Algorithms of the Manipulator Drive Control System

Bulat F. Bairamov, Artur A. Fardeev, Albert R. Fardeev
Kazan Federal University
Email: o.a.fardeeva@mail.ru

Received: 20th August 2019, Accepted: 30th September 2019, Published: 31st October 2019

Abstract
The article presents a developed algorithm for the manipulator drive control system using a processor. This algorithm consists of three parts:
- the functioning algorithm of the manipulator drive control system, in order to ensure the movement of the workpiece along a given path;
- the functioning algorithm of the manipulator drive control system, with an aim of eliminating the deviation of the true path of the manipulated object from the set;
- the functioning algorithm of the manipulator drive control system, ensuring the capture of the manipulated object.

Methods
Based on these models, it is necessary to create algorithms that ensure the operation of manipulator.

Introduction
When developing the functioning algorithm of the manipulator drive control system, it is necessary to use functional models that provide control for: the movement of the manipulated object along a given path; elimination of the deviation of the true trajectory from the set; bypassing obstacles and capturing the workpiece. These functional models have been previously developed [1]. Based on these models, it is necessary to create algorithms that ensure the operation of manipulator.

Algorithms of the Manipulator Drive Control System

1) input: \(a_{\text{nom}}, v_{\text{nom}}, l_1, l_2, l_3\)
2) assignment: \(t_0 = t; t_p = \frac{v_{\text{nom}}}{\nu_{\text{nom}}}; t_T = \max (l_1, l_2, l_3); T = \frac{l}{v_{\text{nom}}} + \frac{1}{2}t_p + \frac{1}{2}t_T; k_p = \frac{t_p}{T}; k_T = \frac{t_T}{T};\)
3) verification of condition fulfillment: \(t - t_0 \leq t_p;\)
   if the condition is met, go to step 4, 5, then to 3;
   if the condition is not met, then go to step 6;
4) assignment: \(a_1 = \frac{l_1}{r^2k_p(1-\frac{1}{2}(k_p+k_T))}; a_2 = \frac{l_2}{r^2k_p(1-\frac{1}{2}(k_p+k_T))}; a_3 = \frac{l_3}{r^2k_p(1-\frac{1}{2}(k_p+k_T))};\)
5) Report acceleration \(a_1, a_2, a_3\) using appropriate controls
6) verification of condition fulfillment: \(t - t_0 \leq T - t_p;\)
   if the condition is met, go to step 7, 8, then to 6;
   if the condition is not met, then go to step 9;
7) assignment: \(a_1 = 0; a_2 = 0; a_3 = 0; v_2 = \frac{l_1}{T(1-\frac{1}{2}(k_p+k_T))}; v_3 = \frac{l_1}{T(1-\frac{1}{2}(k_p+k_T))};\)
8) Report acceleration \(a_1, a_2, a_3\) and speed \(v_1, v_2, v_3\) using appropriate controls
9) verification of condition fulfillment: \(t - t_0 \leq T;\)
if the condition is met, go to step 10, 11, then to 9;
if the condition is not met, then go to step 12;
10) assignment: 
\[ a_1: = \frac{i_1}{T^2k_1(1-\frac{1}{2}\nu_0k_1^n)}; \quad a_2: = \frac{i_2}{r^2k_1(1-\frac{1}{2}\nu_0k_1^n)} \]

11) Report acceleration \( a_1, a_2, a_3 \) using appropriate controls
12) assignment: \( a_1: = 0; a_2: = 0; a_3: = 0; u_1: = 0; u_2: = 0; u_3: = 0 \)
13) Report acceleration \( a_1, a_2, a_3 \) and speed \( v_1, v_2, v_3 \) using appropriate controls

Where \( a_{nom}, v_{nom} \) – rated acceleration and speed of processor controls (controller); \( i_1, i_2, i_3 \) – projections of the trajectory onto the corresponding coordinate axes; \( \nu_0 \) – initial time; \( \nu \) – current time; \( \nu_p \) – acceleration time; \( \nu_r \) – braking time; \( T \) – time of the manipulated object movement from the start point to the end; \( k_0, k_1 \) – acceleration and braking factors; \( a_1, a_2, a_3 \) – acceleration projections on the corresponding coordinate axes; \( v_1, v_2, v_3 \) – speed projections on the corresponding coordinate axes.

