Speed Computation in Movement followed by Accurate Positioning of Industrial Robots

Liliana Marilena Matica
Faculty of Electrical Engineering and Information Technology,
University of Oradea,
Oradea, Romania
lmatica@uoradea.ro

Horea Oros
Faculty of Sciences,
University of Oradea,
Oradea, Romania
horos@uoradea.ro

***Abstract*— The paper describe a method of speed (velocity) computation, named mixt profile. The method assures an accurate positioning at the end of movement, in a well determinate time lapse; the method is linked with position vector computation of position matrix, about an industrial robot.**

Index Terms-cinematics of industrial robots; linear or circular trajectory; acceleration and deceleration stage of movement;

I. INTRODUCTION

The intelligent management of an industrial process involves the parameters adaptation of process commands. Concerning industrial robots, the motion programmable commands define the motion time or motion precision. Usually, if the motion time is defined, the positioning precision at the motion end is not very good. A very good positioning precision, at the motion end, can't be reach, in a defined motion time. Both conditions are very hard to accomplish. The mixt profile speed variation (described in this paper), during industrial robots motion, may accomplish this two conditions.

Concerning movement (motion) command of an industrial robot, [1; 2] it is necessary to define direct and inverse cinematics. A cinematics analysis example of an industrial robot is illustrated in fig.1. [3]

The notations in fig.1 define several Cartesian coordinate systems with its axles: X_i ; Y_i ; Z_i and its origins: O_i (index i goes from 1 to 6; i=1..6); then six rotation driving cinematics couples (d.c.c.) of industrial robot: C_i (i=1..6); variable parameters of d.c.c.: Θ_i (i=1..6) ; constant parameters of the industrial robot: d_1 ; a_2 ; d_4 ; d_6 ; the versors: $\vec{n}$; $\vec{o}$; $\vec{a}$ (about sense and direction of axles OX_6; OY_6 and OZ_6 (the coordinate system of index 6 has the origin in the tool point of the industrial robot and it has the position of axles according with the Denawitt-Hartenberg convention, about an industrial robot arm). [3]

During a motion, the speed profile (variation) is important for different reasons. Those reasons may be: inertial reasons; imposed time of motion; precision of motion; precision for reach the final motion point.

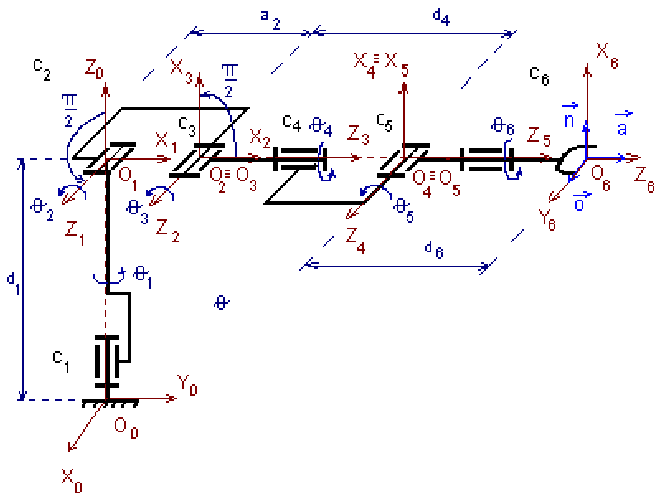

Fig.1. Cinematics analysis of an industrial robot, type RRRRRR.

II. THE POSITION VECTOR OF AN INDUSTRIAL ROBOT

The industrial robot position (location) is defined by the location matrix that contains position vector: $\vec{p}$ and orientation versors: $\vec{n}$; $\vec{o}$; $\vec{a}$; fig.2. [3]

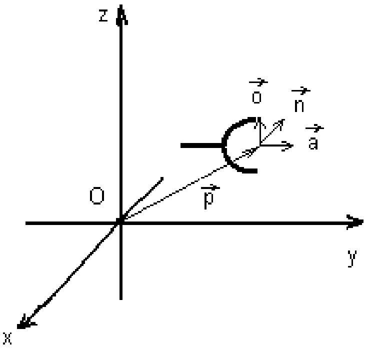

Fig.2. The position vector and orientations versors that define the location matrix of an industrial robot.

