WSEAS TRANSACTIONS on SYSTEMS
DOI: 10.37394/23202.2021.20.19

Dynamic Programming in Data Drivenmodel Predictive Control ?

WANG JIANHONG
School of Electronic Engineering and Automation, Jiangxi University of Science and Technology, Ganzhou, CHINA

Abstract: In this short note, one data driven model predictive control is studied to design the optimal control sequence. The idea of data driven means the actual output value in cost function for model predictive control is identi_ed through input-output observed data in case of unknown but bounded noise and martingale di_erence sequence. After substituting the identi_ed actual output in cost function, the total cost function in model predictive control is reformulated as the other standard form, so that dynamic programming can be applied directly. As dynamic programming is only used in optimization theory, so to extend its advantage in control theory, dynamic programming algorithm is proposed to construct the optimal control sequence. Furthermore, stability analysis for data drive model predictive control is also given based on dynamic programming strategy. Generally, the goal of this short note is to bridge the dynamic programming, system identi_cation and model predictive control. Finally, one simulation example is used to prove the e_ciency of our proposed theory.

Key words: Model predictive control; Data driven; Dynamic programming; Nonlinear estimation; Stability analysis

Received: January 10, 2021. Revised: May 21, 2021. Accepted: July 13, 2021. Published: July 21, 2021.

1 Introduction

Model predictive control(MPC)is one special form of suboptimal control problem, whose control objective is to keep the state of a system near some desired points. MPC combines elements of several ideas that we have put forth, for example, certainty equivalent control, multistage lookahead and rollout algorithms. MPC tends some applied properties for classical linear quadratic control, i.e. there are two main reasons for replacing that classical linear quadratic control by MPC. (1) The considered system may be nonlinear, and using for control purposes a model that is linearized around the desired point may be inappropriate. (2) There may be control and state constraints, which are not handled adequately through a quadratic penalty on state and control. The solution obtained from a linear quadratic model, is not suitable for this , because the quadratic penalty on state and control tends to blur the boundaries of the constraints. Generally, MPC converts one optimal control problem into one numerical optimization problem with equality or inequality constraints, which correspond to the control and state constraints. Moreover, when the considered system is either deterministic, or else it is stochastic, it is replaced with a deterministic version by using typical values in place of all uncertain quantities, such as the certainty control approach in implementing MPC. More specifically, at each stage, an optimal control problem is solved over a fixed length horizon, starting from the current state. The first component of the corresponding optimal policy is then used as the control of the current stage, while the remaining components are discarded. The optimization process is then repeated at the next stage, once the next state is revealed or the optimization algorithm is terminated iteratively.

From above detailed description on MPC, MPC corresponds to one numerical optimization problem, whose cost function or loss function is one error error value between the actual output and its desired output reference. In reality, the desired output is given, but the actual output is unknown in priori, so firstly we need to model the considered system and collect its actual output through persistly exciting the system by one appropriate input signal. It means the considered system is identified and used to calculate the actual output. There are two modelling approaches, used to identify the considered system, i.e. first principle and system identification. The first principle needs lots of priori information about the considered system, such as Newton law, mathematical or physical laws, etc. Then main essence of

---

⋆ This paper was not presented at any IFAC meeting. Corresponding author Wang Jianhong.

Email address: wangjianhong@nuaa.edu.cn (Wang Jianhong).

system identification is to excite the considered system, then use these collected input-output data to identify or estimate the unknown parameters, as the parameters are estimated online and used to describe the considered system, whatever in open loop or closed loop condition. The advantage of the second system identification approach is that no any priori information are needed, but only input-output data. In this big data period, this requirement for input-output data is tolerable. Roughly speaking, when to obtain the actual output in cost function for MPC, the input-output data , corresponding to the considered system in open loop or closed loop, are collected to identify the system through some statistical methods, for example, least squares method, maximum likelihood method, etc. Then the identified system is applied to express or describe the actual output, so the actual output depends on the accuracy of the identified system. In practice, system identification is a well-developed technique for estimating system parameters from operational data, typically taken during dedicated system testing or excitation, so system identification is also named as data driven.

