Seventh International Conference on Information Science and Technology
Da Nang, Vietnam; April 16-19, 2017

Sorting System Algorithms Based on Dynamic Picking for Delta Robot

Tingting Su, Haojian Zhang, Yunkuan Wang, Shaohong Wu and Jun Zheng
Institute of Automation, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Beijing, China
sutingting2013@ia.ac.cn

Jianzhang Chang and Hongsheng Sun
Bohai Shipyard Group Co., Ltd
Huludao, China

Shuhai Zhang
Huludao Bohai Mechanical Engineering Co., Ltd
Huludao, China

Abstract—**In this paper, different conditions of pick-and-place trajectory are analyzed, and on this basis, the dynamic picking algorithm for the workpieces on the conveyor belt is proposed by using the Ferrari's method. Then, the trajectory planning algorithm of sorting multiple types of workpieces is proposed by analyzing the trajectory planning model of two types of workpieces. The workpiece to be sorted is determined first, and then will be picked up at the picking point calculated by dynamic picking algorithm. Experiments show that the dynamic picking algorithm is accurate and fast; does not need to set the initial value and is suitable for practical engineering applications. And the algorithm of trajectory planning for multiple types of workpieces has high accuracy and stability.**

Keywords—delta robot; dynamic picking; Ferrari's method; industrial sorting; trajectory planning

I. INTRODUCTION

Delta robot [1-3] is one of the most successful parallel mechanisms. It is a type of parallel robot [4-5] with high velocity and light load, and usually has 3 to 4 degrees of freedom. Because of its light weight, small size, fast speed, accurate positioning, low cost and high efficiency, it is widely used in production lines such as rapid sorting, grabbing and assembly of food, medicine and electronic products [6], and especially suitable for the fast identification and grasping operation of the scattered workpieces on the conveyer belt. For the working environment where the starting point and target point are fixed, we can use the teaching method or off-line programming method to generate the trajectory to complete the task [7]; for the complex working environment where starting point or target point are uncertain, we can use sensors such as visual sensors to detect the target, and then calculate the physical coordinates of the target in order to achieve the sorting or other purposes. Conventional industrial production lines usually sort a specific kind of workpiece, which can not meet the needs of sorting multiple types of workpieces. At present, there are few studies on the sorting of different types of workpieces with delta robots. Ni Hepeng [8] established the non-linear mathematical model for workpiece tracking which was based on geometric analysis, and proposed a dynamic picking algorithm based on Newton-Raphson method in order to improve sorting efficiency. This algorithm was easy to comprehend, but the Newton-Raphson iterative algorithm should obtain the approximate solution, and should be set the initial value. Zhang Wenchang [9] proposed a method based on servomotor plus synchronous-conveyor for multiple objects' tracking and used Newton's dichotomy method for object's grasping position calculation. However, this method was difficult to achieve and can't meet the production process that have the requirements for velocity of the conveyor belt. Xie Zexiao [10] proposed a method of trajectory planning for high-velocity delta parallel robots in pick-and-place operations. The transitions between the vertical and horizontal segments of the operation trajectory were smoothed with Lamé curves. The method effectively reduced the residual vibration of the mechanism and improved the accuracy of placement, but the calculation was complex and the time performance was relatively poor.

In this paper, different conditions of pick-and-place trajectory are analyzed. On this basis, the dynamic picking algorithm for the workpieces which are on the conveyor belt at a uniform velocity is studied, and Ferrari's method is used to calculate the picking point. Then, the trajectory planning algorithm of sorting multiple types of workpieces is proposed by analyzing the trajectory planning model of two types of workpieces.

II. DYNAMIC PICKING ALGORITHM BASED ON FERRARI METHOD

A. Problem Background

The vision sensor obtains the coordinates of the workpiece in the image coordinate system, and gets the category information by image recognition. The coordinates of the workpiece in robot coordinate system are obtained from the image coordinate and the perspective projection equation. Then the workpiece will be picked up based on the coordinate and category information. In order to meet the need of production lines and improve production efficiency, the robot system can dynamically track the target workpiece and then pick it.

