2 What Does An Optimal Control Algorithm Look Like?

There are really three standard approaches to optimal control, roughly:

1. direct approach (discretize in time, then optimize the resulting finite dimensional problem)
2. indirect approach (optimize, then discretize resulting ordinary or partial differential equation to find a numeric representation)
3. functional analysis approach (work directly with optimization over curves, include numerics at each step)

We will focus on the third approach.

2.1 Notation

Symbol	Notes
$\bar{\xi} = (\bar{x}, \bar{u})$	$\bar{\xi} \in \mathscr{L}$ is a curve—a pair of $\bar{x}$ and $\bar{u}$ not necessarily satisfying any properties except smoothness with respect to time t.
$\xi = (x, u)$	$\xi \in \mathscr{T}$ is a *trajectory*—a pair of x and u that satisfy the differential equation $\dot{x} = f(x, u)$. In the context of iterative methods in optimal control, ξ will be the projection of a $\bar{\xi}$. That is $\xi = \mathscr{P}(\bar{\xi})$
$\mathscr{P}(\cdot)$	$\mathscr{P} : \mathscr{L} \to \mathscr{T}$ is a projection defined by $$\mathscr{P}(\bar{\xi}) = \begin{cases} \dot{x} = f(x, u) \quad x(0) = x_0 \\ u = \bar{u} + K(\bar{x} - x). \end{cases}$$
$\bar{\zeta} = (\bar{z}, \bar{v})$	$\bar{\zeta} \in T_{\bar{\xi}}\mathscr{L}$ is a pair in the "tangent space" of the space of curves manifold at the curve $\bar{\xi}$.
$\zeta = (z, v)$	$\zeta \in T_{\eta}\mathscr{T}$ is a pair in the "tangent space" of the set of trajectories at the trajectory ξ.
J	$J(\bar{\xi})$ is the cost, most often of the form $J(\bar{\xi}) = \int_{t_0}^{t_f} \ell(\bar{\xi}(t))dt + m(\bar{x}(t_f))$, where ℓ is a "running cost".

None of this should feel even remotely familiar at this point, but you should definitely come back to this page over and over as you make progress in the course!

The algorithm that uses these ideas is the following.

Data:
$\bar{\xi}_0 \in \mathscr{L}$ or $\xi_0 \in \mathscr{T}$,
$J(\xi)$
$\alpha \in (0, \frac{1}{2})$, (a real number)
$\beta \in (0, 1)$, (a real number)
and ϵ (a small real number)
i=0
while $||\zeta_{i-1}|| > \epsilon$ **do**
    compute descent direction;

$$\zeta_i = \arg \min_{\zeta \in T_{\mathscr{P}(\xi_i)} \mathscr{T}} DJ(\mathscr{P}(\xi_i)) \circ D\mathscr{P}(\xi_i) \circ \zeta + \frac{1}{2}||\zeta||^2$$

    compute line search using ζ_i;
    $n = 0$;
    while $J(\mathscr{P}(\xi_i + \gamma\zeta_i)) > J(\mathscr{P}(\xi_i)) + \alpha\gamma DJ(\mathscr{P}(\xi_i)) \circ D\mathscr{P}(\xi_i) \circ \zeta_i$ **do**
        $n = n + 1$;
        $\gamma = \beta^n$;
    end
    $\xi_{i+1} = \mathscr{P}(\xi_i + \gamma\zeta_i)$;
    $i = i + 1$
end

Algorithm 2.1: An iterative optimal control algorithm

Probably you do not look at this algorithm and understand what it is doing. But what operations does it assume you have at your disposal?

1. The term $\gamma\zeta_i$ implies scalar multiplication must make sense;
2. The term $\xi_i + \gamma\zeta_i$ implies that addition must make sense;
3. The DJ term implies that we should be able to linearize the objective function;
4. The $\mathscr{P}(\cdot)$ term implies we are able to project *infeasible* curves $\bar{\xi}$ to *feasible* trajectories ξ;
5. The $\mathscr{P}(\cdot)$ term will end up using something called Riccati equations to compute projections;
6. The $D\mathscr{P}(\cdot)$ term implies that we should be able to linearize the projection, which will imply linearizing the dynamics.
7. The ξ and ζ imply we must be able to evaluate differential equations;
8. The structure of the algorithm implies a termination condition $||\zeta_{i-1}|| > \epsilon$.