Due to the application of system identification into MPC or other control strategy, such as adaptive control, internal model control and robust control, a new concept-identification for control was proposed in 2010s. Here we give a concise introduction or contribution on identification for control. In case of the unknown but bounded noise, one bounded error identification is proposed to identify he unknown systems with time varying parameters (Bravo J M, Alamo T, Camacho E F,2016). Then one feasible parameter set is constructed to include the unknown parameter with a given probability level. In (Bravo J M, Alamo T, Vasallo M, 2017), the feasible parameter set is replaced by one confidence interval, as this confidence interval can accurately describe the actual probability that the future predictor will fall into the constructed confidence interval. The problem about how to construct this confidence interval is solved by a linear approximation/programming approach (Casini M, Garulli A, Vicino A, 2017), which can identify the unknown parameter only for linear regression model. According to the obtained feasible parameter set or confidence interval, the midpoint or center can be deemed as the final parameter estimation, further a unified framework for solving the center of the confidence interval is modified to satisfy the robustness. This robustness corresponds to other external noises, such as outlier, unmeasured disturbance (Cerone V, Lasserre J B, Piga D,2014). The above mentioned identification strategy, used to construct one set or interval for unknown parameter, is called as set membership identification, dealing with the unknown but bounded noise. There are two kinds of descriptions on external noise, one is probabilistic description, the other is deterministic description, corresponding to the unknown but bounded noise here (Milanese M, Novara C, 2004). For the probabilistic description on external noise,the noise is always assumed to be one white noise, and its probabilistic density function (PDF) is known in advance. On the contrary for deterministic description on external noise, the only information about noise is bound, so this deterministic description can relax the strict assumption on probabilistic description. In reality or practice, bounded noise is more common than white noise. Within the deterministic description on external noise, set membership identification is adjusted to design controllers with two degrees of freedom (Novara C, Formentin S, Savaresi S M,2016), it corresponds to data driven control or set membership control. Set membership control is applied to design feedback control in a closed loop system with nonlinear system in (Tanaskovic M, Fagiano L, Novara C, 2017), where the considered system is identified by set membership identification, and the obtained system parameter will be benefit for the prediction output. After substituting the obtained system parameter into the prediction output to construct one cost function , reference (Tanaskovic M, Fagiano L, Smith R,2014) takes the derivative of the above cost function with respect to control input to achieve one optimal input. Set membership identification can be not only applied in MC, but also in stochastic adaptive control (Vidyasagar M, Karandikar R L, 2008), where a learning theory -kernel is introduced to achieve the approximation for nonlinear function or system. Based on the bounded noise, many parameters are also included in known intervals in prior, then robust optimal control with adjustable uncertainty sets are studied in (Zhang X, Kamgarpour M, Georghiou A, 2017), where robust optimization is introduced to consider uncertain noise and uncertain parameter simultaneously. To solve the expectation operation with dependence on the uncertainty, sample size of random convex programs is considered to replace the expectation by finite sum (Zhang X, Sergio Grammatico, John Lygeros, 2015). In recent years, the first author also studies the application of set membership into MPC, for example, bounded error identification is applied in MPC (Wang Jianhong, 2018), and to guarantee the obtained interval predictor to be a minimum interval predictor, two optimal vectors, used to adjust the width of the obtained interval predictor are suggested to be piecewise affine forms. To apply interval prediction model into robust MPC, min-max optimization problem is solved by Gauss-Seidel algorithm (Wang Jianhong,2019), and convergence of this Gauss-Seidel algorithm is provided too only through our own derivations.