B. Robot Motion Control Method

When the robot moves at a high velocity, the acceleration of the end effector is big, and it is prone to shock, vibration and overshoot. The arm is slender and easy to tremble, which seriously affects the accuracy and life of the robot. Therefore, flexible acceleration and deceleration control method must be used. Linear acceleration and deceleration control method, linear plus parabolic acceleration and deceleration control method, and exponential acceleration and deceleration control method are not suitable for the control of high-velocity delta robots because of the abrupt and discontinuous acceleration at the initial and final stages. In this paper, the modified trapezoidal acceleration and deceleration control method is chosen, because its acceleration and deceleration are more compliant and the calculation is simpler [11].

Acceleration of the modified trapezoidal acceleration and deceleration control method can be described as follows:

$$a=\begin{cases} a_{\max}\sin(\frac{4\pi}{T}t), & 0\le t\le\frac{1}{8}T \\ a_{\max}, & \frac{1}{8}T<t\le\frac{3}{8}T \\ a_{\max}\cos[\frac{4\pi}{T}(t-\frac{3}{8}T)], & \frac{3}{8}T<t\le\frac{5}{8}T \\ -a_{\max}, & \frac{5}{8}T<t\le\frac{7}{8}T \\ -a_{\max}\cos[\frac{4\pi}{T}(t-\frac{7}{8}T)], & \frac{7}{8}T<t\le T \end{cases} \tag{1}$$

The displacement can be obtained by integrating the time of (1) twice and combining the boundary conditions.

It is assumed that the maximum velocity in the above formula does not exceed the velocity limit of the end-effector. The above formula is integrated once, and we can see the maximum velocity v appearing at time $T/2$. The velocity of this moment is:

$$v=\sqrt{S(\frac{1}{\pi}+\frac{1}{2})a_{\max}} \tag{2}$$

The maximum acceleration of the robot end-effector is $a_{\max}$. The upper limit of the velocity of the end-effector is $v_{\lim}$. In order to make the velocity during all the movement not exceed the upper limit of the velocity of the end-effector $v_{\lim}$, S should satisfy:

$$S\le\frac{v_{\lim}^2}{(\frac{1}{\pi}+\frac{1}{2})a_{\max}} \tag{3}$$

C. Path Planning Algorithm

Pick-and-Place Operations (PPO) [12] is the main mode of operations for parallel robots, and the Adept motion curve [13] is commonly used in Pick-and-Place Operations for Delta robot, as shown in Fig. 1.

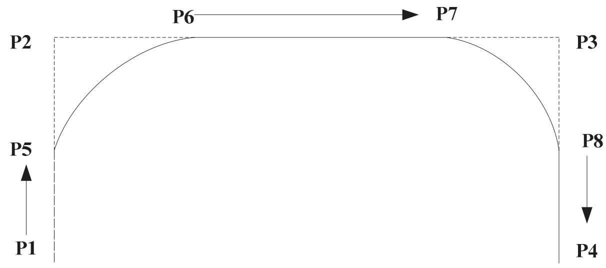

Fig. 1. the Adept motion curve

The modified trapezoidal acceleration and deceleration control method is separately used in the horizontal path P_2P_3 and vertical paths P_1P_2 and P_3P_4 of the Adept motion curve. Assume: T_1 is the movement time of P_1 to P_6, T_2 is the movement time of P_5 to P_8, T_3 is the movement time of P_7 to P_4, the distance between P_1 and P_2 is S_1, the distance between P_3 and P_4 is S_3, and the distance between P_2 and P_3 is S_2.

