INDUSTRY 4.0 CELL

INSTITUTE OF AUTOMATION AND
COMPUTER SCIENCE

# Programming for robots and manipulators
## Lecture 6

Roman Parak
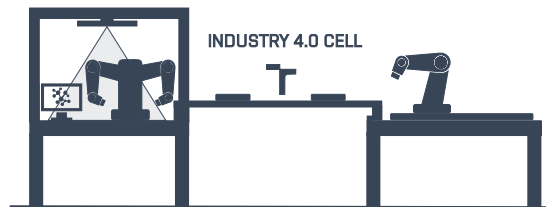
INDUSTRY 4.0 CELL

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

Brno, 2023

# Content

# Introduction

Motion planning is the problem of finding a robot motion from a start state to a goal state that avoids obstacles in the environment and satisfies other constraints, such as joint limits or torque limits. Motion planning is one of the most active subfields of robotics.

## Motion planning in industrial and mobile robotics
Determining where to move without hitting obstacles!

## Key issues in robotic planning:
1. Where am I?            Localization
2. Where have I been?   Mapping
3. Where am I going?    Planning
4. How do I get there?   Navigation
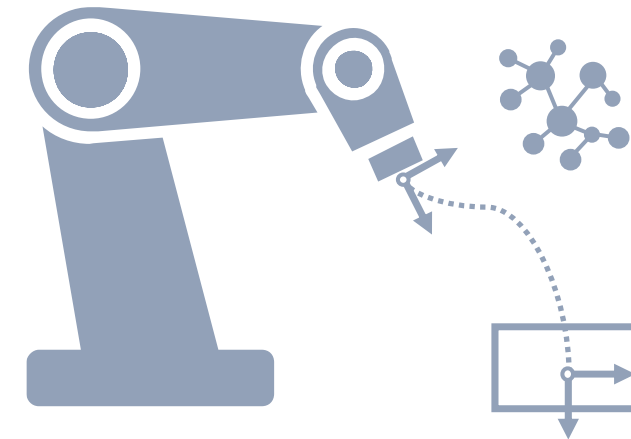
# Motion Planning

## 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.

## Trajectory in the Joint-Space:

○ Calculate the inverse kinematics solution from the initial point to the final point.

○ Discretize the individual joint trajectories in time.

○ Blend a continuous function between these point.