The vector $\vec{p}$ is the position of industrial robot tool point, relative to index *i* coordinate system. The location matrix of the industrial robot, about index *i* Cartesian coordinate systems, contains the three components (along the three axles) of those versors and vector, rel.1.

$$G_i = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

The orientation versors: $\vec{n}$; $\vec{o}$; $\vec{a}$ have the module equal to , its describe only the orientation, about the robot arm.

According to the Denawitt-Hartenberg convention, a coordinate system, index *i*, is obtained by homogeneous transformations, from previous one, index *i-1*. Those homogeneous transformations are (always in this order): [3]

1) *rotation of* $\theta_i(t)+\beta$ *angle, around* OZ_{i-1} *axle (the parameter t show that the angle varies in time, because that d.c.c. is an rotation one)*;
2) *translation of* d_i distance*, along* OZ_{i-1} *axle;*
3) *translation of* l_i distance, *along* OX_i *axle*;
4) *rotation of* α_i angle, *around* OZ_{i-1} *axle.*

Those homogeneous transformations define the transformation matrix, ${}^{i-1}A_i$: [3]

$$\begin{aligned} {}^{i-1}A_i &= Rot(OZ_{i-1}, \theta_i(t) + \beta_i) \cdot Trans(OZ_{i-1}, d_i) \cdot \\ &\cdot Trans(OX_i, l_i) \cdot Rot(OX_i, \alpha_i) \end{aligned} \quad (2)$$

The relationships between position matrix indexed *i* and position matrix indexed *i-1*, is:

$$G_{i-1} = {}^{i-1}A_i \cdot G_i \quad (3)$$

According with the Denawitt-Hartenberg convention, the matrix G_6 is the neutral matrix for multiplication; the relation between position matrix index 0 and index 6 is: [3]

$$\begin{aligned} G_o &= {}^0T_6 \cdot G_6 = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4 \cdot {}^4A_5 \cdot {}^5A_6 \cdot G_6 = \\ &= {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4 \cdot {}^4A_5 \cdot {}^5A_6 \end{aligned} \quad (4)$$

Considering the position matrix elements, it results the relationships for the direct cinematics of an industrial robot. The inverse cinematics (as a result of direct cinematics) compute the d.c.c. parameters starting with position matrix elements values. It result this conclusion: in purpose to command an industrial robot motion, it is necessary to compute the position matrix components, for every sampling period of time; those components describe the position and orientation of the industrial robot. The speed (velocity) of motion is defined by vector $\vec{p}$ (position vector) variation.

III. Acceleration, Motion on Trajectory and Deceleration

A linear or circular motion trajectory, for the industrial robot, contains several intermediary positions. The generation of trajectory is named interpolation process. If the trajectory type is not required, intermediary points may be determined according to other conditions; for example it ensure no jerking, in this situation the interpolation process is named joint interpolation (about an industrial robot). Another condition could be a certain programed speed. During acceleration, motion on trajectory and deceleration, intermediary positions of the industrial robot are defined by different location matrix.

About an industrial robot, if the trajectory is imposed (linear or circular), it is computed location matrices for intermediary points (named waypoints). Considering inverse cinematics, the commands for every d.c.c. of the industrial robot are computed, starting with every location matrix that composes the trajectory. For example, the formulas for compute position vector components of industrial robot type RRRRRR, fig.1, are (notations S_i; i=1..6, means sine of ϴi angle and C_i means cosine of same angle, the others notation are identical with those explained): [3]

$$\begin{aligned} p_x &= (S_1 \cdot C_2 \cdot S_3 + S_1 \cdot S_2 \cdot C_3) \cdot C_4 \cdot S_5 \cdot d_6 + \\ &+ C_1 \cdot S_4 \cdot S_5 \cdot d_6 - S_1 \cdot C_2 \cdot a_2 + \\ &+ (S_1 \cdot S_2 \cdot S_3 - S_1 \cdot C_2 \cdot C_3) \cdot (C_5 \cdot d_6 + d_4) \end{aligned} \quad (5)$$

$$\begin{aligned} p_y &= (-C_1 \cdot C_2 \cdot S_3 - C_1 \cdot S_2 \cdot C_3) \cdot C_4 \cdot S_5 \cdot d_6 + \\ &+ S_1 \cdot S_4 \cdot S_5 \cdot d_6 + + C_1 \cdot C_2 \cdot a_2 + \\ &+ (C_1 \cdot C_2 \cdot C_3 - C_1 \cdot C_2 \cdot S_3) \cdot (C_5 \cdot d_6 + d_4) \end{aligned} \quad (6)$$

$$\begin{aligned} p_z &= (-S_2 \cdot S_3 + C_2 \cdot C_3) \cdot C_4 \cdot S_5 \cdot d_6 + \\ &+ (S_2 \cdot C_3 + C_2 \cdot S_3) \cdot (C_5 \cdot d_6 + d_4) + S_2 \cdot a_2 + d_1 \end{aligned} \quad (7)$$