1) When $S_1\le S_2$ and $S_3<S_2$: when $t=0$, the delta robot starts to move upward in the vertical direction; when $t=T_1/2$, the delta robot starts to move in the horizontal direction; when $t=T_1/2+T_2-T_3/2$, the delta robot starts to move downward in the vertical direction. Therefore the time that the robot moves from the current position P_1 to the target point P_4 by the Adept motion curve is :

$$\begin{aligned} t&=T_1/2+T_2+T_3/2 \\ &=\sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Bigg/2+\sqrt{\frac{S_2}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} \\ &+\sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Bigg/2 \end{aligned} \tag{4}$$

2) When $S_1>S_2$ and $S_3\ge S_2$: when $t=0$, the delta robot starts to move upward in the vertical direction; when $t=T_1-T_2/2$, the delta robot starts to move in the horizontal direction; when $t=T_1$, the delta robot starts to move downward in the vertical direction. In this case, P_6 and

This work is supported by Major Science and Technology Project of Henan Province under Grant No. 161100210300.

P_7 coincide. So the time that the robot moves from P_1 to P_4 is :

$$t = T_1 + T_3 = \sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} + \sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} \tag{5}$$

3) When $S_1 \le S_2$ and $S_3 \ge S_2$: when $t=0$, the delta robot starts to move upward in the vertical direction; when $t = T_1/2$, the delta robot starts to move in the horizontal direction; when $t = T_1/2 + T_2/2$, the delta robot starts to move downward in the vertical direction. So the time that the robot moves from P_1 to P_4 is :

$$t = T_1/2 + T_2/2 + T_3 = \sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 + \sqrt{\frac{S_2}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 + \sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} \tag{6}$$

4) When $S_1 > S_2$ and $S_3 < S_2$: when $t=0$, the delta robot starts to move upward in the vertical direction; when $t = T_1 - T_2/2$, the delta robot starts to move in the horizontal direction; when $t = T_1 + T_2/2 - T_3/2$, the delta robot starts to move downward in the vertical direction. So the time that the robot moves from P_1 to P_4 is :

$$t = T_1 + T_2/2 + T_3/2 = \sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} + \sqrt{\frac{S_2}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 + \sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 \tag{7}$$

D. Dynamic Picking Algorithm

Dynamic target picking is to set the pickup position based on the conveyor velocity. This is to ensure that the robot sorts the workpieces on the moving conveyor belt.

As shown in Fig. 2, assume that the belt moves at a uniform velocity. The delta robot completes the sorting of the last workpiece, and the current position is at point A. The robot will pick up the workpiece at point B. Since the moving of the conveyor belt, the robot needs to pick up the workpiece at point C. Δt represents the time that the delta robot goes from point A to the point C by Adept motion curve. The coordinates of B (x_b, y_b, z_b) can be obtained by image recognition and perspective projection transformation. The coordinates of the placement point A are known as (x_a, y_a, z_a). The conveyor velocity is v_c, so:

$$l_{BC} = \Delta t * v_c \tag{8}$$

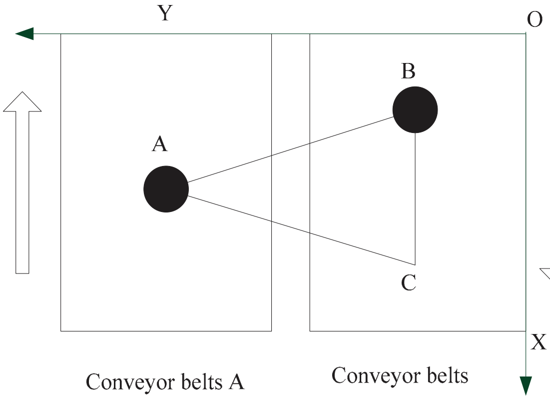

Fig. 2. Sketch of dynamic picking

Then the coordinates of C are $(x_b + \Delta t * v_c, y_b, z_b)$, and:

$$l_{AC} = \sqrt{(x_a - x_b - \Delta t * v_c)^2 + (y_a - y_b)^2} \tag{9}$$

l_{BC} represents the distance between point B and point C. l_{AC} represents the distance between point A and point C on the same horizontal plane. When S_1 and S_3 are not equal, it means that the two conveyors are not in the same horizontal plane. At this time, we can project them to a horizontal plane.