Motion Planning

FACULTY institute
OF MECHANICAL of automation
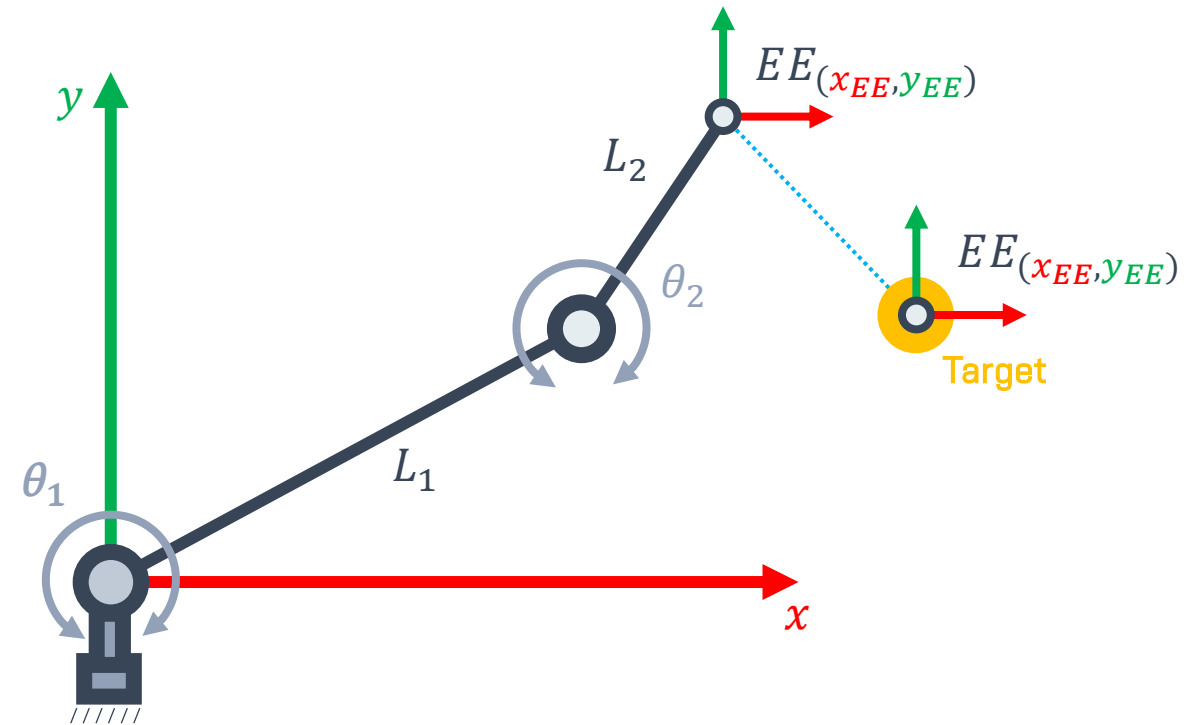ENGINEERING and computer science
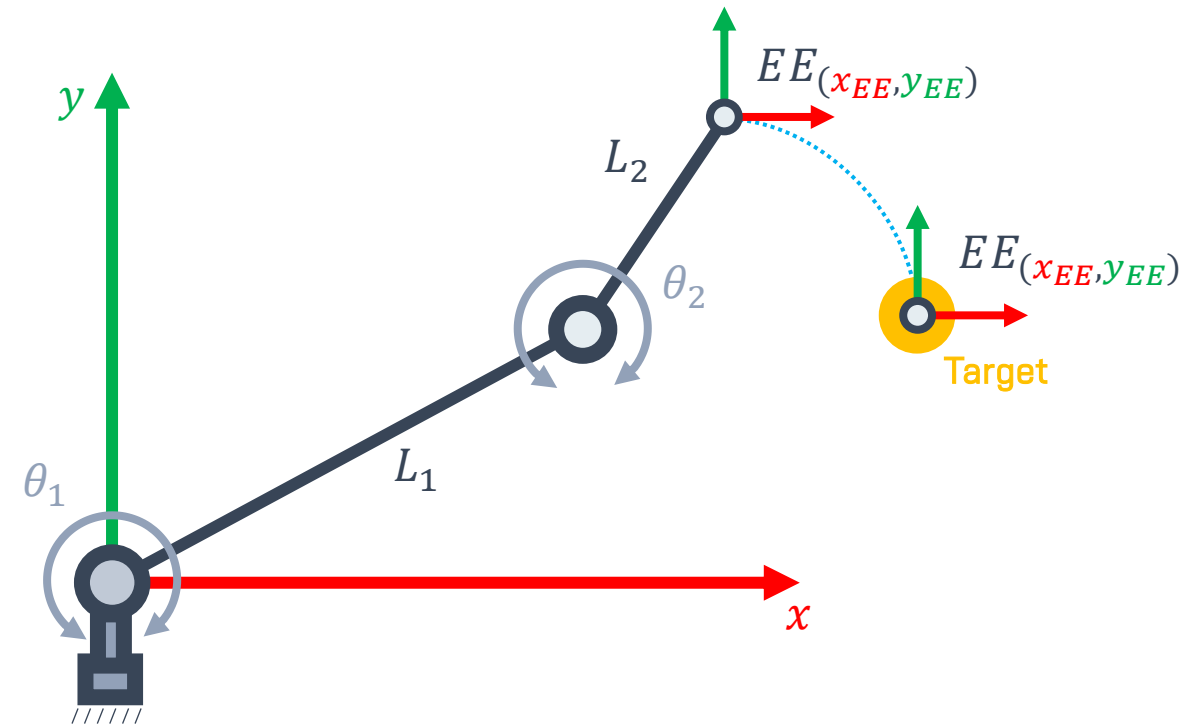
**Operational-Space (Cartesian-Space) Description:**

- 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.

**Trajectory in the Operational-Space (Cartesian-Space) :**

- Calculate the path from the initial point to the final point.

- Discretize points in time and space.