In this short note, we continue to do this research on applying system identification into MPC,i.e. data driven MPC equals to system identification for MPC. Consider one first order discrete time nonlinear dynamic control system, its output must be needed in the cost function for MPC, so firstly we need to construct the actual output for the considered nonlinear dynamical system. But here it is very different with the references, as the nonlinear dynamical system is considered. To implementing the proposed MPC well, that actual output, corresponding to the nonlinear dynamical system, is identified or constructed by our own derivations. Moreover, our own derivations belong to the idea of system identification, i.e. the input-output data are proposed to construct the actual output directly, not to identify the nonlinear system parameter. Roughly speaking, data are used to describe the actual output directly, thus avoiding to estimate the nonlinear system parameter. The process of using the input-output data to denote the actual output directly coincides to the essence of data driven. After substituting the obtained actual output in the cost function for MPC, the idea of dynamic programming is applied to solve the optimal input value. Dynamic programming deals with situations, where decisions are made in stages (D P Bertsekas, 2019). The outcome of each decision may not b fully predictable, but can be anticipated to some extent before the next decision is made. As MPC formulates the problem of designing optimal controller into one constrain optimization problem, so dynamic programming can be applied to analyze the minimum principle of this constrain optimization, which corresponds to one general case. Moreover, after defining one concept of stability for MPC, stability condition is also derived through dynamic programming. To the best of our knowledge, dynamic programming is only studied in the numerical optimization theory, not in control science yet. Only in reference (Francesco Borrelli, Mato Baotic, ALberto Bemporad,2005), dynamic programming is firstly introduced to balance the desire for low present cost with the undesirability of high future cost for constrain optimal control. Therefore,this short note is the first analysis, regarding dynamic programming into data driven MPC¿ Generally, the main contributions of our short note are formulated as follows. (1) Construct the actual output directly by input-output data for cost function in MPC, (2) Reformulate one optimal control as one constrain optimization problem, (3) The minimum principle and stability analysis are presented for MC through dynamic programming.

This short note is organized as follows. In section 2, one first order discrete time nonlinear dynamical control system is considered, and some preliminaries are formulated about the noise for the nonlinear function. In section 3, the estimate of the unknown nonlinear function is studied. Two estimations for the unknown nonlinear function are derived based on the detailed noise, i.e. unknown but bounded noise and martingale difference noise. This process about constructing the unknown nonlinear function corresponds to the nonlinear function estimation on basis of input-output data.It means we apply the collected input-output data to identify the unknown nonlinear function. In section 4, the obtained estimation for the unknown nonlinear function is substituted into the cost function in MPC. The minimum principle for the numerical optimization is given to solve one optimal control input or controller. After defining the stability concept, dynamic programming is used to obtain the stability condition for our considered MPC in section 5. In section 6, one simulation example illustrates the effectiveness of the proposed theories. Section 7 ends the paper with a final conclusion and points our the next topic.

2 Nonlinear dynamic system

Due to linear system is widely studied in lots of references, so here in this short note we consider a more general case, i.e. consider the following first order discrete time nonlinear dynamical system.

$$y(t+1) = f(y(t)) + u(t) + w(t+1), t \geq 0 \tag{1}$$

where in equation (1), $y(t)$ and $u(t)$ are the system output and input respectively at time instant t. Nonlinear function $f()$ is completely unknown, the goal of next section is to estimate or identify this nonlinear function by input-output data sequence $\{u(t), y(t)\}$, external noise $w(t)$ is one unknown but bounded noise, which extends the spacial case of white noise, and its upper bound $w > 0$ satisfies.

$$\|w(t)\| \leq w, \forall t \geq 0 \tag{2}$$

To apply MPC to design the predictive control $u(t)$, the estimation of the nonlinear function $f(y(t))$ is expected to track one desired output reference $y_{des}(t)$. To measure its discrepancy, error value $(f(y(t)) - y_{des}(t))$ is needed to expand at time instant t. Due to nonlinear function $f(y(t))$ is unknown, so the urgent mission is to estimate this nonlinear function and use its estimation $\hat{y}(t)$ in error value, i.e. $(\hat{y}(t) - y_{des}(t))$.

In case of unknown but bounded noise $w(t)$, the nearest neighbor estimation for nonlinear function $f()$ is described to achieve the tracking.

Set

$$\begin{cases} \bar{b}(t) = \max_{0 \leq i \leq t} y(i) \\ \underline{b}(t) = \min_{0 \leq i \leq t} y(i) \end{cases} \tag{3}$$

and

$$i_t = \arg\min_{0 \leq i \leq t} \|y(t) - y(i)\| \tag{4}$$

Then we set

$$\|y(t) - y(i_t)\| = \min_{0 \leq i \leq t-1} \|y(t) - y(i)\| \tag{5}$$

So at each time instant $t \geq 1$, the nearest neighbor estimation for nonlinear function $f(y(t))$ is given as

$$\hat{f}(y(t)) = y(i_t + 1) - u(i_t) \tag{6}$$

Equation (6) can be rewritten as

$$\hat{f}(y(t)) = f(y(i_t+1)) + w(t+1) \quad (7)$$

To make the actual output $y(t)$ track the desired output reference $y_{des}(t)$, we define

$$\begin{cases} u^{'}(t) = -\hat{f}(y(t)) + \frac{1}{2}(\bar{b}(t) + \underline{b}(t)) \\ u^{''}(t) = -\hat{f}(y(t)) + y_{des}(t), t \geq 1 \end{cases} \quad (8)$$

Based on the definition (8), then the controller is defined as follows.

$$u(t) = \begin{cases} u^{'}(t) \;\; if \;\; \|y(t) - y(i)\| > \epsilon \\ u^{''}(t) \;\; if \;\; \|y(t) - y(i)\| \leq \epsilon \end{cases} \quad (9)$$

where $\epsilon > 0$ is one arbitrary small positive value.

