TRAJECTORY PLANNING FOR ROBOT MANIPULATORS

Prof. Rohan Munasinghe
Based on MSc Research by Chinthaka Porawagama

Industrial Robotics Involves in
- Pick-and-place operations
- Assembling operations
- Loading and stacking
- Automated welding, etc.

Proper motion planning is needed in these applications
Manipulators with multi degree of freedom for accomplishing various complex manipulation in the work space

Path: only geometric description  
Trajectory: timing included

Václav Hlaváč  
Czech Technical University in Prague
Joint Space Vs Operational Space

- **Joint-space description:**
  - The description of the motion to be made by the robot by its joint values.
  - The motion between the two points is unpredictable.

- **Operational space description:**
  - In many cases operational space = Cartesian space.
  - The motion between the two points is known at all times and controllable.
  - It is easy to visualize the trajectory, but it is difficult to ensure that singularity does not occur.

Planning in Operational Space

Sequential motions of a robot to follow a straight line.

Cartesian-space trajectory
(a) The trajectory specified in Cartesian coordinates may force the robot to run into itself, and
(b) the trajectory may require a sudden change in the joint angles.
Planning in Operational Space

- Calculate path from the initial point to the final point.
- Assign a total time $T_{path}$ to traverse the path.
- Discretize the points in time and space.
- Blend a continuous time function between these points.
- Solve inverse kinematics at each step.

Advantages
- Collision free path can be obtained.

Disadvantages
- Computationally expensive due to inverse kinematics.
- It is unknown how to set the total time $T_{path}$.

Planning in Joint Space

- Calculate inverse kinematics solution from initial point to the final point.
- Assign total time $T_{path}$ using maximal velocities in joints.
- Discretize the individual joint trajectories in time.
- Blend a continuous function between these point.

Advantages
- Inverse kinematics is computed only once.
- Can easily take into account joint angle, velocity constraints.

Disadvantages
- Cannot deal with operational space obstacles.
Path Definition

“Expressing the desired positions of a manipulator in the space, as a parametric function of time”


Task to Trajectory

1. Task planning
2. Sequence of pose points (“knots”) in Cartesian space
3. Interpolation in Cartesian space
4. Cartesian geometric path (position+orientation) $p = p(u)$
5. Path sampling and inverse kinematics
6. Sequence of pose points (“knots”) in joint space
7. Interpolation in joint space
8. Geometric path in joint space $q = q(t)$

Cartesian space

Joint space

$A \rightarrow B \rightarrow C$
Types of Motion

1. Point to point motion:
   - End effector moves from a **start point** to **end point** in work space
   - All joints' movements are coordinated for the point-to-point motion
   - End effector travels in an arbitrary path

2. Motion with Via Points
   - End effector moves through an intermediate point between start and end
   - End effector moves through a via point without stopping

Joint Space Planning

**Point to point motion:**
“Describing of joints’ motions from start to end by smooth functions”

Basic stages of solving of joint space trajectory planning problem?
Point to point motion:

“Describing of joints’ motions from start to end by smooth functions”

1. Inverse kinematics of start and end points (A & B)

2. Joint angles for start and end points

\[ \theta_{1A} \rightarrow \theta_{1B} \]
\[ \theta_{2A} \rightarrow \theta_{2B} \]
\[ \theta_{3A} \rightarrow \theta_{3B} \]

3. Interpolation of start and end joint angles by smooth functions

\[ \theta_1(t): \theta_{1A} \rightarrow \theta_{1B} \]
\[ \theta_2(t): \theta_{2A} \rightarrow \theta_{2B} \]
\[ \theta_3(t): \theta_{3A} \rightarrow \theta_{3B} \]

4. Joint space trajectories for each joint
Smooth Motion → Quality of Work

- Non smooth trajectories lead to low quality in production.

Linear Trajectory

\[ q(t) = q^s + \frac{q^g - q^s}{t_g} t \]
\[ \dot{q}(t) = \frac{q^g - q^s}{t_g} \]
\[ \ddot{q}(t) = \begin{cases} \infty & t = 0, t_g \\ 0 & 0 < t < t_g \end{cases} \]

- Infinite accelerations at endpoints
- Discontinuous velocity when two trajectory segments are connected (at via points)
**Triangular Velocity Trajectory**

\[ q^g - q^s = \frac{t_g}{2} \left( \ddot{q}_{\text{min}} \frac{t_g}{2} \right) \]

\[ \ddot{q}_{\text{min}} = \frac{4(q^g - q^s)}{t_g^2} \]

\[ q(t) = \begin{cases} 
q^s + 0.5\ddot{q}_{\text{min}}t^2 & 0 \leq t < 0.5t_g \\
0.5(q^s + q^g) - 0.5\ddot{q}_{\text{min}}t^2 & 0.5t_g \leq t \leq t_g 
\end{cases} \]

- Acceleration discontinuity at endpoints and at the midpoint of the trajectory

**Linear Trajectory with Parabolic Blends**

- Zero acceleration in middle segments.
- Constant acceleration at end segments.