The motion of an industrial robot may contain three stages:

1) *the acceleration from zero motion speed to the programed motion speed;*
2) *the motion with programed motion speed (constant);*
3) *the deceleration from programed speed to zero.*

Commonly, acceleration to the programed speed depends on the speed profile that was selected, trapezoidal or parabolic, fig.3 and fig.4 (graphics consider continuous time).

This paper describes another method about acceleration and deceleration; the method describes another speed profile, named mixt profile of speed, fig.5.

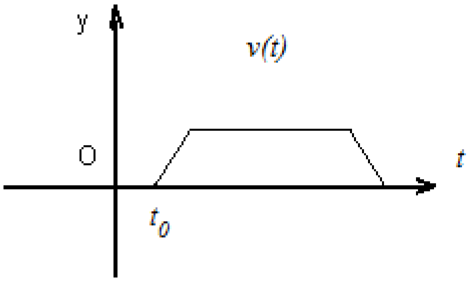

Fig.3. Trapezoidal profile (of speed).

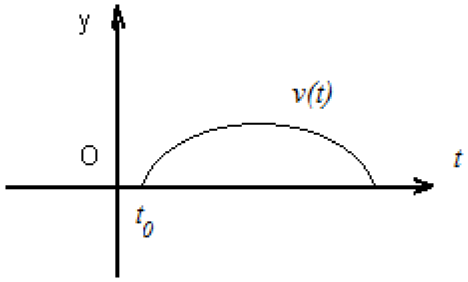

Fig.4. Parabolic profile.

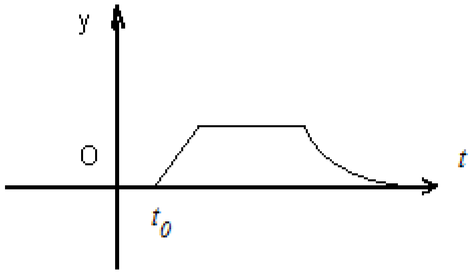

Fig.5. Mixt profile.

IV. Acceleration and Deceleration Stages for Mixt Profile of Velocity

The acceleration variation depends of the maximum acceleration possible, on a sample period of time (in a numerical computation system with numerical processor).

The numerical process of command computation, about an industrial robot motion, is a discrete one.[1] The variation of industrial robot position, motions speed, acceleration and deceleration values depend of a discrete variable defined by relation: $k \cdot T$, where T is the sampling period of time, and k is the number of the sample periods of time considered (for example, the variable had the value $10 \cdot T$ after ten sampling periods of time from the start of motion).

In the computation described in this paper, it is considered the value of maximum possible acceleration in a sample period of time, named a_{max} . About this new method, the acceleration stage for mixt speed profile, is described by the relation:

$$v(kT) = v_0 + k \cdot a_{\max} \tag{8}$$

The initial value of motion speed is zero, it results: $v_0 = 0$. After every sampling period of time, the position varies with the values: $T \cdot v(kT) = T \cdot k \cdot a_{\max}$; this variation may be computed for every component of the position vector, considering the maximum possible acceleration along every axle, named a_M (instead of a_{max}) $k \cdot T \cdot a_M$.

Considering the initial position of the industrial robot defined by those components of position vector: $p_{0,x}$; $p_{0,y}$; $p_{0,y}$. During the acceleration, after k sampling periods of time, the components of the position vector (indexed k), have the values:

$$p_{k,x} = p_{0,x} + k \cdot T \cdot a_M \tag{9}$$

$$p_{k,y} = p_{0,y} + k \cdot T \cdot a_M \tag{10}$$

$$p_{k,z} = p_{0,z} + k \cdot T \cdot a_M \tag{11}$$

Considering the maximum value of motion speed, named v_M , the acceleration ends when $v_M = k \cdot T \cdot a_M$, this special value of k, named k_A , may be computed. The maximum value of motion speed defines the number of sampling periods of time for the acceleration, named k_A:

$$k_A = v_M / T \cdot a_M \tag{12}$$

The value of k_A must be an integer value (the value must be adapted of this condition, it is the next bigger integer value of the computed value).