Using the above controller $u(t)$(9), the tracking mission will reach, i.e. $\|y(t) - y(i)\| \to 0$ and observing above description, the estimation for nonlinear function is important in determining the controller. To return back to our considered MPC, this estimation for nonlinear function is also needed, so in section 3, we derive other estimation for nonlinear function based our own derivation.

3 Nonlinear function estimation

The nearest neighbor estimation (6) is efficient in case of bounded noise (2). To this end, as a consequence we assume that noise $w(t)$ is a martingale difference sequence, i.e.

$$E[w(t+1)B_t] = 0, t \geq 0 \quad (10)$$

where E mens the expectation, and $B_t = \sigma\{w(0), \cdots w(t)\}$ is σ-algebra, generated by sequence $\{w(0), \cdots w(t)\}$. Making use of the property of martingale difference sequence, set

$$\delta_j = [j\epsilon, (j+1)\epsilon], j \in Z = \{all \;\; integers\} \quad (11)$$

where Z is the set of all integers, $\epsilon > 0$ is one arbitrary small positive value, and it holds that.

$$\bigcup_{j \in Z} = (-\infty, +\infty) \quad (12)$$

and

$$\delta_i \bigcap \delta_j = \phi, for \;\; i \neq j \quad (13)$$

We define the following interval value function $\triangle()$ as, for each y

$$\triangle(y) = \delta_{j-1} \bigcup \delta_j \bigcup \delta_{j+1}, if \;\; y \in \delta_j, j \in Z \quad (14)$$

It means $\triangle(y)$ covers the ϵ neighborhood of y. For every $t \geq 1, \|y(t) - y(i)\| \leq \epsilon$, then it holds that.

$$\sum_{i=0}^{t-1} I_{\triangle(y(t))}(y(i)) > 0 \quad (15)$$

where $I_{\triangle(y(t))}$ is the indicator function. Therefore, if $\|y(t) - y(i)\| \leq \epsilon$, the estimation of that nonlinear function $f()$ is defined as.

$$\begin{aligned} \hat{y}(t) &= \hat{f}(y(t)) \\ &= \frac{\sum_{i=0}^{t-1}(y(i+1) - u(i)) I_{\triangle(y(t))}(y(i))}{\sum_{i=0}^{t-1} I_{\triangle(y(t))}(y(i))} \end{aligned} \quad (16)$$

To give a more detailed description on the above estimation, we list some special estimations for $t = 1, 2, 3 \cdots N$, where N is a finite time horizon,i.e.

$$\begin{aligned} \hat{y}(1) &= \frac{(y(1) - u(0)) I_{\triangle(y(1))}(y(0))}{I_{\triangle(y(1))}(y(0))} \\ &= y(1) - u(0) \end{aligned}$$

$$\begin{aligned} \hat{y}(2) &= \frac{(y(1) - u(0)) I_{\triangle(y(2))}(y(0))}{I_{\triangle(y(2))}(y(0)) + I_{\triangle(y(2))}(y(1))} \\ &+ \frac{(y(2) - u(1)) I_{\triangle(y(2))}(y(1))}{I_{\triangle(y(2))}(y(0)) + I_{\triangle(y(2))}(y(1))} \end{aligned}$$

$$\begin{aligned} \hat{y}(3) &= \frac{(y(1) - u(0)) I_{\triangle(y(3))}(y(0))}{I_{\triangle(y(3))}(y(0)) + I_{\triangle(y(3))}(y(1)) + I_{\triangle(y(3))}(y(2))} \\ &+ \frac{(y(2) - u(1)) I_{\triangle(y(3))}(y(1))}{I_{\triangle(y(3))}(y(0)) + I_{\triangle(y(3))}(y(1)) + I_{\triangle(y(3))}(y(2))} \\ &+ \frac{(y(3) - u(2)) I_{\triangle(y(3))}(y(2))}{I_{\triangle(y(3))}(y(0)) + I_{\triangle(y(3))}(y(1)) + I_{\triangle(y(3))}(y(2))} \end{aligned}$$