- Blend a continuous time function between these points.

- Solve inverse kinematics at each step.

$EE_{(x_{EE}, y_{EE})}$

$L_2$

$\theta_2$

$EE_{(x_{EE}, y_{EE})}$

Target

$y$

$\theta_1$

$L_1$

$x$

© RomanParak
2023  | Page 8/31

Motion Planning

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

The Path consists of an ordered number of points in space (either in the joint space or in the operating space) that the robot should follow.
The path provides a pure geometric description of motion.
The path is usually planned globally taking into account obstacle avoidance, traversing a complicated maze, etc.

A Trajectory is a path with speeds and accelerations at each of its points.
The design of a trajectory does not need global information, which simplifies the task significantly.
The trajectory is specified and designed locally. Parts of the path are covered by individual trajectories

**Note:**
Terms path, trajectory are often exchanged because they are perceived as synonyms informally.
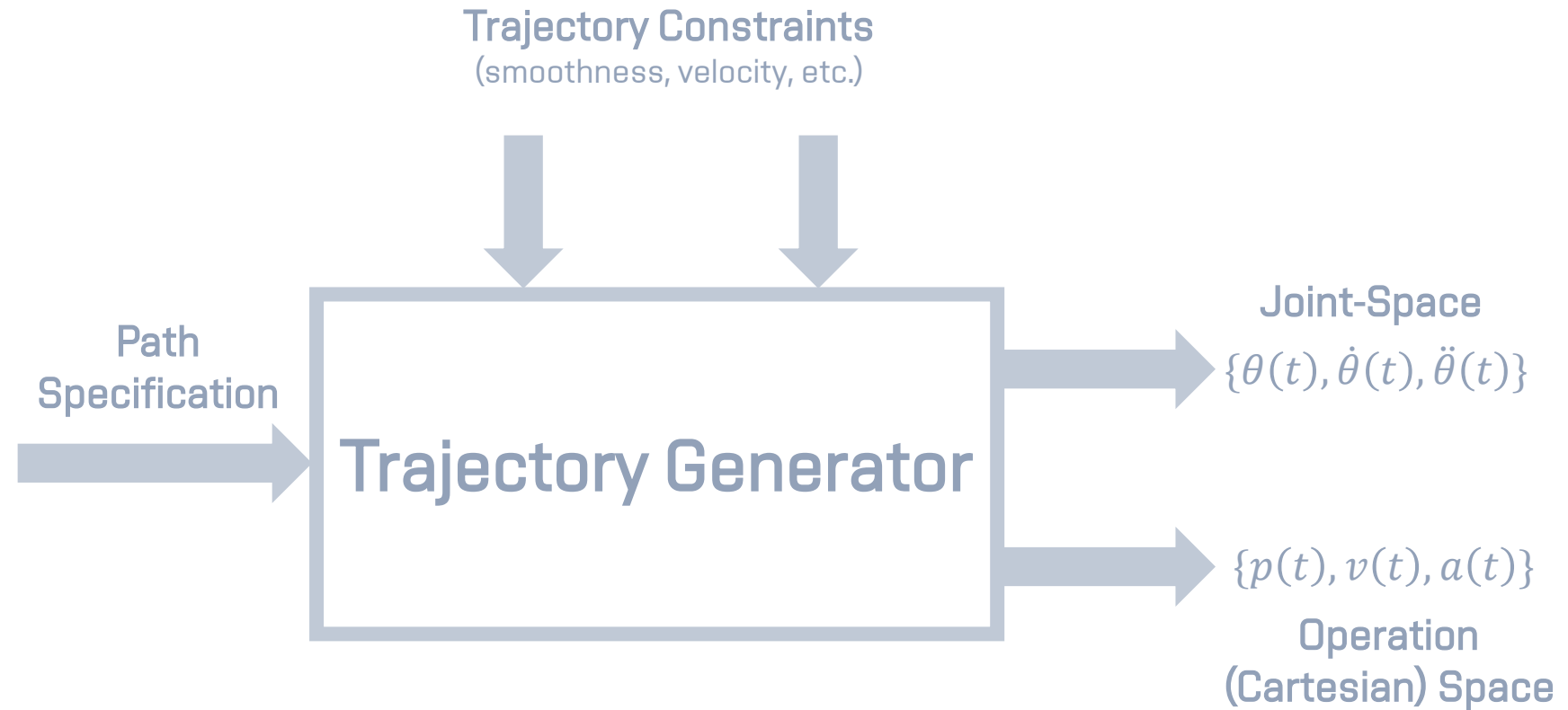
## Path Planning

- ◉ The (geometric) path is a sequence of waypoints defining the trajectory coarsely.
- ◉ Issues solved at this level: obstacle avoidance, shortest path.