It is a different situation for a motion with a programed speed, named v_P . Considering its component $v_{P,x}$; $v_{P,y}$; $v_{P,z}$; the described method compute different values for each axle; the k_A value is determined by the maximum component (considering each component of programed speed, as previous described in rel.12):

$$k_A = \max(v_{P,x}; v_{P,y}; v_{P,z}) / T \cdot a_M \tag{13}$$

During the acceleration stage of motion, the position vector components are computed according with previous considerations:

$$p_{k,x} = p_{0,x} + k \cdot T \cdot (v_{P,x} - v_{0,x}) / k_A \quad , k = 1 \ldots k_A \tag{14}$$

$$p_{k,y} = p_{0,y} + k \cdot T \cdot (v_{P,y} - v_{0,y}) / k_A \tag{15}$$

$$p_{k,z} = p_{0,z} + k \cdot T \cdot (v_{P,z} - vz) / k_A \tag{16}$$

In the previous relation, k is an integer value and goes from 1 to k_A. The method may be applied for any initial values of motion speed. Usually, the components of initial speed: $v_{0,x}$; $v_{0,y}$; $v_{0,z}$ is zero.

It result the logical conclusion: the motion of the industrial robot, with a programed speed is described by the relations (where k starts from k_A and goes till is necessary the deceleration stage, defined by the value named k_D):

$$p_{k+1,x} = p_{k,x} + T \cdot v_{P,x}$$

$$p_{k+1,y} = p_{k,y} + T \cdot v_{P,y} \qquad (17)$$

$$p_{k+1,z} = p_{k,z} + T \cdot v_{P,z}$$

The previous relationships implement the $\delta_x, \delta_y, \delta_z$ mean the linear space steps that must be performed (executed movement), at every sampling period of time, on every axle, named axle step

The start of deceleration stage is defined by the value named k_D. The speed variation is described by the relationship, where a_D is the deceleration:

$$v(kT) = v_P - T \cdot a_D(kT) = v_P - b \cdot k^2 \ , k = 1 ... k_D \qquad (18)$$

In the previous relation, the value a_D is a variable value and b is a constant value adapted of desired characteristics about robot motion. The deceleration decreases till the motion end. Considering the condition: $0 = v_P - b \cdot {k_D}^2$ and components of speed for each axle, it results the number of sampling period of time for the deceleration, named k_D:

$$k_D = \sqrt{\frac{\max(v_{P,x}; v_{P,y}; v_{P,z})}{b}} \qquad (19)$$

The necessary distance for deceleration stage (named DD), determine the end of motion on trajectory (the motion on trajectory ends in the point situated at distance DD before the end point of motion.

The resulting speed profile is named mixt profile, fig.5 (the graphic considers continuous time). The described method ensures a better precision about stop point proximity. Typically, for a motion with precise positioning at the end about an industrial robot, it can't be specified the time needed; the mixt profile of speed specifies exactly the time needed for the motion with precise positioning at the end.

The described method, named mixt profile (of speed), was implemented at a flexible welding cellule (for manufacture of mining machinery), and the agreed motion characteristics (with the beneficiary) were ok.

V. Example of Computation

For example, considering a linear trajectory and constant orientation of robot arm (along the motion), considering the values of programed speed: $v_P = 5\sqrt{2}$ mm/s; $v_{P,x}$ = 3mm/s; $v_{P,y}$ = 4mm/s; $v_{P,z}$ = 5mm/s; a_M = 25mm/s^2 and T = 10^{-2} s; an example of computation determines $k_A = 20$ of sampling periods of time for acceleration process:

$$k_A = \max(3;4;5) mm/s / (10^{-2} s \cdot 25 mm/s^2) = 20 \qquad (20)$$

After the determination of k_A, the computation about the waypoints of the linear trajectory gets typical. Applying the difference analysing algorithm [3] (about motion on a linear trajectory), the speed for each axle differs (increase) with values: δv_x = 3/20 mm/s; δv_y = 4/20 mm/s; δv_z = 5/20 mm/s; for each sampling period of time. The position differs with values: $\delta p_x = T \cdot \delta v_x = 10^{-2} \cdot 3/20$ mm; $\delta p_y = T \cdot \delta v_y = 10^{-2} \cdot 4/20$mm; $\delta p_z = T \cdot \delta v_z = 10^{-2} \cdot 5/20$mm.