1) When $S_1 \le S_2$ and $S_3 < S_2$:

$$\Delta t = \sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 + \sqrt{\frac{S_2}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} + \sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 \tag{10}$$

and $S_2 = l_{AC}$. Substituting (10) into (9) yields the equation :

$$(\Delta t - t_1 - t_3)^4 * (\frac{1}{4\pi}+\frac{1}{8})^2 * a_{\max}^2 - \Delta t^2 * v_c^{\ 2} + 2\Delta t * v_c * (x_a - x_b) - l_{AB}^2 = 0 \tag{11}$$

2) When $S_1 > S_2$ and $S_3 \ge S_2$:

$$\Delta t=\sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}+\sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} \tag{12}$$

In this case, Δt is calculated directly.

3) When $S_1 \le S_2$ and $S_3 \ge S_2$:

$$\begin{aligned}\Delta t=&\sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2+\\&\sqrt{\frac{S_2}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2+\sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\end{aligned} \tag{13}$$

The following equation is obtained:

$$\begin{aligned}&16(\Delta t-t_1-2t_3)^4*(\frac{1}{4\pi}+\frac{1}{8})^2*a_{\max}^2-\Delta t^2*v_c^{\ 2}+\\&2\Delta t*v_c*(x_a-x_b)-l_{AB}^2=0\end{aligned} \tag{14}$$

4) When $S_1 > S_2$ and $S_3 < S_2$:

$$\begin{aligned}\Delta t=&\sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}+\sqrt{\frac{S_2}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2\\&+\sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2\end{aligned} \tag{15}$$

The following equation is obtained:

$$\begin{aligned}&16(\Delta t-2t_1-t_3)^4*(\frac{1}{4\pi}+\frac{1}{8})^2*a_{\max}^2-\Delta t^2*v_c^{\ 2}+\\&2\Delta t*v_c*(x_a-x_b)-l_{AB}^2=0\end{aligned} \tag{16}$$

among them:

$$t_1=\sqrt{\frac{S_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 \tag{17}$$

$$t_3=\sqrt{\frac{S_3}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}\Big/2 \tag{18}$$

$$l_{AB}=\sqrt{(x_a-x_b)^2+(y_a-y_b)^2} \tag{19}$$

For the above 1), 3) and 4), they are all quartic equations. A quadratic equation can be solved using Ferrari's method. The above (11), (14) and (16) are respectively transformed into the following forms:

$$x^4+bx^3+cx^2+dx+e=0 \tag{20}$$

It can be converted into the following formula:

$$x^4+bx^3=-cx^2-dx-e \tag{21}$$

On both sides of the above formula plus $\left(\frac{1}{2}bx\right)^2$, get the following equation:

$$\left(x^2+\frac{1}{2}bx\right)^2=\left(\frac{1}{4}b^2-c\right)x^2-dx-e \tag{22}$$

Then on both sides plus $\left(x^2+\frac{1}{2}bx\right)y+\frac{1}{4}y^2$, get the following equation:

$$\begin{aligned}&\left(x^2+\frac{1}{2}bx+\frac{1}{2}y\right)^2=\left(\frac{1}{4}b^2-c+y\right)x^2+\\&\left(\frac{1}{2}by-d\right)x+\left(\frac{1}{4}y^2-e\right)\end{aligned} \tag{23}$$

On the right side of the above equation is a quadratic equation about x ,when:

$$\begin{aligned}\Delta&=\left(\frac{1}{2}by-d\right)^2-4\left(\frac{1}{4}b^2-c+y\right)\left(\frac{1}{4}y^2-e\right)\\&=-y^3+cy^2-(bd-4e)y-\left(4c-b^2\right)e+d^2=0\end{aligned} \tag{24}$$

the following equation is obtained:

$$\left(x^2+\frac{1}{2}bx+\frac{1}{2}y\right)^2=\left(\frac{1}{4}b^2-c+y\right)\left[x+\frac{\frac{1}{2}by-d}{2\left(\frac{1}{4}b^2-c+y\right)}\right]^2 \tag{25}$$

So:

$$2x^2+bx+y=\pm\left(x\sqrt{b^2+4y-4c}+\frac{by-2d}{\sqrt{b^2+4y-4c}}\right) \tag{26}$$

Ferrari's method is to solve (24) first (a cubic equation, can be solved by Cardin formula) and get the y , and then solve the above equation (Two quadratic equations). Equation (24) generally has three roots. In general, the y which maximizes $\left|b^2+4y-4c\right|$ is selected.