Acceleration discontinuity at blend points
Linear Trajectory with Parabolic Blends

- Total angular motion
  \[ S = 2(\text{parabolic}) + \text{Linear} \]
  \[ q^g - q^s = 2 \times \frac{1}{2} \ddot{q}^b t_b^2 + \dot{q}^b (t_g - t_b) \]
  \[ \ddot{q}^b t_b^2 - \dot{q}^b t_g t_b + (q^g - q^s) = 0 \]
  \[ t_b = \frac{t_g}{2} \pm \frac{1}{2} \sqrt{t_g^2 - \frac{4(q^g - q^s)}{\ddot{q}^b}} \]

- For a linear part to exist
  \[ t_g^2 - \frac{4(q^g - q^s)}{\ddot{q}^b} > 0 \quad \Rightarrow \quad \ddot{q}^b > \frac{4(q^g - q^s)}{t_g^2} \]

Minimum joint acceleration

Linear Trajectory with Parabolic Blends
Inclusion of Via Points into a Linear Trajectory with Parabolic Blends

- Via points (knot points) can be introduced between start and goal \( (q_s, q_g) \) positions with constant acceleration at via point.

Multi-stage linear parabolic blend spline

**Cubic Polynomial (Bring in Smoothness)**

Joint Position

\[
q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (1)
\]

Joint Velocity

\[
\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2 \quad (2)
\]

Joint Acceleration

\[
\ddot{q}(t) = 2a_2 + 6a_3 t \quad (3)
\]

Joint Jerk

\[
\dddot{q}(t) = 6a_3 \quad (4)
\]
Cubic Polynomial (zero speed at end-points)

- Satisfies position and velocity at end-points
  - Eg: zero speed at end-points
    \[
    \begin{align*}
    q(0) &= q^s \\
    q(t_g) &= q^g \\
    \dot{q}(0) &= 0 \\
    \dot{q}(t_g) &= 0
    \end{align*}
    \]
    (1) for \( t = 0 \), \( q^s = a_0 \)
    (1) for \( t = t_g \), \( q^g = a_0 + a_1 t_g + a_2 t_g^2 + a_3 t_g^3 \)
    (2) for \( t = 0 \), \( \dot{q}^s = a_1 \)
    (2) for \( t = t_g \), \( \dot{q}^g = a_1 + 2a_2 t_g + 3a_2 t_g^2 \)

\[
q(t) = q^s + \frac{3}{t_g^2} (q^g - q^s) t^2 - \frac{2}{t_g^3} (q^g - q^s) t^3
\]
\( 0 \leq t \leq t_g \)

- Acceleration is linear and uncontrollable
  \[
  \ddot{q}(t) = 2a_2 + 6a_2 t \quad 0 \leq t \leq t_g
  \]

Cubic Polynomial (zero speed at end-points)
Cubic Polynomial (nonzero speeds at end-points)

- Satisfies position and velocity at end-points
  - Eg: non-zero speeds at end-points

\[
\begin{align*}
q(0) &= q^s \quad (1) \text{ for } t = 0, \quad q^s = a_0 \\
q(t_g) &= q^s \quad (1) \text{ for } t = t_g, \quad q^s = a_0 + a_1 t_g + a_2 t_g^2 + a_3 t_g^3 \\
\dot{q}(0) &= \dot{q}^s \quad (2) \text{ for } t = 0, \quad \dot{q}^s = a_1 \\
\dot{q}(t_g) &= \dot{q}^s \quad (2) \text{ for } t = t_g, \quad \dot{q}^g = a_1 + 2a_2 t_g + 3a_2 t_g^2
\end{align*}
\]

\[
q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad \forall 0 \leq t \leq t_g
\]

- Acceleration is linear and uncontrollable
  \[
  \ddot{q}(t) = 2a_2 + 6a_2 t \quad \forall 0 \leq t \leq t_g
  \]

- Acceleration is linear and uncontrollable

Cubic Spline Trajectory

- Stitching cubic polynomials together

- Acceleration is not continuous at via (stitching points)
5th Order Polynomial (more oscillatory)

\[ q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \]

\[ \dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4 \]

\[ \ddot{q}(t) = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3 \]

\[ \dddot{q}(t) = 6a_3 + 24a_4 t + 60a_5 t^2 \]