The path planning task is used to find a collision-free path for the robot from one configuration to another.

## Trajectory Generation

- ◉ The path provided by path planning constitutes the input to the trajectory generator.
- ◉ Trajectory generator "approximates" the desired path waypoints by a class of polynomial functions and generates a time-based control sequence moving the manipulator/mobile platform from its initial configuration to the destination.

Trajectory generation aims at creating inputs to the motion control system, which ensures that the planned trajectory is smoothly executed.

**Trajectory Constraints**
(smoothness, velocity, etc.)

Path
Specification

## Trajectory Generator

Joint-Space
$$\{\theta(t), \dot{\theta}(t), \ddot{\theta}(t)\}$$

$$\{p(t), v(t), a(t)\}$$
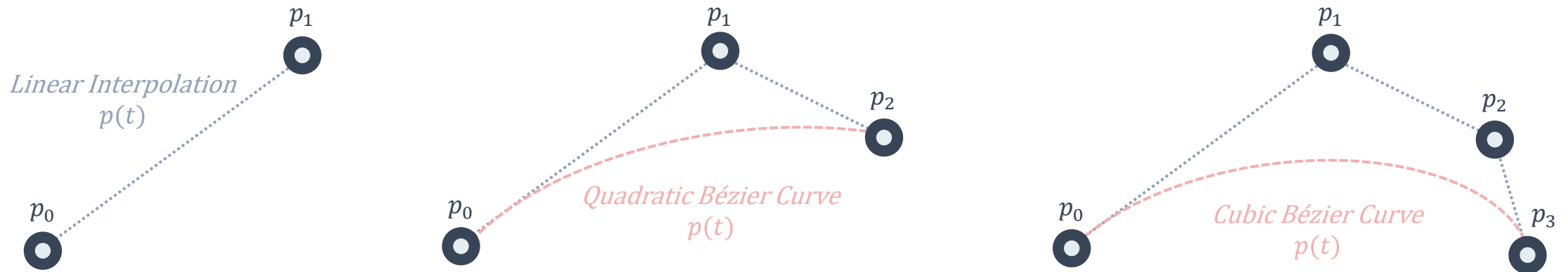Operation
(Cartesian) Space

Motion Planning

# Bézier Curves

The Bézier curve is a parametric curve used in computer graphics and related fields. The curve associated with the Bernstein polynomial is named after Pierre Bézier, who used it in the 1960s to design curves for Renault car bodies.

Other uses include computer font design, animation, and motion planning. Bézier curves can be combined in the form of a Bézier-Spline or generalized to higher dimensions in the form of a Bézier surface.

If we want to save a curve or surface, it is impractical to save each point. The solution is to somehow represent entire curves using a small amount of data.



*Linear Interpolation* $p(t)$ — $p_0$, $p_1$

*Quadratic Bézier Curve* $p(t)$ — $p_0$, $p_1$, $p_2$

*Cubic Bézier Curve* $p(t)$ — $p_0$, $p_1$, $p_2$, $p_3$

© RomanParak
2023    | Page 13/31

Bézier Curves

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

**Definition:**

Given two control points $p_0$ and $p_1$ we define the linear Bézier curve to be the curve parametrized by:

$$p(t) = (1 - t)p_0 + tp_1, t \in [0, 1]$$

*Linear Interpolation*
*$p(t)$*

$p_1$

$p_0$

© RomanParak
2023    | Page 14/31

Bézier Curves

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science
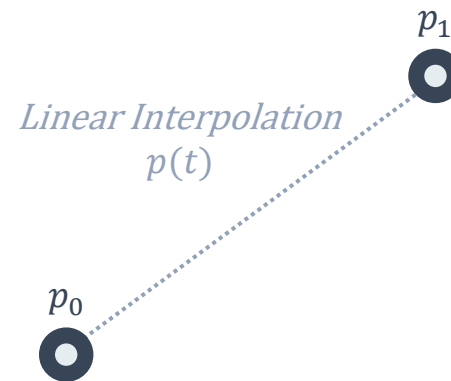
**Definition:**