When these three roots all make $\left|b^2+4y-4c\right|$ equal to zero, this means (24) has a triple root. Assume the root is y_0, (24) should be:

$$\left(y-y_0\right)^3=0 \Rightarrow y^3-3y_0y^2+3y_0^2y-y_0^3=0 \tag{27}$$

Comparing the above formula with (24), we can get:

$$\begin{cases} c=3y_0 \\ bd-4e=3y_0^2 \\ \left(4c-b^2\right)e-d^2=-y_0^3 \end{cases} \tag{28}$$

Combining (28) with $b^2+4y_0-4c=0$, we can obtain:

$$\begin{cases} b=2\sqrt{2y_0} \\ c=3y_0 \\ d=\sqrt{2y_0}\,y_0 \\ e=y_0^2/4 \end{cases} or \begin{cases} b=-2\sqrt{2y_0} \\ c=3y_0 \\ d=-\sqrt{2y_0}\,y_0 \\ e=y_0^2/4 \end{cases} \tag{29}$$

Substituting the above formula and $y=y_0$ into (23), we can obtain:

$$\frac{1}{4}\left(2x^2+bx+y_0\right)^2=0 \tag{30}$$

This is to say, when (24) has three roots, (20) can be transformed into the above equation. To solve the above equation, we can get the four roots of a quadratic equation (two pairs of double roots).

Combined with the practical situation, we should select the positive real solution. If there is more than one positive real number, choose the smaller one. And we should verify that the solution is consistent with the actual situation. If there is no solution to meet the requirements, then give up the sorting of this workpiece and do the missing record. Once Δt is obtained, the position of the picking point C can be determined and then we can pick up the workpiece.

III. TRAJECTORY PLANNING FOR MULTIPLE TYPES OF WORKPIECES

Aiming at the demand for sorting multiple types of workpieces for delta robot，we first analyze the situation of the two types of workpiece. In this paper, we first model the trajectory planning for two types of workpieces using dynamic picking algorithm, analyze the model and then get the trajectory planning algorithm for multiple types of workpieces.

A. the Model of the Trajectory Planning for Two Types of Workpieces

As shown in Fig. 3, assume there are i workpieces in the image, m workpieces of type A and $i-m$ workpieces of type B. The coordinates of placing the workpiece of type A and type B are known, which are point a $\left(x_a,y_a,z_a\right)$ and point b $\left(x_b,y_b,z_b\right)$. Assume that every time Pick-and-Place Operation uses the Adept motion curve as described above. And assume that the height of the Adept motion curve at each operation is the same, and $s_1=s_3, s_2>s_3$.

When the n th workpiece is going to be picked, the coordinates of the workpiece are $\left(x_n+v_c*(t_{1f}+t_{1b}+t_{2f}+t_{2b}...+t_{nf}),y_n,z_n\right)$, t_{nf} represents the time the robot takes from the placement point a or b of the $(n-1)$ th workpiece to the n th workpiece's picking position(t_{1f} represents the time the robot takes from the origin to the first workpiece's picking position), t_{nb} represents the time the robot takes from the n th workpiece's picking position to the placement point a or b of the n th workpiece, $\sqrt{(x_{p(n-1)}-x_n-v_c*(t_{1f}+t_{1b}+t_{2f}+t_{2b}...+t_{nf}))^2+(y_{p(n-1)}-y_n)^2}$ represents the horizontal distance from the placement point a or b of the $(n-1)$ th workpiece to the n th workpiece's picking position, $\sqrt{(x_n+v_c*(t_{1f}+t_{1b}+t_{2f}+t_{2b}...+t_{nf})-x_{pn})^2+(y_n-y_{pn})^2}$ represents the horizontal distance from the n th workpiece's picking position to the placement point a or b of the n th workpiece. Here $\left(x_{pn},y_{pn},z_{pn}\right)$ represents the placement point of the n th workpiece based on the type, for example, if the n th workpiece is type A, then $\left(x_{pn},y_{pn},z_{pn}\right)$ are $\left(x_a,y_a,z_a\right)$, or if the n th workpiece is type B, then $\left(x_{pn},y_{pn},z_{pn}\right)$ are $\left(x_b,y_b,z_b\right)$. So every one-way Adept motion curve takes the time:

$$\begin{aligned} t &= \frac{t_1}{2}+t_2+\frac{t_3}{2}=t_1+t_2 \\ &= \sqrt{\frac{s_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}+\sqrt{\frac{s_2}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} \end{aligned} \tag{31}$$

In the above formula, s_1 is fixed. In order to find the minimum time, we should minimize the following time:

$$t_{all}=2\sum_{1}^{i}\sqrt{\frac{s_1}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}+$$
$$\sum_{1}^{i}\sqrt{\frac{\sqrt{(x_{p(n-1)}-x_n-v_c*(t_{1f}+t_{1b}+t_{2f}+t_{2b}...+t_{nf}))^2+(y_{p(n-1)}-y_n)^2}}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}}$$
$$+\sum_{1}^{i}\sqrt{\frac{\sqrt{(x_n+v_c*(t_{1f}+t_{1b}+t_{2f}+t_{2b}...+t_{nf})-x_{pn})^2+(y_n-y_{pn})^2}}{(\frac{1}{4\pi}+\frac{1}{8})a_{\max}}} \quad (32)$$

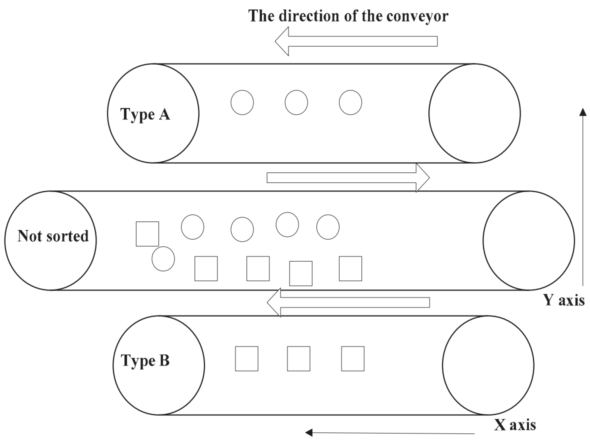

Fig. 3. Sketch of sorting two types of workpieces

B. Trajectory Planning for Multiple Types of Workpieces

The process of sorting multiple types of workpieces by delta robots is as follows:

1) Create a queue to be picked: Category information, physical coordinates, which obtained from the image coordinates through the perspective projection transformation, and image acquisition time are added to the queue. So each object in the queue has time property, position property and category property. Then update the position property according to the current time, for example, for the object (t_1, x, y, z, n), when the time is t_2, the object is updated to $(t_2, x+v_c*(t_2-t_1), y, z, n)$. Where v_c is the velocity of the belt, n is the category, (x, y, z) are the coordinates at the time t_1.

2) Rearrange the queue in the following way:

a) The workpiece outside the optimum pick-up area have the highest priority: If there is an object in the current queue and its position property is beyond the optimum pick-up area, that is, in front of the optimum pick-up area, it should be preferentially picked up and be ahead of the queue. If there are more than one object in front of the optimum pick-up area, the more the object's position is at the front of the conveyor, the more the object is in the front of the queue. The solution of t_{nf} is obtained from (11) above, t_{nb} is calculated from the picking position according to t_{nf} and the placement point of the nth workpiece.

