spectrum of the rectengular function in the frequency domain.
So:

$$H(f_n) = \int_{-\infty}^{\infty} h(s) e^{-j2\pi f_n s} ds * G(rect_n) \tag{3}$$

and

$$G(rect_n) = T_p \cdot si(\pi f T_p) \; , \; T_p = \frac{1}{f_n} \tag{4}$$

This convolusion causes new spectral components which are not included in the measured signal whereby the original spectrum can not be analysed properly. The solution for this problem is finding a adequate window function, where the leakage effect is lower, so that the important spectral components can be detected properly.
The appropriate window function, a bartlett window was found in some special analyses and after some experiments in comparing different window functions on the measurement values the discrete fourier transform gets now an improved spectrum of the linear track scan depicted in figure 2:

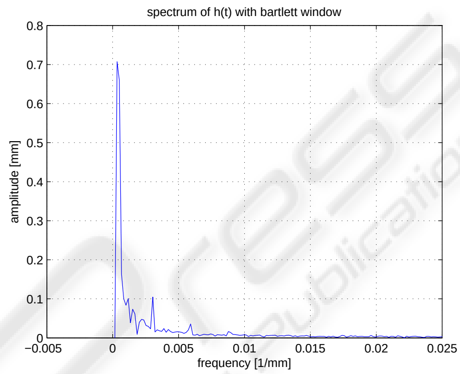

Figure 2: Spectrum of the track scan.

Due to the windowing three different frequency peaks can be seen clearly. The first peak with an amplitude of 0.7 mm at a frequency of 0.0003/mm represents the maximum error caused by the track which was measured and as you can see it is at the lowest frequency. The other two peaks at 0.003/mm and 0.006/mm are caused by the robots vibration and by the measurement system.
To get the needed number of sampling points for an identification of the linear track the new spectrum of the windowed function is used, considering the sampling theorem. With this new results the number of sampling points can be reduced by 25%.

3 POSITIONING ERRORS DUE TO SPLINE LINEARISATION

The benefit of the interpolation of the identified track coordinate systems via cubic splines is a continous description of the linear track in 6 dimensions $(x, y, z, \alpha, \beta, \gamma)$. As already mentioned the single track coordinate systems are identified in the sampling positions depending on the frequency spectrum after the continous track scan. So what we get after calculating the track coordinate systems is:

$$T_W^{R_i} = f_i(\alpha(l_i), \beta(l_i), \gamma(l_i), t_x(l_i), t_y(l_i), t_z(l_i)) \tag{5}$$

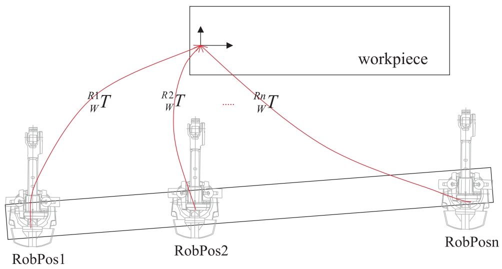

Figure 3: Track coordinate systems in sampling points.

This $T_W^{R_i}$ is depending on the sampling point position i. Now we make use of the cubic spline interpolation method and connect the discrete pairs of varieties (α_i, l_i), (β_i, l_i), etc. with smooth and also smooth in the first derivate functions, so that we can create track coordinate systems on every arbitrary position on the linear track.

$$T_W^R(l) = f_i(\alpha(l), \beta(l), \gamma(l), t_x(l), t_y(l), t_z(l)) \quad (6)$$

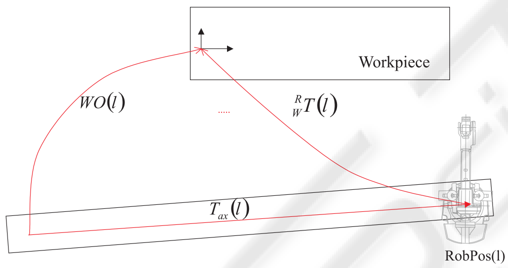

Figure 4: Track coordinate systems arbitrary.