$$\vdots$$

$$\begin{aligned} \hat{y}(N) &= \frac{(y(1) - u(0)) I_{\triangle(y(N))}(y(0))}{I_{\triangle(y(N))}(y(0)) + \cdots + I_{\triangle(y(N))}(y(N-1))} \\ &+ \cdots \\ &+ \frac{(y(N) - u(N-1)) I_{\triangle(y(N))}(y(N-1))}{I_{\triangle(y(N))}(y(0)) + \cdots + I_{\triangle(y(N))}(y(N-1))} \end{aligned}$$

Based on our derived estimation of that nonlinear function, then $\hat{y}(t)$ can be regarded as the actual output at time instant t. Furthermore $\{u(0), y(0)\}$ is initial data sequence.

4 Model predictive control

As the goal of MPC is to make the considered system to track that desired output reference $y_{des}(t)$ and reject noise from $t = 0$ up to a finite time horizon N. MPC turns one optimal control problem into one numerical optimization problem, whose cost function is always set as that.

$$\min_{u(0)\cdots u(N)} \sum_{t=0}^{N} [\hat{y}(t) - y_{des}(t)]^T Q[\hat{y}(t) - y_{des}(t)] + u(t)^T S u(t) \tag{17}$$

where $\hat{y}(t)$ is the actual output, coming from equation (16), $y_{des}(t)$ is the desired output reference, Q and S are two positive definite weighting matrices.

4.1 Numerical optimization problem

In order to apply dynamic programming algorithm to solve the cost function in MPC (17), we need to simplify that cost function to satisfy the standard form for dynamic programming. The main step in MPC is to obtain those optimal control input or optimal controller $\{u(0), u(1), \cdots u(N)\}$, while minimizing the cost function.

Expanding the cost function (17) to that.

$$\sum_{t=0}^{N} \hat{y}^T(t) Q \hat{y}(t) - 2y_{des}(t) Q \hat{y}(t) + y_{des}^T(t) Q y_{des}(t) + u(t)^T S u(t) \tag{18}$$

Neglecting the third term $y_{des}^T(t) Q y_{des}(t)$, as it is independent of the control input $\{u(0), u(1), \cdots u(N)\}$, then the simplified cost function is that.

$$\sum_{t=0}^{N} \hat{y}^T(t) Q \hat{y}(t) - 2y_{des}(t) Q \hat{y}(t) + u(t)^T S u(t) \tag{19}$$

Rewriting the above simplified cost function as its standard from form, and denoting it as notation $J(y_0, u_0, u_1 \cdots u_{N-1})$,i.e.

$$\begin{aligned} &\sum_{t=0}^{N-1} \hat{y}^T(t) Q \hat{y}(t) - 2y_{des}(t) Q \hat{y}(t) + u(t)^T S u(t) \\ &\quad + \hat{y}^T(N) Q \hat{y}(N) \\ &= \sum_{t=0}^{N-1} g_t(\hat{y}(t), u(t)) + g_N(\hat{y}(N)) \\ g_t(\hat{y}^T(t), u(t)) &= \hat{y}^T(t) Q \hat{y}(t) - 2y_{des}(t) Q \hat{y}(t) + u(t)^T S u(t) \\ g_N(\hat{y}(N)) &= \hat{y}^T(N) Q \hat{y}(N) \end{aligned} \tag{20}$$

According to the standard form in dynamic programming algorithm, for a given initial output $y_0 = y(0)$, the total cost of a control sequence $\{u(0), u(1), \cdots u(N)\}$ is

$$J(y_0, u_0, u_1 \cdots u_{N-1}) = \sum_{t=0}^{N-1} g_t(\hat{y}(t), u(t)) + g_N(\hat{y}(N)) \tag{21}$$

where for the sake of brevity, we denote $u_0 = u(0), u_1 = u(1), \cdots, u_{N-1} = u(N-1)$. Equation (21) coincides with the cost function, and system equation is that.

$$\begin{aligned} \hat{y}(t) &= \frac{\sum_{i=0}^{t-1} (y_{i+1} - u_i) I_{\triangle(y_t)}(y_i))}{\sum_{i=0}^{t-1} I_{\triangle(y_t)}(y_i)} \\ &= f_{t-1}(y_{t-1}, y_{t-1}), t = 1, 2 \cdots N \end{aligned} \tag{22}$$

Combing equation (20),(21)and (22)m these three equations are studied in our introduced dynamical programming algorithm.