This is a constrained optimization problem. The computational cost of this optimization problem is great. If the intelligent optimization algorithm is used, the calculation is so complicated that it may not meet the requirements of the real-time control system. The optimal solution may change with time, and the complexity will increase when the category of sorting increases. Thus, the solution should be considered, which determines the sorting order of the workpiece firstly and then calculates the dynamic picking position.

b) The priority of the workpiece in the optimum pick-up area is second: In the optimum pick-up area, the workpiece that is closest to the current placement point is preferentially picked up. That is, the closer to the current placement point, the more in the front of the queue.

3) Set the head of the queue be the workpiece to be picked up, let the head out of the queue and add a new object at the end of the queue: update the head of the queue according to the time described as 1), then use the dynamic picking algorithm combined with the position property of the object to calculate the coordinates of the picking point. The robot needs to move to the coordinates to pick up this workpiece, and place this workpiece in the placement point according to the category.

4) Update the queue at current time according to the method described as 1) and rearrange the queue according to 2): repeat 3) and 4) until all the workpieces are sorted. If there is a workpiece that is about to leave the working space of the delta robot and the delta robot has not picked it up yet, we should stop the conveyor belt and to pick up the workpiece, or give up the sorting of this workpiece and do the missing record.

In order to meet the requirements of real-time system, the time to maintain the queue should be as short as possible. So the length of the queue should be limited, that is, the queue only contains a fixed number of workpieces in the images. The workpiece at the front of the conveyor belt in the image are added to the queue first.

The optimum pick-up area described above should be in the working space of the delta robot. It is pointed out that [15] the robot pose error is distributed symmetrically in the xy direction along the coordinate axis and symmetrically in the z direction along the origin, and the closer the robot is to the edge region, the larger the pose error is [15]. Therefore, the optimal pick-up area can adaptively remove the boundary area of the corresponding working space according to the requirements of the accuracy of delta robot.

IV. EXPERIMENTAL VERIFICATION

A. Dynamic Picking Algorithm Experiment

The dynamic picking algorithm is tested in C++ language based on windows operating system. The results are shown in

Table I. Among them, the maximum acceleration in the Adept motion curve $a_{max} = 30\ m/s^2$, the upper limit of the velocity of the end-effector $v_{lim} = 6\ m/s$, $\bar{t}$ represents the average time for calculating 10,000 times using dynamic picking algorithm.

The experimental results show that the algorithm is accurate and fast, which can meet the real-time requirements. And the algorithm does not need to set the initial value, less restrictive conditions make it more convenient.

TABLE I. EXPERIMENTAL RESULTS OF DYNAMIC PICKING ALGORITHM

V_c (mm/s)	S1 (mm)	S3 (mm)	Xa –Xb (mm)	Ya –Yb (mm)	equation number	Δt (ms)	$\bar{t}$ (μs)
150	100	100	70	300	(11)	348.929	10.9268
150	100	100	-50	350	(11)	371.726	9.5103
180	100	100	80	320	(11)	356.126	9.6398
150	50	150	50	50	(14)	153.351	10.3476
150	150	50	60	70	(16)	144.448	10.6048

B. Robot experiment

The experimental system includes industrial camera, robot controller, host computer, delta robot and corresponding servo, etc. Fig. 4 is experimental prototype. The picking experiment is carried out to test the accuracy and stability of the algorithm under different conveyor speed. Considering coordinate error and calibration error, we allow certain time tolerances and position tolerances. The experimental results are shown in TABLE II.

In the experiment, the maximum pickup speed is 90 times per minute. Optimization of the delta mechanism used in the experiment can lead a faster pick-up speed. The missing rate is less than 2 per thousand, and the misclassification rate is 0, which proves the accuracy and stability of the algorithm.