Formulas for calculating such quantities as \( T, k_0, k_1 \), as well as projections of speed and acceleration vectors at the stages of acceleration, motion and braking, used in this algorithm, were obtained earlier [3].
This algorithm is designed for a straight path. Straightness is achieved due to the same acceleration, braking and movement time of the manipulated object across all degrees of mobility [4], as well as due to constant speeds across all degrees of mobility in the movement section [5].
If the presence of an obstacle or process conditions do not allow moving along a straight path, then one can use a broken path with its own speeds and accelerations in each straight section [6].

**Results and Discussion**

To create the manipulator drive control program that ensures the workpiece returns to a given path when the true path deviates from the given path based on the functional model previously developed [1], we will draw up an algorithm for the system functioning in solving this problem. We will use the method of successive adjustments [2].
The resulting algorithm has the following form:

1) input:

\[
\Delta S, a_{nom}, v_{nom}, x, \left( \frac{T}{\nu_0} \right), y, \left( \frac{T}{\nu_0} \right), z, \left( \frac{T}{\nu_0} \right), \beta_2, \alpha_2, \alpha_2, \alpha_2
\]

2) assignment: 
\[ x_n \left( \frac{T}{\nu_0} \right): = \frac{i_1}{\nu_0} = \Delta S; \quad y_n \left( \frac{T}{\nu_0} \right): = \frac{i_2}{\nu_0} + \Delta S; \]
\[ z_n \left( \frac{T}{\nu_0} \right): = \frac{i_3}{\nu_0} + \Delta S; \quad x_\beta \left( \frac{T}{\nu_0} \right): = \frac{i_3}{\nu_0} + \Delta S; \]
\[ i = 1, n; l_{110} = \frac{l_1}{n}; l_{120} = \frac{l_2}{n}; l_{130} = \frac{l_3}{n}; l_0: = \frac{i}{n} \max(l_{110}, l_{120}, l_{130}); \]
\[ t_p: = \frac{v_{nom}}{a_{nom}}; t_{p1}: = t_p; t_1: = t_p + \frac{l_0}{2} + \frac{l_1}{2} + \frac{l_2}{2} + \frac{l_3}{2}; \]
\[ k_{p1}: = \frac{t_{p1}}{t_1}; \]

3) assignment: 
\[ t_0: = t \]
4) Value update \( x(t - t_0), y(t - t_0), z(t - t_0) \)
5) verification of condition fulfillment: \( x \left( \frac{T}{\nu_0} \right) - x_n \left( \frac{T}{\nu_0} \right) < 0 \)
if the condition is met, go to step 6, then bypassing 7 to 8;
if the condition is not met, then go to step 7;
6) assignment: 
\[ l_{11}: = x_n \left( \frac{T}{\nu_0} \right) - x \left( \frac{T}{\nu_0} \right); \quad a_{11}: = \frac{i_{11}}{t_1^2k_{p1}(1-\nu_0k_{p1}^n)} \]
\[ a_{12}: = -a_{11}; \quad v_{11}: = v_1 + \frac{i_{11}}{t_1^2k_{p1}(1-\nu_0k_{p1}^n)}; \quad v_{01}: = v_1 \]
7) assignment: \( a_{12}: = 0; a_{12}: = 0; u_{11}: = u_1; u_{01}: = u_1 \)
8) verification of condition fulfillment: \( y \left( \frac{T}{\nu_0} \right) - y_n \left( \frac{T}{\nu_0} \right) < 0 \)
if the condition is met, go to step 9, then bypassing 10 to 11;
if the condition is not met, then go to step 10;
9) assignment: 
\[ l_{12}: = y_n \left( \frac{T}{\nu_0} \right) - y \left( \frac{T}{\nu_0} \right); \quad a_{22}: = \frac{i_{12}}{t_1^2k_{p1}(1-\nu_0k_{p1}^n)} \]

