**Model Predictive Control**

Model Predictive Control (MPC) is a control approach which contains different algorithms based on specific key principles:

* Predicting the system future behavior
* Solving an optimization problem
* Receding horizon

The key components of MPC are:

* System model
* Optimization problem

**Strategy**

1. MPC at each sampling time k:
   1. performs a ***prediction*** of the future behavior of the system over a time horizon N, through a model of the system and its past states and inputs
   2. obtains an optimal control sequence by solving an ***optimization*** problem
   3. applies only the first term of the input sequence to the system and obtain the resulting state
2. At new sampling time, the whole procedure is repeated, shifting forward the time horizon (***receding horizon***) and taking the current state as the initial state for the new prediction.

**Assumptions**

The standard MPC formulation is valid under some assumption:

* the system is linear
* uncertainties do not affect the system
* the system is controllable
* the system is observable

**Discussion**

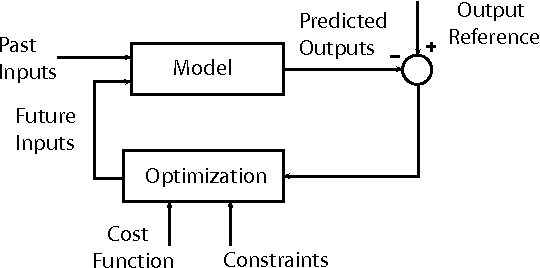
**Pros**

* Extremely flexible control design
* Performance oriented and easy tunable
* Constraint handling
* Easy generalized to MIMO system

**Cons**

* Suitable only for linear models and convex optimization
* Slower due to the complexity of the optimization
* Suboptimality

**Block scheme**



**System Model**

**Modelling**

System modelling it is important for prediction: the simplest model that gives accurate enough prediction (10-20% out from the steady state).

MPC uses a linear model of the system, most of the time in the state-space representation in continuous time domain:

where: state

output

input

Note: in the state-space usually since the input does not affect the output.

A real system operates in continuous time, but since the controller operates in discrete time, the system model has to be discretized.

**Prediction**

The state prediction is obtained recursively using the state-space equation:

starting from the initial (current) state, till the prediction horizon:

It is possible splitting the known part (based on current and past measurements) and the unknown part (based on the input sequence choice), and stack the vectors:

where:

**Optimization**

MPC is based on precise numerical optimum, which can be defined arbitrarily through a cost function, subject to certain constraints:

According to the form of the optimization problem, it can be reformulated into class of optimization problems:

* Convex optimization
  + LP (Linear Programs) for unconstrained linear systems
  + QP (Quadratic Programs) for linear constrained systems
  + Nonlinear Programs for general systems
* Nonconvex optimization

**Objective/Cost function**

The objective function is called also cost/loss function since it implies a penalization of the terms involved, which represent some performance indices. Most of the time, these indices are in opposition from each other, then it consists in a tradeoff between optimality performance and stability.

**Penalty form**

The penalty is always a convex function. The common forms are:

* l2-norm: energy dissipation penalization
* l1-norm: fuel consumption penalization
* l∞-norm

**Performance indices**

* state:
* output tracking error:
* input activity:
* input rate:

**Constraints**

* State constraints:
* Input constraints:
* Input rate constraints:

**Optimal solution**

The solution of the optimization problem is the optimal control input sequence at the sample k:

The optimal input at generic time depends on the predicted state at the same time , obtained from the dynamic model , which depends on the initial state . Thus, this means the optimal input sequence is an open loop control since it depends only on the initial conditions.

**Receding Horizon Strategy**

Only the first element of the optimal input sequence is applied to the system:

At each sampling instant, the process of obtaining the first element of the optimal input sequence is performed, shifting the forward the prediction horizon and set the actual initial state .

Since in the prediction, the MPC is an open-loop control, the receding horizon introduce a feedback, continuously updating the prediction.

