Planning in robotics of activity, path, motion, . . .

Václav Hlaváč

Czech Technical University in Prague
Center for Machine Perception (*bridging groups of the*)
Czech Institute of Informatics, Robotics and Cybernetics and
Faculty of Electrical Engineering, Department of Cybernetics

http://people.ciirc.cvut.cz/hlavac, hlavac@ciirc.cvut.cz

Outline of the talk:

- Holonomicity.
- Motion planning, formulation.
- Terminology, path vs. trajectory.
- Robotic planning as a spatial reasoning.
- Motion planning algorithms.
- Dummy 6.
Motion planning in industrial and mobile robotics

Determining where to move without hitting obstacles.
Three key questions in robotic planning

1. Where am I?  
   **Localization.**

2. Where am I going?  
   **Planning.**

3. How do I get there?  
   **Navigation.**

Are two given points connected by a path?
Holonomicity in robotics

- **Holonomicity** refers to the relationship between the controllable and total degrees of freedom of a given robot (or part thereof).
- **Holonomic**: if the controllable degrees of freedom is equal to the total degrees of freedom.
- **Non-holonomic**: if the controllable degrees of freedom are less than the total degrees of freedom.
- **Redundant robot**: if it has more controllable degrees of freedom than degrees of freedom in its task space.
Example: A car = non-holonomic

- Three degrees of freedom: its position in two axes, and its orientation relative to a fixed heading.
- Only two controllable degrees of freedom: acceleration/braking and the angle of the steering wheel.
- A car’s heading (the direction in which it is traveling) must remain aligned with the orientation of the car, or $180^\circ$ from it if the car is in reverse. It has no other allowable direction, assuming there is no skidding or sliding. Thus, not every path in phase space is achievable.
Approximation by a holonomic path

- For a car, not every path in the space is achievable;
- However, every path can be approximated by a holonomic path – this is called a (dense) homotopy principle (comes from mathematics, differential equations).
- The non-holonomicity of a car makes parallel parking and turning in the road difficult, but the homotopy principle says that these are always possible, assuming that clearance exists.
A human arm is holonomic

- A human arm is a holonomic.

- It is a redundant system because it has 7 degrees of freedom (3 in the shoulder - rotations about each axis, 2 in the elbow - bending and rotation about the lower arm axis, and 2 in the wrist, bending up and down (i.e. pitch), and left and right (i.e. yaw)).

- There are only 6 physical degrees of freedom in the task of placing the hand (x, y, z, roll, pitch and yaw), while fixing the seven degrees of freedom fixes the hand.
Holonomic locomotion

- Holonomic forms of locomotion allow vehicles to immediately move in any direction without needing to turn first.

- Example: Mecanum (Sweedish) wheel, e.g Holbot (Igor Kruhák, 2003).

- Counterexample: Segway (inverted pendulum principle), 2 DOFs of a robot, 3 DOFs of the environment.
Motivation task for motion planning

The motivation task

- **The task:**
  Transform the high-level task specification (provided by a human) into the low-level commands controlling the actuators.

- **The solution:**
  Motion planning algorithms provide the (geometric) path enabling to move a robot (or a manipulator gripper) from the start to the goal taking into account all operational constraints.

Asimo robot by Honda.

BMW spot welding.
**Motion planning, the problem formulation**

- **Motion planning** (a robotics term) is the process of breaking down a desired movement task into (discrete) motions satisfying given constraints (as not hitting obstacles, keeping speed limits) and possibly address optimality aspects.

- Known also as the navigation problem or piano mover’s problem.

- The geometric aspect of the task (spatial reasoning) induces use of methods from computational geometry.

A computational geometry example: The **Moving ladder problem**

- What is the longest ladder that can be moved around a right-angled corridor of unit width?

- For a straight, rigid ladder, the answer is \(2\sqrt{2}\), which allows the ladder to just pivot around the corner at a \(45^\circ\) angle.

![Diagram of a moving ladder around a corridor](image)
C-space, a reminder

We studied the configuration space in the “robot world representation” lecture.

Robot arm with two DOFs. The task is to move from the point (1) to the point (2) not touching the obstacles.
Piano mover’s problem

Given an open subset $U$ (free space) in $n$-dimensional $C$—space and two compact subsets $C_0$ (start) and $C_1$ (goal) of $U$, where $C_1$ is derived from $C_0$ by a continuous motion, is it possible to move $C_0$ to $C_1$ while remaining entirely inside $U$?

A side effect of the algorithm: the 3D trajectory and the “piano” 3D configuration in any trajectory point.

References

Piano mover’s problem, a video example

Courtesy: Jan Faigl et al., The Czech Technical University in Prague
Piano mover’s problem, a formal guarantee

Given:

- $p$ – dimension of the configuration space, abbreviated $C$–space.
- $m$ – number of polynomials describing $C_{\text{free}}$.
- $d$ – Maximal degree of polynomials (in the preceding item).

Theorem (which is not very useful practically):
A path (if it exists) can be found in time exponential in $p$ and polynomial in $m$ and $d$.

Terminology: path vs. trajectory

- **Note:** Terms *path*, *trajectory* are often confused. They are used as synonyms informally.

- **Path** is an ordered locus of points in the space (either joint or operational), which the robot should follow.
  - Path provides a pure geometric description of motion.
  - Path is usually planned globally taking into account obstacle avoidance, traversing a complicated maze, etc.

- **Trajectory** is a path plus velocities and accelerations in its each point.
  - A design of a trajectory does not need global information, which simplifies the task significantly.
  - The trajectory is specified and designed locally. Parts of a path are covered by individual trajectories.
  - It is often required that pieces of trajectories join smoothly, which induces that a single trajectory design takes into account only neighboring trajectories from the path.
Robot motion planning, an overview

Path planning (global)

- Geometric path.
- Issues: obstacle avoidance, shortest path.

Trajectory generating (local)

- The path planning provides the input – the chunk of a path usually given as a set of points defining the trajectory.
- “Approximate” the desired path chunk by a class of polynomial functions and
- generate a sequence of time-based “control set points” for the control of manipulator from the initial configuration to its destination.
Problem solving vs. planning

Basic problem solving

- Problem solving (search in the space space, a basic tool in AI) and planning have a similar core. However, they are considered different.
- Basic problem solving searches a state-space of possible actions, starting from an initial state and following any path to the goal state.

Planning differs from the basic problem solving in:

1. Planning “opens up” the representation of states, goals and actions so that the planner can deduce direct connections between states and actions.
2. The planner does not have to solve the problem in order. It can suggest actions to solve any sub-goals at anytime.
3. Planners assume that most parts of the world are independent. Decomposition to subproblems into practically sized chunks.
Path planning (dealt in this lecture)

◆ Goals:

- **Achieve high-level goals**, e.g.:
  Assemble/disassemble the engine. Build a map of the hallway. Find a collision free path for the robot from one configuration to another configuration.

- **Compute motion strategies**, e.g.:
  Geometric paths; Sequence of sensor-based motion commands. Time-parameterized trajectories.

◆ Path planning is a difficult search problem.

- The involved task has an exponential complexity with respect to the degrees of freedom (controllable joints).

- With industrial robots, path planning has been often solved by a computer as human operators plan the paths.
Trajectory generating (covered in a separate lecture)

- Planned path is typically represented by via-points.
  - Via-points = sequence of points (or end-effector poses) along the path.
- Trajectory generating = creating a trajectory connecting two or more via points.
  - Trajectory generating approximates / interpolates the path.
  - In industrial settings, a trajectory is performed by a human expert and later played back (by teach-and-playback).
  - Recent research utilizes as the input several tens of trajectories performed by human experts. They vary statistically.
  - Machine learning techniques are used to create the final trajectory.
Robotic planning as spatial reasoning

- Application of earlier search approaches from artificial intelligence. (A*, stochastic search, etc.)
- Search in geometric structures ⇒ **Spatial reasoning**.
- A more complex variant: **Spatiotemporal reasoning**.
- Challenges:
  - Continuous state space.
  - Large dimensional space.

**The main strategy in motion planning:**

- Reduction to point robot.
- Configuration space.
- Solution: search problem, usually graph search.
Collision and proximity queries

Requires geometric reasoning of spatial relationships among objects, often in a dynamic environment.

- Collision detection
- Contact points and normals
- Closest points and separation distance
- Penetration depth
Collision and proximity computations

- A key component of motion planning algorithms (estimated 90% of a total run time).
- Widely used in CAD/CAM, simulation and virtual prototyping.
- Supported in robot simulation and CAD systems
- Studied in academia for 30+ years.
- Widely used recent implementations:
  - FCL (The Flexible Collision Library, University of Northern Carolina, Chapel Hill).
  - MoveIt! (part of ROS).
Motion planning algorithms, two main groups

Optimization-based algorithms

Random sampling-based algorithms

The green circle denotes the start. The orange circle denotes the goal.