4.2 Dynamic Programming algorithm

Consider that numerical optimization problem again (21) for MPC, whose actual output $\hat{y}(t)$ is estimated from equation (22),i.e.

$$J(y_0, u_0, u_1 \cdots u_{N-1}) = \sum_{t=0}^{N-1} g_t(\hat{y}(t), u(t)) + g_N(\hat{y}(N))$$

The idea for principle of optimizality with the structure of dynamic programming. Let $\{u_0^* \cdots u_{N-1}^*\}$ be an optimal control sequence, consider the subproblem that starts at $\hat{y}^*(k)$ at time instant k, and want to minimize the cost-to-go from time instant k to time horizon N.

$$g_k(\hat{y}^*(k), u(k)) + \sum_{m=k+1}^{N-1} g_m(\hat{y}(m), u(m)) + g_N(\hat{y}(N)) \tag{23}$$

over $\{u_k, \cdots u_{N-1}\}, m = k \cdots N-1$. Then the truncated optimal sequence $\{u_k^* \cdots u_{N-1}^*\}$ is optimal for this subproblem. Dynamic programming algorithm construct the optimal cost functions.

$$J_N^*(y_N), J_{N-1}^*(y_{N-1}), \cdots J_0^*(y_0) \tag{24}$$

Sequentially starting from J_N^* and proceeding backwards to $J_{N-1}^*, J_{N-2}^* \cdots J_0^*$, i.e. start with $J_N^*(y_N) = g_N(\hat{y}(N))$ and for $k = 0, 1 \cdots N-1$, let

$$J_k^*(y_k) = \min_{u_k \cdots u_{N-1}} [g_k(\hat{y}(k))$$

$$+\sum_{m=k+1}^{N-1} g_m(\hat{y}(m),u(m))+g_N(\hat{y}(N))]$$
$$=\min_{u_k}[\sum_{m=k}^{N-1} g_m(\hat{y}(m),u(m))+g_N(\hat{y}(N))]$$
$$=\min_{u_k}[g_k(\hat{y}(k))$$
$$+\min_{u_{k+1}\cdots u_{N-1}}[\sum_{m=k+1}^{N-1} g_m(\hat{y}(m),u(m))+g_N(\hat{y}(N))]$$
$$=\min_{u_k}[g_k(\hat{y}(k))+J_{k+1}^*(\hat{y}(k+1))] \quad (25)$$

where in equation (25), the principle of optimizality is used.

Then for every initial output value y_0, the number $J_0^*(y_0)$, obtained at the last step, is equal to the optimal cost $J^*(y_0)$. More specifically, once the optimal cost function $J_0^*,\cdots J_N^*$ have been obtained, dynamic programming algorithm to construct an optimal control sequence $\{u_0^*\cdots u_{N-1}^*\}$ and corresponding output trajectories $\{\hat{y}_1^*\cdots\hat{y}_N^*\}$ for the given initial output value y_0.

