EN530.678 Nonlinear Control and Planning in Robotics
Lecture 1: Introduction

January 29, 2018

Prof: Marin Kobilarov

Robotic Systems.

This course is concerned with the control and motion planning of dynamical systems. While the theory and developed methods will be generally applicable, our primary application area will be robotics. We can consider the following general classes of robots:

- *robot manipulators*: fixed base, typically a chain of links connected by joints
- *mobile robots*: ground wheeled/legged robots, underwater/surface vehicles, aerial robots (helicopters, fixed-wing planes), space robots (satellites, other spacecraft)
  - *mobile manipulators*: manipulators on a moving base (ground wheeled robots, underwater robots, aerial robots); humanoid robots

We are interested in the planning and control of mobile robots, which could include mobile manipulators.

System Representation.

We next define the problems of interest more formally (although still at a high level). We will describe the *configuration* of a system (e.g. position, orientation, joint angles) using the variables

$$q = (q_1, q_2, \dots, q_n)$$

and its velocity by

$$\dot{q} = (\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n).$$

The system is actuated using *control inputs*

$$u = (u_1, u_2, \dots, u_m),$$

such as joint motor torques, engine output, rocket burn rate, etc... The *state* of the system is often denoted by

$$x = (q, \dot{q})$$

although in some cases the velocity $\dot{q}$ will be replaced by another variable v, called *reduced-* or *pseudo*-velocity, which might have less components than $\dot{q}$. This occurs when there are constraints which restrict the possible directions of motion. We will assume that the state evolves according to a known *deterministic* differential equation (ODE)

$$\dot{x}(t) = f(t, x(t), u(t)).$$

In some cases a more general ODE will be employed:

$$\dot{x}(t) = f(t, x(t), u(t), p, w(t)),$$

where p are constant *parameters* such as mass, joint links, etc..., and $w(t)$ is *process noise* accounting for unknown/uncertain part of the dynamics.

Finally, we assume that we can observe part of the state through *measurements* or *outputs* denoted by y and specified through the function

$$y(t) = h(t, x(t))$$

or, in the more general form

$$y(t) = h(t, x(t), p, v(t)),$$

where $v(t)$ is measurement noise.

Planning and Control Tasks.

Our goal is to enable such dynamic systems to achieve a desired behavior given their dynamics, uncertainty, and available measurements. This can be broadly defined as solving and control and motion planning problems. In particular, we have the following problems:

- Regulation/Stabilization
  - control to a desired state x_d: find $u(t)$ such that $x(t) \to x_d$
  - applicable when the system can easily reach that state with little effort
- Trajectory Tracking
  - control along a specified trajectory $x_d(t)$: find $u(t)$ so that $x(t) \to x_d(t)$
  - assumes $x_d(t)$ was already computed
- Trajectory generation (motion planning / trajectory optimization)
  - design a trajectory $x_d(t)$ which optimizes a given metric, satisfies given constraints, and in some cases be robust to uncertainties
  - essentially a *constrained optimal control* problem:

$$J = \phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x(t), u(t), t)dt, \tag{1}$$

subject to:

$$x(t_0) = x_0, \quad x(t_f) \text{ and } t_f \text{ free} \tag{2}$$
$$\dot{x}(t) = f(x(t), u(t), t) \tag{3}$$
$$c(x(t), u(t), t) \leq 0, \text{ for all } t \in [t_0, t_f] \tag{4}$$
$$\psi(x(t_f), t_f) \leq 0, \tag{5}$$

where

* $[t_0, t_f]$ - time horizon, x_0-initial state

* L - trajectory cost: e.g. control effort, energy, time, distance
* ϕ - terminal cost: e.g. reaching a desired region
* c - trajectory constraints: e.g. joint angle bounds, control input bounds, forbidden regions in state space due to environmental obstacles, or dangerous operation zones
* ψ - terminal constraint defines algebraically a goal region

Underactuation and Nonholonomic Constraints.

Next, we specify what type of system we are interested in and why they deserve special treatment. The dynamics of a robotic system can often be written as

$$M(q)\ddot{q} + b(q, \dot{q}) = B(q)u, \tag{6}$$

where M is the *mass matrix*, b are called the *bias* forces, and B is the *control transformation matrix*. Assume that the systems starts at some state $x_0 = (q_0, \dot{q}_0)$ and is required to reach a desired state $x_d = (q_d, 0)$. When B is full-rank, e.g. in the case of a standard manipulator, it is typically easy to design a control law (assuming there are no control/state constraints or uncertainties). One of the most effective laws is the *computed torque* control law:

$$u = B(q)^{-1} \left[M(q)(K_p(q_d - q) + K_d\dot{q} + b(q, \dot{q})\right]$$

which when applied causes the configuration to evolve according to

$$\ddot{q} = K_p(q_d - q) + K_d(0 - \dot{q})$$

which is a linear ODE for which one can show that the system is guaranteed to reach x_d.

But when $B(q)$ is not full rank such a control law is not applicable. This occurs when we have *underactuation*: e.g. there are simply less controls than degrees of freedom.

Nonholonomic constraints pose another type of complication in control, i.e. the possible directions of motion (i.e. velocities) are restricted. In this case, the allowed velocities may be written as

$$\dot{q} = G(q)v,$$

where $v \in \mathbb{R}^m$ denotes the *reduced velocity*, i.e. we have that $m < n$. As we will see (through a formal derivation) the dynamics for such systems is given by

$$M(q)\dot{v} + b(q, v) = B(q)u, \tag{7}$$

As we will see, given such dynamics we will be able to select u to achieve a desired velocity v_d but achieving the full state (q_d, v_d) will still be non-trivial.

Underactuation and velocity constraints are common in robotics:

- by design, e.g. to achieve efficient locomotion: wheeled vehicles, most aerial vehicles, some surface vessels and underwater vehicles
- from interaction with the environment: rolling, sliding, manipulating with fingers
- from actuator failure

- from conservation laws: e.g. spacecraft attitude control using momentum wheels, based on total angular momentum conservation

We will be concerned designing control algorithms and trajectory planning algorithms that apply to standard (e.g. fully-actuated) systems, to underactuated systems, and also to certain classes of Nonholonomic systems.

Other systems. Finally, note that the control methods developed are applicable to any system consistent with our modeling approach. This could include electric circuits, biological systems, economic models, etc...