5644 © 2019 The Author (s); Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495
\[a_{2k} = -a_{2n}; \quad v_{12} = v_2 + \frac{l_{12}}{\tau_1(1 - \frac{1}{2}(k_p + k_t))}; \quad v_{02} = v_2\]

10) assignment: \(a_{2n} = 0; \quad a_{2k} = 0; \quad v_{12} = v_2; \quad v_{02} = v_2\)

11) verification of condition fulfillment: \(z\left(\frac{x_1}{n}\right) - z\left(\frac{x_2}{n}\right) < 0\)
if the condition is met, go to step 12, then bypassing 13 to 14;
if the condition is not met, then go to step 13;

12) assignment: \(l_{13} = z\left(\frac{x_1}{n}\right) - z\left(\frac{x_2}{n}\right); \quad a_{3n} = \frac{l_{13}}{\tau_1^2k_p(1 - \frac{1}{2}(k_p + k_t))}; \quad a_{3k} = -a_{3n}\)

\[v_{13} = v_3 + \frac{l_{13}}{\tau_1(1 - \frac{1}{2}(k_p + k_t))}; \quad v_{03} = v_3\]

13) assignment: \(a_{3n} = 0; \quad a_{3k} = 0; \quad v_{13} = v_3; \quad v_{03} = v_3\)

14) verification of condition fulfillment: \(x\left(\frac{x_1}{n}\right) - x\left(\frac{x_2}{n}\right) > 0\)
if the condition is met, go to step 15, then bypassing 16 to 17;
if the condition is not met, then go to step 16;

15) assignment: \(l_{14} = x\left(\frac{x_1}{n}\right) - x\left(\frac{x_2}{n}\right); \quad a_{1n} = -\frac{l_{14}}{\tau_1^2k_t_1(1 - \frac{1}{2}(k_p + k_t_1))}; \quad a_{1k} = -a_{1n}; \quad v_{14} = v_1 - \frac{l_{14}}{\tau_1(1 - \frac{1}{2}(k_p + k_t_1))}; \quad v_{04} = v_1\)

16) assignment: \(a_{1n} = 0; \quad a_{1k} = 0; \quad v_{14} = v_1; \quad v_{04} = v_1\)

17) verification of condition fulfillment: \(y\left(\frac{x_1}{n}\right) - y\left(\frac{x_2}{n}\right) > 0\)
if the condition is met, go to step 18, then bypassing 19 to 20;
if the condition is not met, then go to step 19;

18) assignment: \(l_{12} = y\left(\frac{x_1}{n}\right) - y\left(\frac{x_2}{n}\right); \quad a_{2n} = -\frac{l_{12}}{\tau_1^2k_t_1(1 - \frac{1}{2}(k_p + k_t_1))}; \quad a_{2k} = -a_{2n}; \quad v_{12} = v_2 - \frac{l_{12}}{\tau_1(1 - \frac{1}{2}(k_p + k_t_1))}; \quad v_{02} = v_2\)

19) assignment: \(a_{2n} = 0; \quad a_{2k} = 0; \quad v_{12} = v_2; \quad v_{02} = v_2\)

20) verification of condition fulfillment: \(z\left(\frac{x_1}{n}\right) - z\left(\frac{x_2}{n}\right) > 0\)
if the condition is met, go to step 21, then bypassing 22 to 23;
if the condition is not met, then go to step 22;

21) assignment: \(l_{13} = z\left(\frac{x_1}{n}\right) - z\left(\frac{x_2}{n}\right); \quad a_{3n} = -\frac{l_{13}}{\tau_1^2k_t_1(1 - \frac{1}{2}(k_p + k_t_1))}; \quad a_{3k} = -a_{3n}\)

\[v_{13} = v_3 - \frac{l_{13}}{\tau_1(1 - \frac{1}{2}(k_p + k_t_1))}; \quad v_{03} = v_3\]