**Tuning parameters**

* Ts: It should be sufficiently small to track the system dynamic but not too small to ignore high frequency disturbances.
* Np: It should large enough to cover the significant system dynamic.
* Nc: It is usually between 1 and 5 since only the first value is applied to the system.
* Q: penalty proportional to the state/output.
* R: penalty proportional to the input; it decreases the original performances, leading to a slower response
* S: penalty proportional to the input rate.
* P: penalty proportional to the terminal state/output.

**Notes**

**Discrete/Continuous-time prediction model**

In standard application the optimization is solved periodically at time:

where T is the optimization sampling time, which depends on the time required by the computation of the optimum.

However, this choice could not match the prediction sampling time, which depends on the system dynamics and it is usually smaller with respect the optimization one:

where m is an integer.

It is also possible using continuous time model for prediction.

**Hard/Soft constraints handling**

**Terminal cost**

Due to the difference between predicted response and closed-loop response, there is no guarantee that a common MPC (receding horizon on a finite horizon), which can lead to instability.

This problem is solved by means of the infinite prediction horizon but defining a predicted input sequence in such a way that the number of optimization variables are finite.

**Dual mode approach**

This is achieved through the ***dual mode*** predictions:

* Mode I: the input is obtained by the optimization problem over a finite horizon
* Mode II: the input is obtained by a stabilizing feedback law over an infinite horizon

With this approach the infinite horizon cost can be evaluated explicitly over mode I, in the optimization problem:

This is achieved by choosing an appropriate value of the terminal weighting matrix P, so that the terminal cost is equal to the cost over the mode II. This matrix is the solution of the Lyapunov equation:

which has a unique solution if and only if the eigenvalues of lie inside the unit circle.

**Terminal constraint**

**Unconstrained optimization**

If no inequalities constraints are present, the optimization problem has a closed-form solution:

which corresponds to a feedback control approach.

**Appendix: Notation**

continuous time instant

discrete time instant

sampling time

prediction horizon

control horizon

prediction for i future samples ahead at time k

weight matrices

**Appendix: Properties**

**Reachability/Controllability and Observability**

Controllability and observability are dual aspects of the same problem.

A system is controllable if:

where: controllability matrix

A system is observable if:

where: observability matrix

**Feasibility and Optimality**

**Robustness and Performances**

**MPC Overview**

The standard MPC formulation produces optimal performances only under the assumptions of linearity of the system and the absence of uncertainties.

**Nonlinearity**

The most of systems are nonlinear, then the LTI (Linear Time Invariant) approximation can produces an inaccurate control strategy. According to the performances required, it can be applied:

* ***LTV MPC*** (Linear Time Varying MPC): the system model is still linear, but the system matrices are time dependant
* ***NMPC*** (Nonlinear MPC): the system model is nonlinear, implying the loss of convexity

**Uncertainties**

In the presence of uncertainties, according to the properties of these disturbances can be followed two approaches:

* ***RMPC*** (Robust MPC): If the disturbance is bounded it is possible to design a robust MPC that ensure that the constraints are satisfied for all the possible disturbances sequence.
* ***SMPC*** (Stochastic MPC): If the disturbance is unbounded it is possible to design a stochastic MPC to ensure that the constraints are satisfied on a specific probability.

Some approaches to RMPC and SMPC are:

* Min-max MPC: The optimization problem is performed with respect to all possible trajectory evolutions
* Constraint Tightening MPC: The optimization is subject to enlarged constraints by a given margin, so that the trajectory feasibility is met
* Tube-based MPC: The optimization is done over a nominal dynamic model, while a feedback controller ensures the convergence of the actual state to the nominal one.
* Multi-stage MPC

|  |  |  |
| --- | --- | --- |
|  | **PRO** | **CONS** |
| MPC | Convex optimization | Performance degraded as assumptions fading |
| LTV MPC | Faster than NMPC |  |
| NMPC |  |  |
| Min-max MPC |  | Computational expensive |
| Constraint Tightening MPC |  |  |
| Tube-based MPC |  |  |
| Multi-stage MPC |  | Computational expensive (exponentially with the uncertainty number and Hp) |