Using the continous description of the linear track one robot programm can be modified and every single position can be corrected considering its corresponding linear track value. This method works quite good for programs in which the robot moves to static points. One crucial situation are linear movements, because for this the robot uses two programmed points and moves along the straight path between these two points. The ability to correct the first and the last point of this linear movement causes that the robot moves at the start and end position of its movement quite right, considering the errors given through the linear track. But whats about all the positions between the start and end point? Lets make a small example:

The two positions p_1 and p_2 as shown in figure 5 are

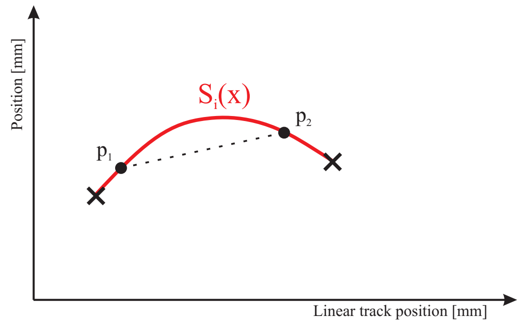

Figure 5: Programmed points between sampling positions.

programmed on a linear track position between two sampling points. Between this two sampling points there is the interpolation cubic spline function $S_i(x)$. Assuming that the robot moves from the point p_1 to the point p_2 it will calculate a straight line movement using the new corrected positions for p_1 and p_2. This calculation will effect that during the movement no correction can be processed because the splines are not directly integrated into the positional control of the robot. Mathematically the linearisation can be described as follows:

Taking the general equation for cubic spline in the range from one sampling position to the next:

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \quad (7)$$

in the intervall $I = [x_i, x_{i+1}]$. To find one spline function in the given interval there are four different parameters to identify $(a_i ... d_i)$. Due to the side conditions of the cubic spline interpolation there are for each unknown spline four different equations to find the four unknown parameters. As shown in (5) it is possible to transform this set of equations to a set of equation only depending on one parameter c_i so that:

$$d_i = \frac{1}{3(x_{i+1} - x_i)} c_{i-1} - c_i \quad (8)$$

$$b_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} - \frac{x_{i+1} - x_i}{3}(c_{i+1} + 2c_i) \quad (9)$$

and

$$a_i = y_i \quad (10)$$

with x_n as x-coordinate in the sampling points and therefore as linear track value at this positions and y_n as the corresponding y-coordinate of the single dimension $(x, y, z, \alpha, \beta, \gamma)$. The remaining c_i-values can be calculated through a linear equation using a tridiagonal, symmetric and positive Matrix:

$$\vec{c} = \overrightarrow{A^{-1}} \cdot \vec{a} \tag{11}$$

with

$$\vec{A} = \begin{pmatrix} 2(h_0+h_1) & h_1 & 0 & 0 & 0 \\ h_1 & 2(h_1+h_2) & h_2 & 0 & 0 \\ 0 & h_2 & 2(h_2+h_3) & h_3 & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & & & & 0 \\ 0 & 0 & 0 & h_{n-2} & 2(h_{n-2}+h_{n-1}) \end{pmatrix} \tag{12}$$

using

$$h_i = x_{i+1} - x_i \tag{13}$$

and

$$\vec{a} = \begin{pmatrix} 3\frac{y_2-y_1}{h_1} - 3\frac{y_1-y_0}{h_0} - h_0 c_0 \\ 3\frac{y_3-y_2}{h_2} - 3\frac{y_2-y_1}{h_1} \\ \vdots \\ 3\frac{y_n-y_{n-1}}{h_{n-1}} - 3\frac{y_{n-1}-y_{n-2}}{h_{n-2}} - h_{n-1} c_n \end{pmatrix} \tag{14}$$

Basing on this the prefactor d_1 of a spline between two arbitrary points can be calculated:

$$d_1 = \frac{6}{15h^3}\left[(y_1 - y_0) - 2(y_2 - y_1) + (y_3 - y_2)\right] \tag{15}$$

Special mathematical simulations have shown that the greatest deviation from a straight line between two points on the spline and the spline function itself is when the cubic part and therefore the prefactor d_i of the corresponding spline is zero. Basing on the symmetry in (15) this is the case for $y_2 = y_1$ and $(y_1 - y_0) - (y_3 - y_2) = 0$. Given this conditions the maximal deviation is:

$$\Delta s = S_i\left(\frac{b_i}{2c_i} + x_i\right) - y_1 \tag{16}$$

With practical meanings this is on a linear track with an inaccuracy of 0,5 mm in the heigth of its two rails an Δs-value of 0.325mm. That means, that if it would be possible to correct the robots position during the movement, the postioning of the TCP would be 0.325 mm better than without. Methods of correcting the robots TCP-motion dynamically during the movement were analysed and we expect to obtain Δs-values greater than 0.5mm.