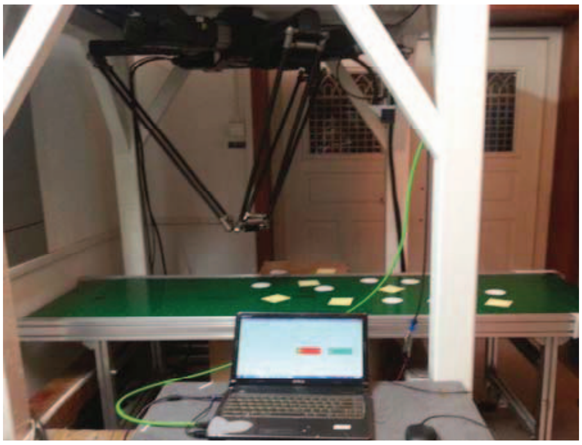

Fig. 4. Experimental robot

TABLE II. ROBOT EXPERIMENTAL RESULTS

V_c (mm/s)	total number of picking	missing number	misclassification number
100	557	0	0
150	590	1	0
160	550	1	0

V. CONCLUSION

In this paper, we analyzed different conditions of the Adept motion curve, and proposed a dynamic picking algorithm based on Ferrari's method. The algorithm is accurate and fast, does not need to set the initial value and is suitable for practical engineering applications. And trajectory planning for multiple types of workpieces is carried out. The workpiece to be sorted is determined first, and then will be picked up at the picking point calculated by dynamic picking algorithm. The algorithm has high accuracy and stability.

REFERENCES

[1] P. Vischer, R. Clavel, "Kinematic calibration of the parallel Delta robot", *Robotica*, vol.16.2, pp. 207-218, 1998.

[2] L. Rey, R. Clavel, "The Delta Parallel Robot", *Parallel Kinematic Machines*, pp. 401-417, 1999.

[3] F. Pierrot, C. R. A. Fournier, "DELTA: a simple and efficient parallel robot", *Robotica*, vol 8.2, pp. 105-109, 1990.

[4] J. P. Merlet, "Parallel Robots", *Solid Mechanics & Its Applications*, vol. 128, pp. 2091-2127, 2006.

[5] L. H. FENG, W. G. ZHANG, Z. Y. GONG, G. Y. LIN, D. K. LIANG, "Developments of Delta-Like Parallel Manipulators", *Robot*, vol.36, 375-384, 2014.

[6] L. M. ZHANG, "Integrated optimal design of delta robot using dynamic performance indices", Tianjin, Tianjin University, 2011.

[7] H. X. LI, "Research on key technologies of automatic teaching of welding robot based on vision feedback", Guangzhou, South China University of Technology, 2010.

[8] H. P. NI, Y. N. LIU, C. R. ZHANG, Y. F. WANG, F. H. XIA, Z. S. QIU, "Sorting System Algorithms Based on Machine Vision for Delta Robot", *Robot*,vol. 38, 49-55, 2016.

[9] W. C. Zhang, "Control technique and kinematic calibration of Delta robot based on computer vision", Tianjin, Tianjin University, 2012.

[10] Z. X. XIE, D. W. SHANG, P. REN, "Optimization and Experimental Verification of Pick-and-place Trajectory for a Delta Parallel Robot Based on Lamé Curves", *Journal of Mechanical Engineering*, vol.51, 52-59, 2015.

[11] X. ZHANG, "Research on Control Method of High Speed Lightweight Parallel Manipulator", Tianjin, Tianjin University, 2005.

[12] T. Huang, Z. Li, M. Li, D. G. Chetwynd, C. M. Gosselin, "Conceptual design and dimensional synthesis of a novel 2dof translational parallel

robot for pick-and-place operations", *Journal of Mechanical Design*, vol. 126.3, pp. 449-455, 2004.

[13] V. Nabat, l. O. R. M. De, O. Company, S. Krut, F. Pierrot, "Par4: very high speed parallel robot for pick-and-place", *Ieee/rsj International Conference on Intelligent Robots and Systems*, pp. 553-558, 2005.

[14] W. C. Zhang, "R&D of Vision Control System for Delta Robot", Tianjin, Tianjin University, 2009.

[15] M. HUANG, "Research on the Working Performance and Pose Error with Sphere Joint Clearance of the Delta Robot for Picking Operation", Guangzhou, South China University of Technology, 2015.