Given three control points $p_0$, $p_1$ and $p_2$ we define the quadratic Bézier curve (degree 2 Bézier curve) to be the curve parametrized by:

$$p(t) = (1 - t)^2 p_0 + 2t(1 - t)p_1 + t^2 p_2, t \in [0, 1]$$

$p_1$

$p_2$

*Quadratic Bézier Curve*
*$p(t)$*

$p_0$

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

NETME Centre
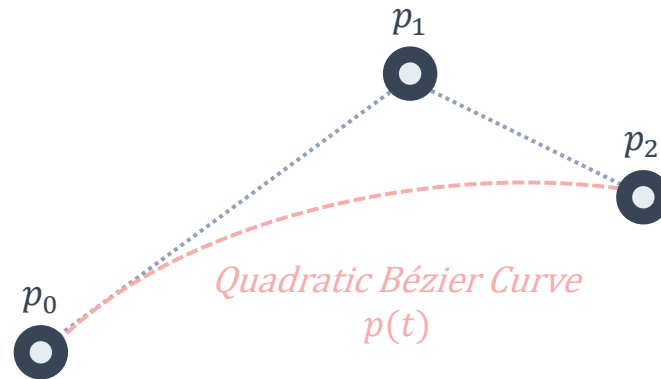
**Definition:**

Given four control points $p_0$, $p_1$, $p_2$ and $p_3$ we define the cubic Bézier curve (degree 3 Bézier curve) to be the curve parametrized by:
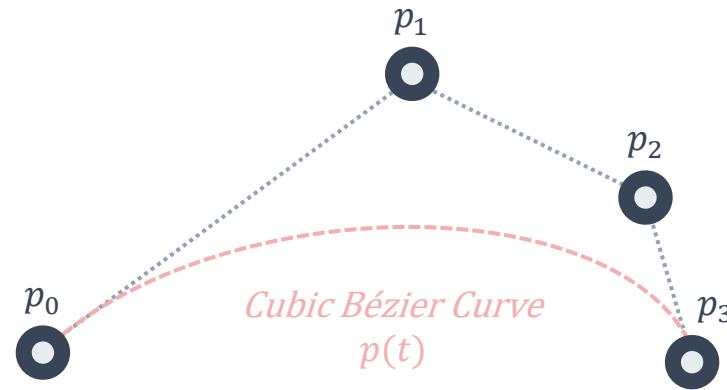
$$p(t) = (1-t)^3 p_0 + 3t(1-t)^2 p_1 + 3t^2(1-t)p_2 + t^3 p_3, t \in [0,1]$$



$p_1$

$p_2$

*Cubic Bézier Curve*
$p(t)$

$p_0$

$p_3$

**Definition:**

Given $n + 1$ control points $p_0, p_1, \ldots, p_n$ we define the degree $n$ Bézier curve to be the curve parametrized by (De Casteljau's algorithm):