4 ROBOT INTERFACE

Due to a new robot interface it is possible to use an uniform interface for variable sensor applications. For this the used sensors are not connected to the robot control via external interfaces but directly integrated in the robots programming language. This is done through a modular built program which is working on the robot control an which is supporting special functions for the user.

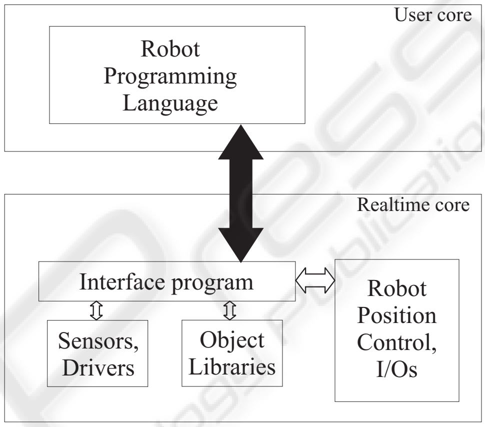

Figure 6: Robot interface for dynamic correction.

Using this new robot interface it is possible to send correction signals to the robot which are processed exactly in the interpolation rate of the robot. This allows to modify the TCP position with such a high time rate that the spline correction can be nearly perfectly used.

5 MEASUREMENT SYSTEM

The used measurement system for the needed measurement tasks like continous scan of one track motion, determining the particular track coordinate systems or identifying of the robots accuracy is represented by a Leica Laser Tracker LTD800 (figure 7). With this measurement device it is possible to do touchless measurements of 3-dimensional points in a range up to 80 meters. The measurement uncertainty of a coordinate is given by 10μm + 5μm/m with a possible maximum measurement rate of 3000 points per second.

Using the world's most accurate absolute distance meter (25μmm within 40m) and two built-in precision encoders for horizontal and vertikal angle measurements, it is a highly accurate measurement system and common in measurement tasks for aircraft and automobile industries.

Figure 7: Leica LTD800 Laser Tracker.

6 CONCLUSION

The dynamic correction of the robot during its movement is the basis for an optimal usage of the spline interpolation. In the first step (6) there was only a correction of single points and as can be seen, this has an disadvantage at controlled movements between two points.
With the improved method for highly integration of track motions this problem is solved and any movement can be corrected. Even the offline correction of robot programs is not needed anymore because the robot gets the corrections in the interpolation cycle of the motion control.
A future prospect will be the implementation of the spline correction in the new interface modul and therefore the correction of arbitrary movements.

REFERENCES

Bachman, G., Narici, L., Beckenstein, E. (2002). *Fourier and wavelet analysis*. Springer, New York.

Bracewell, R. (2000). *The fourier transform and its applications*. McGraw-Hill, Boston.

Convay, J. and Smith, D. (2003). *On quaternions and octonions*. A K Peters, Massachusetts.

Dautray, R. (2000). *Mathematical analysis and numerical methods for science and technology*. Springer, London.

Engeln-Mllges, G., Niederdrenk, K. and Wodicka, R. Numerik-Algorithmen. Springer, Berlin.

Kleinkes, M., Lilienthal, A., Neddermeyer, W. Highly Accurate Integraion of Track Motions. ICINCO Proceedings 2005, Barcelona.

Kleinkes, M., Lilienthal, A., Winkler, W. Static and dynamic accuracy and process ability tests for industrial robots. WMSCI Proceedings 2005, Orlando.

Maas, H. G. (1997). Dynamic photogrammetric calibration of industrial robots, spie's 42 annual meeting, san diego. In *Videometrics V, SPIE proceedings Series Vol. 3174*.

Nitschke, H. (2002). *Zur Bestimmung geometrischer Parameter von Industrierobotern*. Bayrische Akademie der Wissenschaften, Dissertation, TU Munich.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. F. (1989). *Numerical Recipes in C*. Cambridge University Press.

Ramanathan, J. (1998). *Methods of Applied Fourier Analysis*. Birkhuser, Boston.

Spong, Mark W., Vidyasagar, M. (1989). *Robot dynamics and control*. John Wiley and Sons, New York.