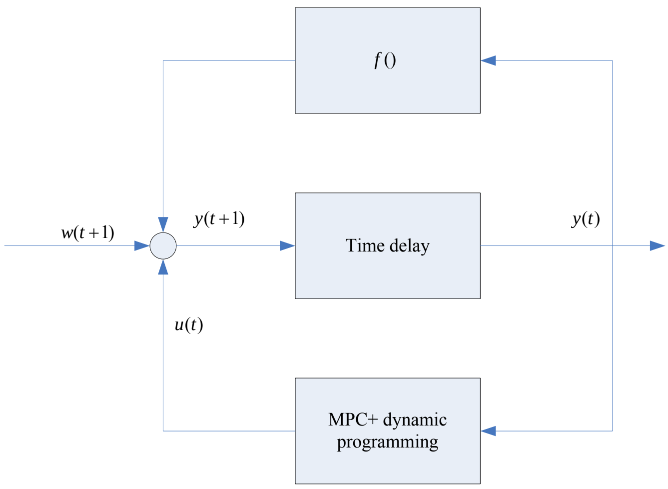

Figure.1 Structure of dynamic programming and data driven MPC

Now dynamic programming algorithm is formulated as follows. Set

$$u_0^*\in\arg\min_{u_0}[g_0(y_0,u_0)+J_1^*(\hat{y}_1(y_0,u_0))]$$

and

$$\hat{y}_0^*=y_1-u_0^*$$

Sequentially, going forward, for $k=1,2,\cdots N-1$, set

$$u_k^*\in\arg\min_{u_k}[g_k(\hat{y}_k^*,u_k)+J_{k+1}^*(\hat{y}_{k+1})]$$

and

$$\hat{y}_{k+1}=\frac{\sum_{i=0}^{k}(y(i+1)-u^*(i))I_{\triangle(y_k)}(y_i)}{\sum_{i=0}^{k}I_{\triangle(y_k)}(y_i)}$$

The above forward optimal control sequence construction is possible only after we have computed $J_k^*(\hat{y}_k)$ by dynamic programming algorithm for all $\hat{y}_k$ and k. Then the structure of combing dynamic programming algorithm and data driven MPC is plotted in Figure 1.

5 Stability analysis based on dynamic program-Ming

This section is one auxiliary part, showing that dynamic programming can also be applied to test the stability analysis for that MPC, then one explicit form about the cost perstage is needed. Without loss generality, the cost perstage is quadratic form, i.e.

$$\hat{y}_0^T(k)Q\hat{y}_0(k)+u_0^T(k)Su_0(k),k=0,1\cdots N \quad (26)$$

where Q and S are the same with equation (17), also we impose output and control constraints.

$$\hat{y}_0(k)\in Y_0(k)\ \ and\ \ u_0(k)\in U_0(k),k=0,1\cdots N \quad (27)$$

for all initial output value $y_0(0)\in Y_0$, the output value of the closed loop system is that.

$$\hat{y}_0(k+1)=\frac{\sum_{i=0}^{k-1}(y(i+1)-u(i))I_{\triangle(y_{k+1})}(y_i)}{\sum_{i=0}^{k-1}I_{\triangle(y_{k+1})}(y_i)} = f_{k-1}(y_{k-1},u_{k-1}) \quad (28)$$

Satisfied the output and control constraints, and $u_0^*(k)$ is one stationary controller. The stability concept is defined as that the total cost over an infinite number of state is finite, i.e.

$$\sum_{k=0}^{\infty}(\hat{y}^T(k)Q\hat{y}(k)+u^*(k)Su^*(k))<\infty \quad (29)$$

Let $y(0),u(0),\hat{y}(1),u(1)\cdots$ be the output and control sequence, generated by data driven MPC, so that.

$$u(k)=u^*(k)$$
$$\hat{y}(k+1)=f_{k-1}(y_{k-1},u_{k-1})$$
$$k=0,1\cdots \quad (30)$$

Denote $\hat{J}(\hat{y}(i))$ the optimal cost of the N stage problem solved by data driven MPC, and $\tilde{J}(\hat{y}(i))$ be the optimal cost starting at $y(0)$ of a corresponding $N-1$ stage problem. It means that the optimal value of the quadratic cost.

$$\sum_{k=0}^{N-2}(\hat{y}^T(k)Q\hat{y}(k)+u^*(k)Su^*(k))<\infty \quad (31)$$

and

$$\hat{y}(N-1)=0 \quad (32)$$

Since that deriving the state to 0 can not decrease the optimal cost, then we have for all $\hat{y}(k)\in Y(k)$

$$\hat{J}(\hat{y})\leq \tilde{J}(\hat{y}) \quad (33)$$

From the definitions of $\hat{J}(\hat{y})$ and $\tilde{J}(\hat{y})$, we have for all k,

$$\begin{aligned}\min_{u\in U}&[\hat{y}^T(k)Q\hat{y}(k)+u_i^T(k)Su_i(k)+\tilde{J}(\hat{y}(k+1))]\\ &=\hat{y}^T(k)Q\hat{y}(k)+u_i^T(k)Ru_i(k)\\ &+\tilde{J}(\hat{y}(k+1))\\ &=\hat{J}(\hat{y}(k))\end{aligned} \quad (34)$$

by using equation (33), we have that.