The number of sampling period of time for deceleration, considering b = 5mm/900s, is:

$$k_D = \sqrt{\frac{\max(3;4;5) mm/s}{5 mm/900s}} = 30 \qquad (21)$$

The computation of waypoints coordinates (during the deceleration) involves speed values:

$$v_x(kT) = v_{P,x} - \frac{v_{P,x}}{k_D^2} \cdot k^2 = 3 - \frac{3}{900} \cdot k^2 \ , k = 1 ... k_D$$

$$v_y(kT) = v_{P,y} - \frac{v_{P,y}}{k_D^2} \cdot k^2 = 4 - \frac{4}{900} \cdot k^2 \qquad (22)$$

$$v_z(kT) = v_{P,z} - \frac{v_{P,y}}{k_D^2} \cdot k^2 = 5 - \frac{5}{900} \cdot k^2$$

For each sampling period of time, the position differs (increase) with values, named axle steps (linear space steps that must be performed, at every sampling period of time, on every axle):

$$\delta p_x = T \cdot v_x(kT) \quad , k = 1 ... k_D$$

$$\delta p_y = T \cdot v_y(kT) \qquad (23)$$

$$\delta p_z = T \cdot v_z(kT)$$

For example, considering the initial values of position vector components: p_x = 1,1 mm; p_y = 2,2 mm; p_z = 3,3 mm; after 10 period of time the components are: $p_x = 1{,}1 + 10 \cdot 3/2000$ =1,115 mm; $p_y = 2{,}2 + 10 \cdot 4/2000$ = 2,22 mm; $p_z = 3{,}3 + 10 \cdot 5/2000$ = 3,325 mm; according to acceleration stage. After 20 period of time, begin the stage of motion on trajectory.

This stage of motion on trajectory is described by relations (similar with rel.23):

$$\delta p_x = T \cdot v_{P,x} \quad ; k = k_D..k_A$$

$$\delta p_y = T \cdot v_{P,y} \tag{24}$$

$$\delta p_z = T \cdot v_{P,z}$$

For each axle, the axle steps have constant values. The algorithm is named: numeric difference analysis, more exactly: linear numeric difference analysis algorithm.

The deceleration start is computed considering rel.19 and the distance for deceleration stage, named DD:

$$DD = \sum_{k=1}^{k_D} T \cdot [\max(v_{P,x}; v_{P,y}; v_{P,z}) - b \cdot k^2)] \tag{25}$$

The DD value considers the maximum distance, necessary for deceleration stage, of the three axles. The deceleration begins when it is distance DD, till the motion end, on respective axle.

The previous example considered a linear trajectory.

A circular trajectory imposes the computation of waypoints on spherical coordinates, named radium: R, polar angle: φ and azimuthal angle: ϕ ; (applying the circular difference analysing algorithm) and conversion on Cartesian coordinates of those values. The acceleration and deceleration is similar with the method described about a linear trajectory.

A variable orientation of robot arm (during the motion) involves the circular difference analysing algorithm, about computation of orientation versors: $\vec{n}$; $\vec{o}$; $\vec{a}$ components.

VI. Conclusions

About an industrial robots motion, those two conditions are very difficult to accomplish: best precision at the motion end and exact defined motion time. Those two conditions are accomplished by mixt profile (of motion speed variation), described in this paper.

The advantages of mixt profile are: the best precision to reach the end point of industrial robots motion, exact determination of motion time and minimum time of acceleration up to programed motion speed.

The method may have others diverse applications, about motion on a linear or circular trajectory; for example about turning or milling process.

Motion execution with exact speed gives processing quality; the described method of mixt profile, about speed variation, was implemented on numerical control equipment and precision was accurate (for example: numerical control equipment for workpieces of sintered metal carbides).

References

[1] T. Horsch, B.Juttler, Cartesian Spline Interpolation for Industrial Robots. University of Technology, Departament of mathematics, Darmstadt, Germany (http://www.ag.jku.at/pubs/csi98.pdf)

[2] L.M. Matica, Z. Kovendi, Structure Analysis for an Industrial Robot. Journal of Computer Science and Control Systems, Vol.4, Nr.1, Conference on Engineering of Modern Electric Systems 2011, Oradea, ISSN 1844-6043, pg.89, 2011

[3] L.M. Matica, Conducerea robotilor industriali. Edit. Univ. din Oradea, ISBN 978-973-759-481-5, 2008