22) assignment: \(a_{3n} = 0; \quad a_{3k} = 0; \quad v_{13} = v_3; \quad v_{03} = v_3\)

23) verification of condition fulfillment: \(t - t_0 \leq t_{p1}\)
if the condition is met, go to steps 24, 25, then to 23;
if the condition is not met, then go to step 26;

24) assignment: \(a_0 = 0; \quad a_{1k} = a_{2n}; \quad a_2 = a_{2n}; \quad a_3 = a_{3n}\)

25) Report acceleration \(a_1, \quad a_2, \quad a_3\) using appropriate controls

26) verification of condition fulfillment: \(t - t_0 \leq T_1 - t_{p1}\)
if the condition is met, go to steps 27, 28, then to 26;
if the condition is not met, then go to step 29;

27) \(a_1 = 0; \quad a_2 = 0; \quad a_3 = 0; \quad v_{11} = v_1; \quad v_{12} = v_2; \quad v_{03} = v_{13}\)

28) Report acceleration \(a_1, \quad a_2, \quad a_3\) and speed \(v_1, v_2, v_3\) using appropriate controls

29) verification of condition fulfillment: \(t - t_0 \leq T_1\)
if the condition is met, go to steps 30, 31, then to 29;
if the condition is not met, then go to step 32;

30) assignment: \(a_0 = a_{1k}; \quad a_2 = a_{2k}; \quad a_3 = a_{3k}\)

31) Report acceleration \(a_1, \quad a_2, \quad a_3\) using appropriate controls

32) assignment: \(a_1 = 0; \quad a_2 = 0; \quad a_3 = 0; \quad v_{10} = v_{01}; \quad v_{21} = v_{02}; \quad v_{23} = v_{03}\)

33) Report acceleration \(a_1, \quad a_2, \quad a_3\) and speed \(v_1, v_2, v_3\) using appropriate controls

34) verification of condition fulfillment: \(i < n\)
if the condition is met, then go to step 3;
if the condition is not met, then no parameters are changed

Where \(\Delta S\) – permissible deviation in each coordinate of the true trajectory from the given; \(\alpha_{nom}, v_{nom}\) – rated acceleration and speed of processor controls (controller) [7]; \(x_3\left(\frac{x_1}{n}\right), y_3\left(\frac{x_1}{n}\right), z_3\left(\frac{x_1}{n}\right)\) – coordinate values of a given
The resulting algorithm has the following form:

Algorithm for the system functioning in fulfilling the specified purpose.

To create the manipulator drive control program that provides capture of the manipulation object on the basis of the technical vision previously developed [1], we will draw up an algorithm for the system functioning in fulfilling the specified purpose. The resulting algorithm has the following form:\n
1) input: $\phi_{10}, \phi_{20}, \phi_{30}, \phi_{1m}, \phi_{2m}, \phi_{3m}, x_{n0}, y_{n0}, x_{n1}, y_{n1}, v_{nom}, \alpha_{nom}, \Delta t$

2) verification of condition fulfillment: $x_{n0} - x_n < 0$

   if the condition is met, then go to steps 3, 4, then bypassing 5, 6, 7 to 8;

   if the condition is not met, then go to steps 5, 6, 7, then return to 2

   3) assignment: $v_1 := 0$

   4) report speed $v_1$ to the appropriate controls

   5) assignment: $v_1 := v_{nom}$

   6) report speed $v_1$ to the appropriate controls

   7) value update $x_{n0}, y_{n0}, x_{n1}, y_{n1}, z_n$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision

   8) verification of condition fulfillment: $y_{n0} - y_n < 0$

   if the condition is met, then go to steps 9, 10, then bypassing 11, 12, 13 to 14;

   if the condition is not met, then go to steps 11, 12, 13, then return to 8

   9) assignment: $v_2 := 0$

   10) report speed $v_2$ to the appropriate controls

    11) assignment: $v_2 := v_{nom}$

    12) report speed $v_2$ to the appropriate controls

    13) value update $x_{n0}, y_{n0}, z_n, x_{n1}, y_{n1}, z_n$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision

    14) verification of condition fulfillment: $z_{n0} - z_n < 0$

    if the condition is met, then go to steps 15, 16, then bypassing 17, 18, 19 to 20;

    if the condition is not met, then go to steps 17, 18, 19, then return to 14

    15) assignment: $v_3 := 0$

    16) report speed $v_3$ to the appropriate controls

    17) assignment: $v_3 := v_{nom}$

    18) report speed $v_3$ to the appropriate controls

    19) value update $x_{n0}, y_{n0}, z_n, x_{n1}, y_{n1}, z_n$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision

    20) verification of condition fulfillment: $x_{n0} - x_n > 0$

    if the condition is met, then go to steps 21, 22, then bypassing 23, 24, 25 to 26;

    if the condition is not met, then go to steps 23, 24, 25, then return to 20

    21) assignment: $v_1 := 0$

    22) report speed $v_1$ to the appropriate controls

    23) assignment: $v_1 := -v_{nom}$

    24) report speed $v_1$ to the appropriate controls

    25) value update $x_{n0}, y_{n0}, z_n, x_{n1}, y_{n1}, z_n$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision

    26) verification of condition fulfillment: $y_{n0} - y_n > 0$
if the condition is met, then go to steps 27, 28, then bypassing 29, 30, 31 to 32.
if the condition is not met, then go to steps 29, 30, 31, then return to 26
27) assignment: $v_2 := 0$
28) report speed $v_2$ to the appropriate controls
29) assignment: $v_2 := -v_\text{nom}$
30) report speed $v_2$ to the appropriate controls
31) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision
32) verification of condition fulfillment: $z_{b_0} - z_b > 0$
if the condition is met, then go to steps 33, 34, then bypassing 35, 36, 37 to 38;
if the condition is not met, then go to steps 35, 36, 37, then return to 32
33) assignment: $v_2 := 0$
34) report speed $v_2$ to the appropriate controls
35) assignment: $v_2 := -v_\text{nom}$
36) report speed $v_2$ to the appropriate controls
37) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision
38) verification of condition fulfillment: $\phi_{2n} - \phi_2 < 0$
if the condition is met, then go to steps 33, 34, then bypassing 35, 36, 37 to 38;
if the condition is not met, then go to steps 5, 6, 7, then return to 2
39) assignment: $\omega_2 := 0$
40) report speed $\omega_2$ to the appropriate controls
41) assignment: $\omega_2 := \omega_\text{nom}$
42) report speed $\omega_2$ to the appropriate controls
43) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision
44) verification of condition fulfillment: $\phi_{2n} - \phi_2 < 0$
if the condition is met, then go to steps 45, 46, then bypassing 47, 48, 49 to 50;
if the condition is not met, then go to steps 47, 48, 49, then return to 44
45) assignment: $\omega_2 := 0$
46) report speed $\omega_2$ to the appropriate controls
47) assignment: $\omega_2 := \omega_\text{nom}$
48) report speed $\omega_2$ to the appropriate controls
49) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision
50) verification of condition fulfillment: $\phi_{3n} - \phi_3 < 0$
if the condition is met, then go to steps 51, 52, then bypassing 53, 54, 55 to 56;
if the condition is not met, then go to steps 53, 54, 55, then return to 50
51) assignment: $\omega_3 := 0$
52) report speed $\omega_3$ to the appropriate controls
53) assignment: $\omega_3 := \omega_\text{nom}$
54) report speed $\omega_3$ to the appropriate controls
55) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision
56) verification of condition fulfillment: $\phi_{1b} - \phi_1 > 0$
if the condition is met, then go to steps 57, 58, then bypassing 59, 60, 61 to 62;
if the condition is not met, then go to steps 59, 60, 61, then return to 56
57) assignment: $\omega_1 := 0$
58) report speed $\omega_1$ to the appropriate controls
59) assignment: $\omega_1 := -\omega_\text{nom}$
60) report speed $\omega_1$ to the appropriate controls
61) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision
62) verification of condition fulfillment: $\phi_{2b} - \phi_2 > 0$
if the condition is met, then go to steps 63, 64, then bypassing 65, 66, 67 to 68;
if the condition is not met, then go to steps 65, 66, 67, then return to 62
63) assignment: $\omega_2 := 0$
64) report speed $\omega_2$ to the appropriate controls
65) assignment: $\omega_2 := -\omega_\text{nom}$
66) report speed $\omega_2$ to the appropriate controls
67) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision

68) verification of condition fulfillment: $\phi_3 - \phi_1 > 0$
if the condition is met, then go to steps 69, 70, then bypassing 71, 72, 73 to 74;
if the condition is not met, then go to steps 71, 72, 73, then return to 68

69) assignment: $\omega_3 = 0$

70) report speed $\omega_3$ to the appropriate controls

71) assignment: $\omega_2 = -\omega_{nom}$

72) report speed $\omega_3$ to the appropriate controls

73) value update $x_n, y_n, z_n, x_w, y_w, z_w$ according to the additional locator and value update $\phi_1, \phi_2, \phi_3$ according to technical vision

74) assignment: $u_4 = u_{nom}; \ t_0 = t$

75) report speed $u_4$ to the appropriate controls

76) verification of condition fulfillment: $t - t_0 \geq \Delta t$
if the condition is met, then go to step 78, 78
if the condition is not met, then go to step 76

77) assignment: $u_4 = 0$

78) Report speed $u_4$ to the appropriate controls

Where $\phi_1, \phi_2, \phi_3$ – lower and upper boundaries of acceptable values of rotation angles of the workpiece relative to the coordinate axes before its capture; $x_{nom}, y_{nom}, z_{nom}$, $x_{nom}, y_{nom}, z_{nom}$ – lower and upper boundaries of the area on each coordinate axis, in which the workpiece shall be located before it is captured; $x_{w}, y_{w}, z_{w}, x_{w}, y_{w}, z_{w}$ – lower and upper boundaries of the area along each coordinate axis, in which the workpiece is located at a given time; $\phi_1, \phi_2, \phi_3$ – true values of rotation angles of the workpiece relative to the coordinate axes; $\omega_{nom}$ – rated speed of progressive controls; $\omega_{nom}$ – rated speed of rotational controls; $u_1, u_2, u_3$ – speed of controls for portable degrees of mobility; $\omega_1, \omega_2, \omega_3$ – speed of controls, orienting rotational degrees of mobility; $u_4$ – speed of controls, orienting translational degree of mobility; $\Delta t$ – time required for the workpiece capture.

In this algorithm, giving speed $u_4 = u_{nom}$ the moving jaw of the gripper, and then zeroing this speed after a certain time period $\Delta t$ is the process of workpiece capturing.

Based on this algorithm, a drive control program can be compiled to capture the manipulated object.

Conclusions
The following algorithms have been developed: functioning of the manipulator drive control system using the processor in order to ensure the workpiece movement along a predetermined path, in order to eliminate the deviation of the true path of the manipulated object from the predetermined one and ensure the capture of the manipulated object.

The following algorithms were used at their creation: control method by speed vector, method of sequential adjustments. The last algorithm is based on a change in the grasp position when the generalized coordinates of the workpiece (axial coordinates and rotation angles) go beyond the allowable area.

These algorithms can be used to create the manipulator control programs.

Acknowledgements
The work is performed according to the Russian Government Program of Competitive Growth of Kazan Federal University.

References