$$p(t) = \sum_{i=0}^{n} \binom{n}{i} t^i (1-t)^{n-i} p_i \, , \, t \in [0, 1]$$

$p_1$

$p_2$

$p_3$

$p_{n+1}$

$p_n$

$p_0$

*N-Degree Bézier curve*
*$p(t)$*

$p_{n-2}$

$p_{n-1}$
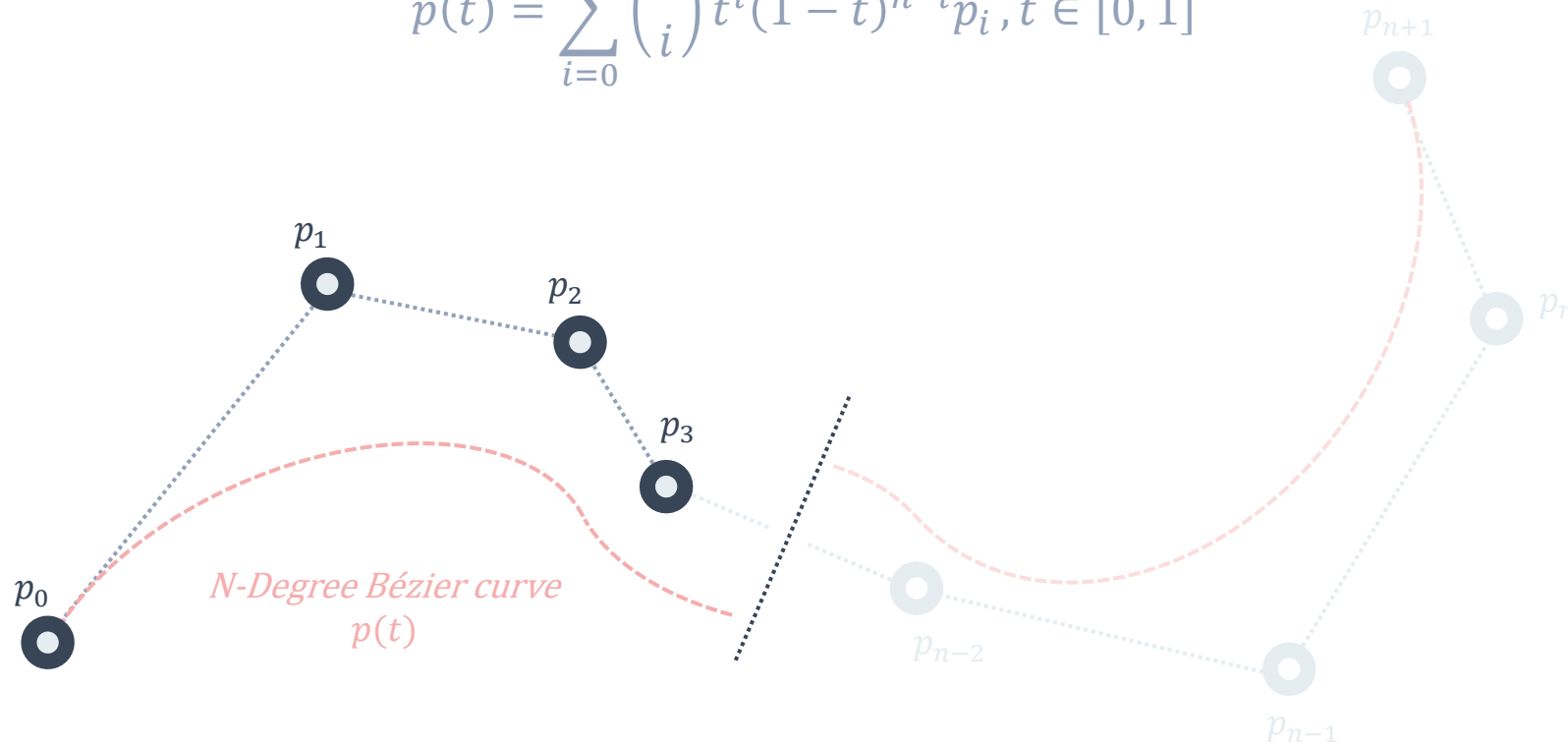
# Planning Methods

Motion planning is the problem of finding a robot motion from a start state to a goal state that avoids obstacles in the environment and satisfies other constraints, such as joint limits or torque limits. Motion planning is one of the most active subfields of robotics.

## Search Approach

Searches algorithms use some function that estimates the cost from the current state to the goal, presuming that such a function is efficient (Heuristic approach).

## Methods:

A* Search (one of the most powerful and popular graph search algorithm)
Dijkstra's algorithm, Breadth-first search, Suboptimal A* Search

The disadvantage of these approaches, however, is their high computational complexity, making them unsuitable for systems having more than a few degrees of freedom.

