

# Motion Control Functions for Industrial Robots

Author: Kiyoshi Maekawa\*

## 1. Introduction

In the industrial robot business, Mitsubishi Electric focuses on compact vertical and horizontal multi-joint robots. The company is working on enhancing their rigidity, improving the drive mechanisms such as the motors, and to develop robotic motion control functions. In the compact robot market, growing requests for the motion control functions include: faster cycle time and higher accuracy, better usability, and application specific functions.

To increase the speed of the cycle time, in addition to the improvement of drive mechanisms for higher velocity and higher acceleration/deceleration, we have introduced an optimal acceleration control, whereby the optimal acceleration/deceleration time is automatically calculated to minimize the traveling time according to the robot's position and posture and the load attached to the tip of the arm. We have also introduced an optimal path connection function that provides a shortest path connection under given constrained conditions and thus a shorter operating time, whereby the next move command is initiated without stopping at a route point that requires no positioning. In addition, to achieve higher accuracy, we have introduced a gravity compensation function, which corrects the deflection of each axis due to gravity.

Usability improvement functions include: impact detection function, which detects without any additional sensor when the robot's hand or body has hit an object and makes an emergency stop to reduce the damage to the robot and object; a position restoration support function, which reduces the number of teaching points

after replacing drive elements such as the motor and belt or after the robot has been reinstalled; and a maintenance forecast function, which analyzes the operating status of the robot to estimate the maintenance timing for lubrication, belt replacement, etc.

Application-specific functions include: an orthogonal compliance control function, which allows the robot to flexibly operate only in the direction specified by an orthogonal coordinate system; and a conveyer tracking function, which makes the robot operate following the conveyer movement.

This article introduces typical examples from these motion control functions.

## 2. Functions for Faster Cycle Time and High Accuracy

Even if a robot is operated at the same acceleration/deceleration rate and at the same velocity, the required driving torque varies depending on its position and posture at the start and end points of motion. Similarly, the required driving torque also varies when the mass of the load attached to the tip of the robot arm is changed. Meanwhile, the motor and reducer used on each robot axis have a maximum allowable torque. Consequently, if the robot is operated at the maximum velocity within the motor and reducer's constraint, the acceleration rate needs to be reduced when a heavy load is attached to the hand or the robot is operated in a stretched posture, because a greater driving torque would be required to produce the same rate of acceleration. On the contrary, in the case of a light load at the tip of the arm or the robot operation in a retracted



Fig. 1 Industrial robots

posture, operation at a high acceleration rate is possible (see Fig. 2). As such, the optimal acceleration function determines the acceleration rate optimized for each robot movement according to its position/posture and the load attached to the tip of its hand.

The equation of motion of the robot is expressed as:

$$\tau = M(q)a + h(q, v) + g(q) + f(v) \quad (1)$$

where  $\tau$  is the vector consisting of the driving torque of each axis,  $M(q)$  is the inertia matrix, and  $q$ ,  $v$ ,  $a$ ,  $h(q, v)$ ,  $g(q)$  and  $f(v)$  are the vectors consisting of the position, velocity, acceleration rate, centrifugal and Coriolis forces, gravity force, and friction force of each axis, respectively. The driving torque of the robot varies according to its position/posture and the load at the tip of its hand because the inertia matrix  $M(q)$ , centrifugal and Coriolis forces  $h(q, v)$ , and gravity force  $g(q)$  are functions of the position and posture of each axis.

Consequently, the optimal acceleration function uses the robot's equation of motion (1) to calculate the acceleration time and deceleration time that give the shortest motion time for each robotic movement within the constraint of motors and reducers. Since the inertia matrix is a function of the position of each axis, and hence varies during the operation, it is impractical to calculate the equation of motion (1) at every robot position during its operation due to computational complexity. Therefore, representative points are determined within the acceleration period and deceleration period, and the calculation of the equation of motion (1) is repeated at these representative points to calculate the acceleration time and deceleration time that give the shortest motion time within the constraint of motors and reducers, without increasing the calculation time.

