Receding Horizon Control

María M. Seron

September 2004

The Receding Horizon Control Principle

Fixed horizon optimisation leads to a control sequence $\{u_i, \ldots, u_{i+N-1}\}$, which begins at the current time i and ends at some future time $i + N - 1$.

This fixed horizon solution suffers from two potential drawbacks:

(i) Something unexpected may happen to the system at some time over the future interval $[i, i + N - 1]$ that was not predicted by (or included in) the model. This would render the fixed control choices $\{u_i, \ldots, u_{i+N-1}\}$ obsolete.

(ii) As one approaches the final time $i + N - 1$, the control law typically "gives up trying" since there is too little time to go to achieve anything useful in terms of objective function reduction.

The Receding Horizon Control Principle

The above two problems are addressed by the idea of *receding horizon optimisation*.

This idea can be summarised as follows:

(i) At time i and for the current state x_i, solve an optimal control problem over a fixed future interval, say $[i, i + N - 1]$, taking into account the *current* and *future* constraints.

(ii) Apply only the first step in the resulting optimal control sequence.

(iii) Measure the state reached at time $i + 1$.

(iv) Repeat the fixed horizon optimisation at time $i + 1$ over the future interval $[i + 1, i + N]$, starting from the (now) current state x_{i+1}.

In the absence of disturbances, the state measured at step (iii) will be the same as that predicted by the model.

Nonetheless, it seems prudent to use the *measured state* rather than the predicted state just to be sure.

The above description assumes that the state is measured at time $i + 1$.

In practice, one would use some form of observer to estimate x_{i+1} based on the available data.

More will be said about the use of observers in the next lecture.

For the moment, we will assume that the full state vector is measured and we will ignore the impact of disturbances.

The Receding Horizon Control Principle

If the model and objective function are time invariant, then the same input u_i will result whenever the state takes the same value.

That is, the receding horizon optimisation strategy is really a particular *time-invariant state feedback control law:*

PSfrag replacements

u_k $x_{k+1} = f(x_k, u_k)$ x_k

RHC
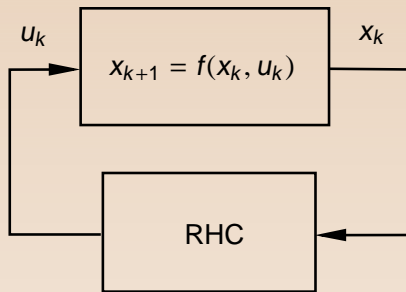
In particular, we can set $i = 0$ in the formulation of the open loop control problem.

The Receding Horizon Control Principle

More precisely, at the current time, and for the current state x, we solve:

$$\mathcal{P}_N(x): \qquad V_N^{\text{OPT}}(x) \triangleq \min V_N(\{x_k\}, \{u_k\}), \tag{1}$$

subject to:

$$x_{k+1} = f(x_k, u_k) \quad \text{for } k = 0, \ldots, N-1, \tag{2}$$

$$x_0 = x, \tag{3}$$

$$u_k \in \mathbb{U} \quad \text{for } k = 0, \ldots, N-1, \tag{4}$$

$$x_k \in \mathbb{X} \quad \text{for } k = 0, \ldots, N, \tag{5}$$

$$x_N \in \mathbb{X}_f \subset \mathbb{X}, \tag{6}$$

where

$$V_N(\{x_k\}, \{u_k\}) \triangleq F(x_N) + \sum_{k=0}^{N-1} L(x_k, u_k). \tag{7}$$

The sets $\mathbb{U} \subset \mathbb{R}^m$, $\mathbb{X} \subset \mathbb{R}^n$, and $\mathbb{X}_f \subset \mathbb{R}^n$ are the *input*, *state* and *terminal* constraint set, respectively.

All sequences $\{u_k\} = \{u_0, \ldots, u_{N-1}\}$ and $\{x_k\} = \{x_0, \ldots, x_N\}$ satisfying the constraints (2)–(6) are called *feasible* sequences.

A pair of feasible sequences $\{u_0, \ldots, u_{N-1}\}$ and $\{x_0, \ldots, x_N\}$ constitute a *feasible solution*.

The functions F and L in the objective function (7) are the *terminal state weighting* and the *per-stage weighting*, respectively.

In the sequel we make the following assumptions:

- f, F and L are continuous functions of their arguments;
- $\mathbb{U} \subset \mathbb{R}^m$ is a compact set, $\mathbb{X} \subset \mathbb{R}^n$ and $\mathbb{X}_f \subset \mathbb{R}^n$ are closed sets;
- there exists a feasible solution to problem (1)–(7).

Because N is finite, these assumptions are sufficient to ensure the existence of a minimum by Weierstrass' theorem.

Typical choices for the weighting functions F and L are quadratic functions of the form

$$F(x) = x^\mathsf{T} P x \quad \text{and} \quad L(x, u) = x^\mathsf{T} Q x + u^\mathsf{T} R u,$$

where $P = P^\mathsf{T} \geq 0$, $Q = Q^\mathsf{T} \geq 0$ and $R = R^\mathsf{T} > 0$.