© RomanParak
2023    | Page 19/31

Planning Methods

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

Motion planning is the problem of finding a robot motion from a start state to a goal state that avoids obstacles in the environment and satisfies other constraints, such as joint limits or torque limits. Motion planning is one of the most active subfields of robotics.

## Sampling Approach

Sampling methods, relies on a random or deterministic function to choose a sample from the space. Sampling methods generally give up on the resolution-optimal solutions of a grid search in exchange for the ability to find satisficing solutions quickly in high-dimensional state spaces.

## Methods:

RRT (Rapidly-Exploring Random Tree), RRT-Connect, Bidirectional RRT, etc.
PRM (Probabilistic Roadmap)

© RomanParak
2023    | Page 20/31

Planning Methods

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

Planning the trajectory of the robotic arm as one of the most basic and challenging research topics in robotics has found considerable interest from research institutes in recent decades. Traditional task and motion planning methods, such as RRT, PRM can solve complex tasks but require full state observability, a lot of time for problem solving and are not adapted to dynamic scene changes.

In recent years, there has been an increase in deep neural network (DNN) methods in several areas of science, technology, medicine, and more, along with significant advances in Deep Reinforcement Learning (DRL) techniques.

Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) methods are a promising approach to solving complex tasks in the real world with physical robots. RL/DRL methods are also used in real-world applications, such as improvements in the gaming industry for the Go game}, as well as in robotic applications for manipulation, goal achievement, Human-Robot Collaboration, and more.

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

## Agent (Reinforcement Learning)

- ○ Q-Learning
- ○ SARSA (State–action–reward–state–action)

## Agent (Deep Reinforcement Learning)

- ○ DQN (Deep Q-Network), DDQN
- ○ Deep-SARSA
- ○ DDPG (Deep Deterministic Policy Gradient)
- ○ TD3 (Twin Delayed Deep Deterministic)

*State* $S_t$    *Reward* $R_t$    **Agent**

$R_{t+1}$

$S_{t+1}$    **Environment**

*Action* $A_t$
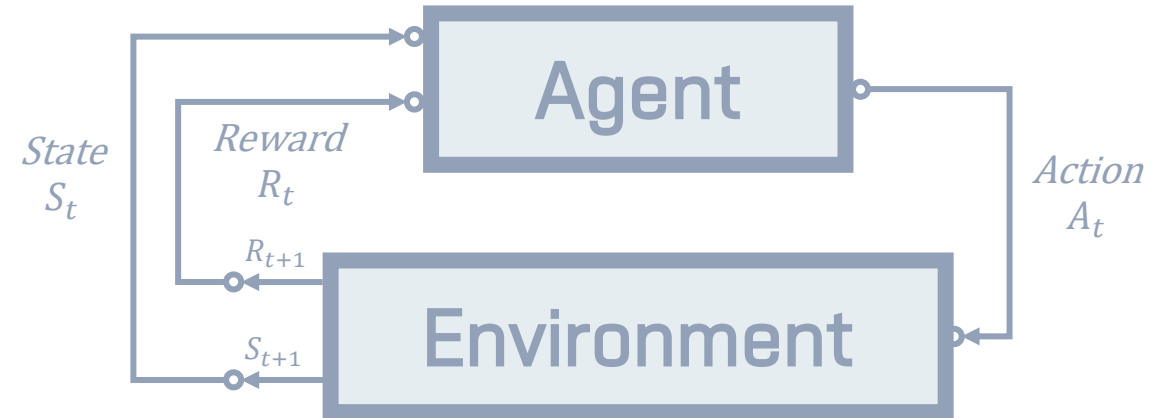
DRL overcomes the limitations of simple RL methods by combining parallel computation and embedded deep neural networks (DNN).