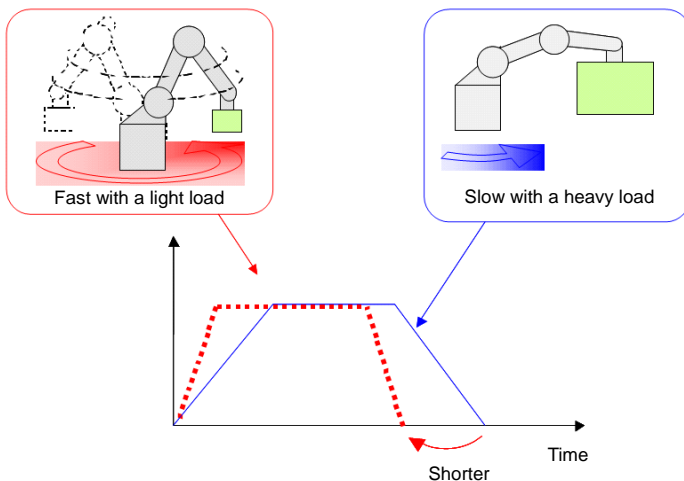


Fig. 2 Optimal acceleration

In addition, the constraint of the motor may change according to its speed of rotation. Specifically, the maximum allowable torque is maintained constant up to a certain speed, but the allowable torque drops above a certain speed. When the robot is operated in a velocity range including a low-torque zone as above with a constant acceleration/deceleration rate throughout the acceleration/deceleration period, such torque-lowering characteristic in the high-velocity range would greatly affect the robot operation, causing the acceleration/deceleration rate to be set unnecessarily low even in the low-velocity range. Therefore, our design allows the acceleration/deceleration rate setting to be independently adjustable for low- and high-velocity zones within the same acceleration/deceleration period.

### 3. Functions for Better Usability

During an operation for robot teaching or program checking, the robot is often made to touch or hit an object. We have therefore introduced an impact detection function, which detects, without any additional sensor, that the robot has hit an object. This impact detection function continually calculates an estimated torque value based on the robot's equation of motion (1), and if the difference between the estimated torque and the actual torque exceeds a threshold value, a collision is judged to have occurred (Fig. 3). However, the friction force included in the equation of motion (1) varies according to the temperature. Consequently, to improve the estimation accuracy, the friction coefficient is identified online and used for calculating the equation of motion (1). Figure 4 shows an example of the improvement in accuracy in estimating the torque thanks to this friction estimation. The optimal threshold value also varies according to the intensity of the robot's action. The threshold level is thus changed in real time according to the operating condition of the robot.

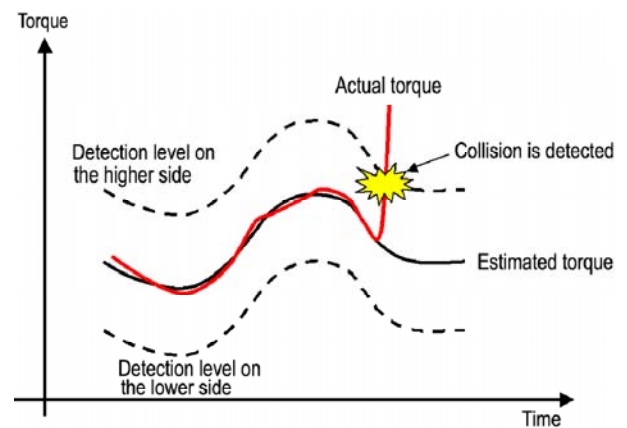


Fig. 3 Impact detection

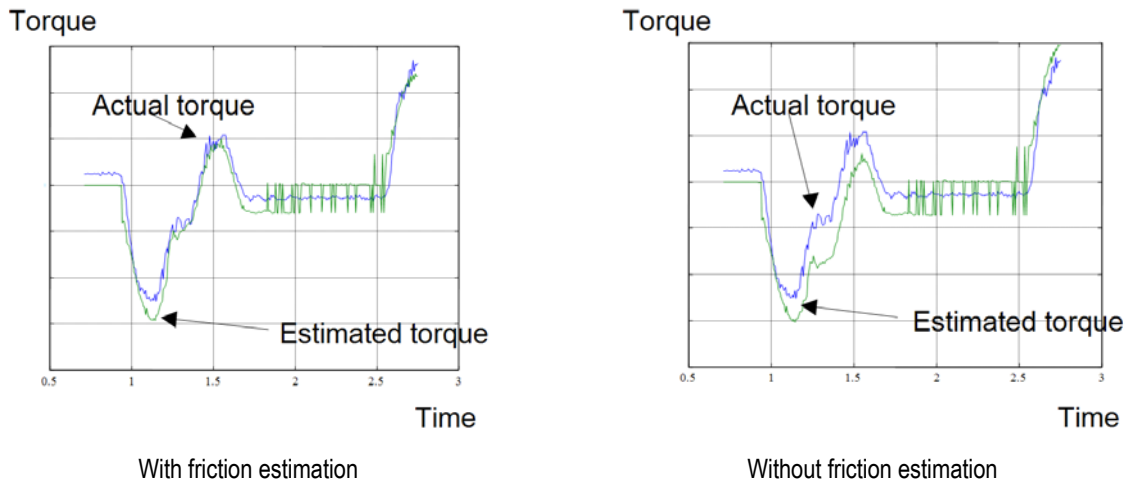


Fig. 4 Accuracy improvement of estimated torque by friction estimation

#### 4. Conclusion

We will continue research and development on robot functions for improving the velocity and accuracy, which constitute the basic performance of robots. We will also improve the usability functions to make robots more user friendly. In addition to these general-purpose functions, requests for application-specific functions are growing. In particular, applied control of sensors, such as three-dimensional sensors, force sensors and vision sensors, will increase in importance.