More generally, one could use functions of the form

$$F(x) = \|Px\|_p \quad \text{and} \quad L(x, u) = \|Qx\|_p + \|Ru\|_p,$$

where $\|y\|_p$ with $p = 1, 2, \ldots, \infty$, is the p-norm of the vector y.

The Receding Horizon Control Principle

Denote the minimising control sequence, which is a function of the current state x_i, by

$$\mathscr{U}_{x_i}^{\text{OPT}} \triangleq \{u_0^{\text{OPT}}, u_1^{\text{OPT}}, \ldots, u_{N-1}^{\text{OPT}}\}; \tag{8}$$

then the control applied to the plant at time i is the first element of this sequence, that is,

$$u_i = u_0^{\text{OPT}}. \tag{9}$$

Time is then stepped forward one instant, and the above procedure is repeated for another N-step-ahead optimisation horizon.

The first element of the new N-step input sequence is then applied, and so on.

The above procedure is called *receding horizon control* (RHC).

The Receding Horizon Control Principle
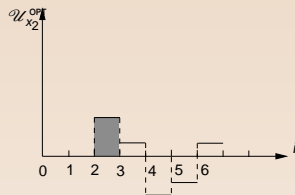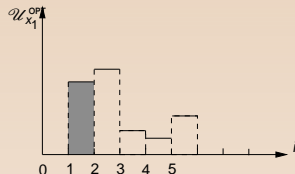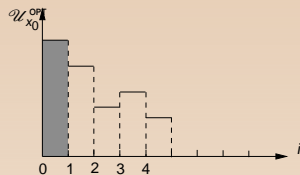
The figure illustrates the RHC principle for horizon $N = 5$.

Each plot shows the minimising control sequence $\mathcal{U}_{x_i}^{\mathsf{OPT}}$ given in (8), computed at time $i = 0, 1, 2$.

Note that only the shaded inputs are actually applied to the system.

We can see that we are continually looking ahead to judge the impact of current and future decisions on the future response.

The above receding horizon procedure *implicitly* defines a time-invariant control policy $\mathcal{K}_N : \mathbb{X} \to \mathbb{U}$ of the form

$$\mathcal{K}_N(x) = u_0^{\text{OPT}}. \tag{10}$$

The receding horizon controller is implemented in closed loop as follows.

The UNIVERSITY
of NEWCASTLE
AUSTRALIA

Centre for Complex Dynamic
Systems and Control
CDSC
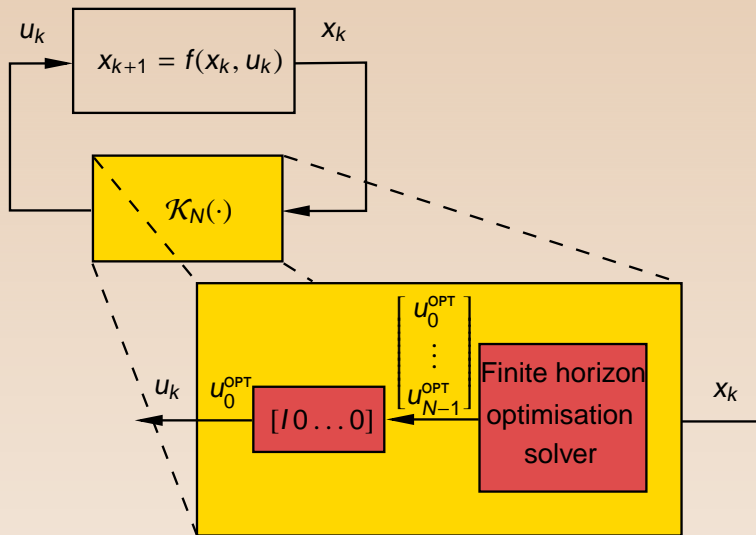
Figure: Receding horizon control

Note that the strict definition of the function $\mathcal{K}_N(\cdot)$ requires the minimiser to be unique.

Most of the problems treated in this course are convex and hence satisfy this condition.

One exception is the "finite alphabet" optimisation case that will be discussed on Day 4, where the minimiser is not necessarily unique.

However, in such cases, one can adopt a rule to select one of the minimisers.

It is common in receding horizon control applications to compute *numerically*, at time i, and for the current state $x_i = x$, the optimal control move $\mathcal{K}_N(x)$. In this case, we call it an *implicit receding horizon optimal policy*.

In some cases, we can explicitly evaluate the *control law* $\mathcal{K}_N(\cdot)$. In this case, we say that we have an *explicit receding horizon optimal policy*.

We will expand on the above skeleton description of receding horizon optimal constrained control as the course evolves.

For example, we will treat linear constrained problems in the next lecture. When the system model is linear, the objective function quadratic and the constraint sets polyhedral, the fixed horizon optimal control problem $\mathcal{P}_N(\cdot)$ is a quadratic programme of the type presented on Day 2. On Day 3 we will study the solution of this quadratic program in some detail.

If, on the other hand, the system model is nonlinear, $\mathcal{P}_N(\cdot)$ is, in the general case, nonconvex, so that only local solutions are available.