$$\begin{aligned}&\hat{y}^T(k)Q\hat{y}(k)+u_i^T(k)Su_i(k)+\hat{J}(\hat{y}(k+1))\\ &\leq \hat{J}(\hat{y}(k))\end{aligned} \quad (35)$$

Summing these equations for arbitrary values $N>0$, it holds that.

$$\begin{aligned}&\hat{J}(\hat{y}(k+1))+\sum_{k=0}^{N}(\hat{y}^T(k)Q\hat{y}(k)+u^T(k)Su(k))\\ &\leq \hat{J}(y(0))\end{aligned} \quad (36)$$

Since $\hat{J}(\hat{y}(k+1))\geq 0$, it holds that.

$$\sum_{k=0}^{N}(\hat{y}^T(k)Q\hat{y}(k)+u^T(k)Su(k))\leq \hat{J}(y(0)) \quad (37)$$

Taking the limit on above inequality as $N\to\infty$, we have that.

$$\sum_{k=0}^{\infty}(\hat{y}^T(k)Q\hat{y}(k)+u^T(k)Su(k))\leq \hat{J}(y(0))<\infty \quad (38)$$

This completes the proof of that stability concept (29), whose total cost over an infinite number of state is finite.

Combing section 4 and 5, we find that dynamic programming is used not only to solve the numerical optimization problem, but also to test the stability of the closed loop system in Figure 1.

6 Conclusion

In this short note, data driven model predictive control is considered to identify the actual output value and design the optimal control sequence. The combination of system identification and dynamic programming is used to complete model predictive control. More specifically, optimal control sequence is constructed by dynamic programming algorithm, and stability is analyzed by dynamic programming for the whole obtained closed loop system. As this is the first analysis, regarding dynamic programming and model predictive control, so that how to combine game theory and dynamic programming for data driven model predictive control is our next ongoing work.

Acknowledgements

This work is partially supported by the Grants from the Foundation of Jiangxi University of Science and Technology (No. jxxjb18020). The authors are grateful to Professor Eduardo F Camacho for his warm invitation in his control lab at the University of Seville, Seville, Spain.

Conict of interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

References

[1] Zhang Xiaojing, Sergio Grammatico, Gerorg Schildbach (2015). On the sample size of random convex programs with structured dependences on the uncertainty. *Automatica 60(10)*, 182–188.

[2] Zhang Xiaojing, Maryam Kamgarpour, Angeios Georghial (2017). Robust optimal control with adjustable uncertainty sets. *Automatica 75(1)*, 249–259.

[3] T Alamo, R Tempo, E F Camacho (2009). Randomized strategy for probabilistic solution of uncertain feasiblity and optimization problems. *IEEE Transactions on Automatic Control 54(11)*, 2545–2559.

[4] D P Bertsekas (2019). Affine monotonic and risk sensitive models in dynamic programming. *IEEE Transactions on Automatic Control 64(8)*, 3117–3128.

[5] D P Bertsekas, V Goyal (2012). On the power and limitations of affine policies in two stage adaptive optimization. *Mathematical Programming 134(2)*, 491–531.

[6] L Blackmore, M Ono, B C Williams (2011).Chance constrained optimal path planning with obstacles. *IEEE Transactions on Robotics 27(6)*, 1080–1094.

[7] G C Calafiore (2010). Random convex programs. *SIAM Journal on Optimizaitons 20(6)*, 3427-3464.

[8] M C Campi, S Garatti (2016). Wait and judge scenario optimization. *Automatica 50(12)*, 3019–3029.

[9] M C Campi, S Garatti, M Prandini (2009).The scenario approach for systems and control design. *Annual Reviews in Control 33(2)*, 149-157.

[10] D Callawy, I Hiskens (2011). Achieving controllability of electric loads. *Proceedings of the IEEE 99(1)*, 184–199.

[11] M Farina, L Giulioni(2016). Stochastic linear model predictive control with chance constraints- a review. *Journal of Process Control 44(2)*, 53–67.

[12] S Garatti,M C Campi (2013). Modulating robustness in control design: principles and algorithm. *IEEE Control Systems magazine 33(2)*, 36–51.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0
https://creativecommons.org/licenses/by/4.0/deed.en_US