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

$p_1$

$y$

*Euclidean Distance*

$$d_t(p_0, p_1) = \sqrt{\sum_{i=1}^{n} (p_{0_i} - p_{1_i})^2}$$

$p_0$

$x$

$$S_t = \{x_+, x_-, y_+, y_-\}$$
$$A_t = \{0,1,2,3\}$$
$$R_t = from\ d_t?$$

**Agent**

*State* $S_t$

*Reward* $R_t$

$R_{t+1}$

$S_{t+1}$
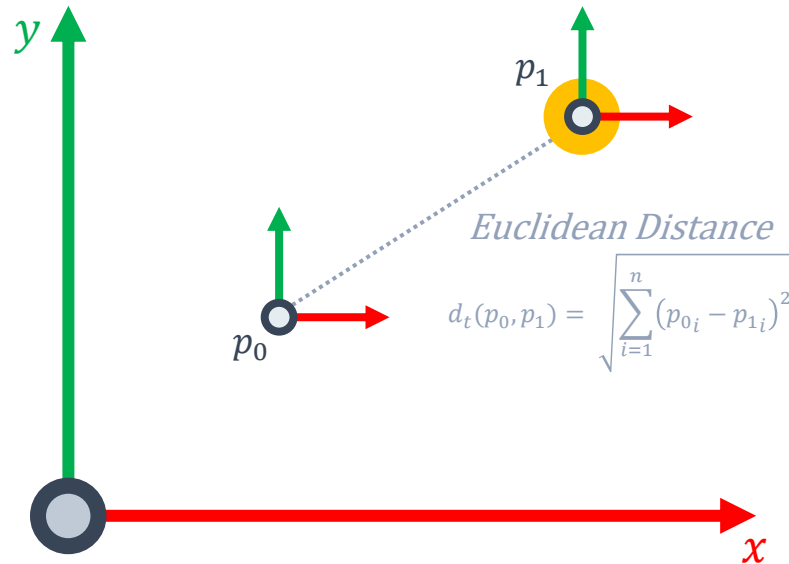
**Environment**

*Action* $A_t$
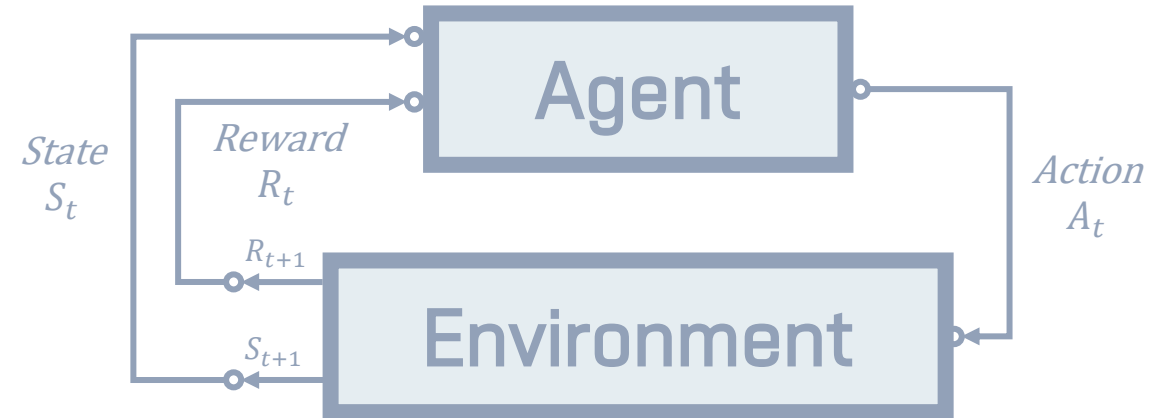
**Python Lib.**
OpenAI, Tensorflow, PyTorch, etc.

**Simulation Environment**
PyBullet, MuJoCo, ROS, Unity3D, NVIDIA
Omniverse, etc.

© RomanParak
2023    | Page 23/31

Planning Methods

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

Why do we need knowledge of motion planning?

Versatile Intelligent Robotics

Why do we need knowledge of motion planning?

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

Advanced Robotics in Healthcare

Why do we need knowledge of motion planning?

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

Robotics & Machine Perception

Why do we need knowledge of motion planning?

NETME Centre

FACULTY **institute**
OF MECHANICAL **of automation**
ENGINEERING **and computer science**

Mobile Robotics

Why do we need knowledge of motion planning?

NETME Centre

FACULTY institute
OF MECHANICAL of automation
ENGINEERING and computer science

Thank You!

Questions?

INDUSTRY 4.0 CELL

INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE