Dynamic Matrix Control of 1 DOF Rotor System

Jae-Heon Park, Yeong-Sang Park, Young-Sam Lee, and Oh-Kyu Kwon

Abstract—**This paper presents DMC algorithm in order to apply to systems with relatively fast process. Firstly, C MEX S-function was used to create the QP solver based on C-code with optimal control algorithm. Secondly, a real-time application is considered using library blocks from Matlab/Simulink. Finally, DMC simulator and PI controller were used for the velocity control of 1 DOF rotor. Through this paper, the implementation of a real-time application was considered for using DMC algorithm by creating C MEX S-function to generate a single DMC block in Matlab/Simulink. Additionally, the proposing theory applied to the SISO system (1 DOF rotor) as a previous experiment for the possibility of controlling the TRMS (Twin Rotor MIMO System).**

Keywords— **Dynamic Matrix Control, 1 Degree of Freedom, DC motor control, Nonlinear system, QP solver, C MEX S-function, Matlab/Simulink.**

I. Introduction

The Model predictive control has been widely used in many industrial processes like refining, chemicals, and power systems. Depending on the variety of models, the model predictive control is classified under control methods based on the state space model(MPC) [1], the transfer function model(GPC) [2 3], the unit step response of the system(DMC) [4] and so on.

Unlike the GPC and MPC, which are based on the mathematical model, DMC is designed to predict the future output using the step response of the system during the prediction horizon. The predicted output calculates the optimal control input through the optimization based on minimized cost function. The strengths of DMC is suitable to use in systems with stable or integrating process.

The differentiation from other paper is a real-time application rather than on theoretical developments. The C Mex S-function was used to generate the QP solver under the c-code based on optimal control algorithm. More and over, comparative experiment between PI control and DMC control was considered under 5ms sampling time in 1 DOF rotor system. Through the simulation, DMC, which was merely used for relatively slower system before, has been found that it could be also apply to the faster system. The proposed experiment uses the TMS320F2812 board with the high speed USB interface and a controller model designed from Matlab/Simulink acts as a real-time controller directly.

Jae-Heon Park, Yeong-Sang Park
School of Electrical Engineering / Inha University
Republic of Korea

Young-Sam Lee, Oh-Kyu Kwon
School of Electrical Engineering / Inha University
Republic of Korea

II. Quadratic Programming

Recent Industry, several papers have been discussed with the problems of DMC or applications and tried to verify the effectiveness through the simulation. However, the research related to the implementation of DMC based on c-code with the actual micro-controller is not relatively received much attention. One of the reason is that it is hard to load DMC algorithm in the micro-controller and it takes long time to solve the QP problem[5]. Due to the fact, it is hard to apply in the fast system with a real-time application. Consequently, to solve the problems, the C MEX S-function was used to create the QP Solver which is based on C-code with optimal control algorithm.

A. QP solver based on the C-code using C MEX S-function

In the predictive control, for every step, DMC calculates the control value that minimizes a given cost function with the satisfaction of the constraints as follows:

$$\text{Minimize } J = \sum_{j=1}^{N_p}\left\|w_{k+j} - \hat{y}_{k+j}\right\|^2 + \sum_{j=1}^{N_p}\left\|\Delta u_{k+j}\right\|_\Lambda^2 \tag{1}$$

$$\text{Subject to } u_{min} \le u_{k+j} \le u_{max}, \ \Delta u_{min} \le \Delta u_{k+j} \le \Delta u_{max} \tag{2}$$

Where w is a reference input, $\hat{y}$ is a predicted output, Δu is a control increment, N_p is a prediction horizon, and N_c is a control horizon. Typically, $N_p \ge N_c$ and Λ is a diagonal matrix that indicates a weighting factor for the control increment. DMC calculates the control increment for every step till $k + N_c - 1$, but only use Δu for the actual system as a control input, which is relative with

$$u_k = u_{k-1} + \Delta u_k$$

Define the reference input vector of the future as

$$W_k = \begin{bmatrix} w_{k+1}^T & w_{k+2}^T & \cdots & w_{k+N_p}^T \end{bmatrix}^T$$

and minimize the cost function. It becomes ΔU_k function as below.

$$J(\Delta U_k) = \left(W_k - \hat{Y}_k\right)^T\left(W_k - \hat{Y}_k\right) + \Delta U_k^T \Xi \Delta U_k$$

Where

$$\Xi = \text{diag}[\Lambda, \Lambda, \cdots \Lambda]$$

To get the secondary form of ΔU_k function, it can be written as

$$J(\Delta U_k) = \Delta U_k^T[G^T G + \Xi]\Delta U_k - 2(W_k - f_k)^T G\Delta U_k$$
$$+(W_k - f_k)^T(W_k - f_k) \tag{3}$$

International Journal of Advances in Electronics Engineering– IJAEE
Volume 6 : Issue 1   *[ISSN : 2278-215X]*

Publication Date: 18 April, 2016

DMC calculates the predicted output from present to N_p step. Therefore, the equation is

$$\hat{Y}_k = G\Delta U_k f_k. \qquad (4)$$

Where

$$\hat{Y}_k = \begin{bmatrix} \hat{y}_{k+1}^T & \hat{y}_{k+2}^T & \cdots & \hat{y}_{k+N_p}^T \end{bmatrix}^T$$

$$\Delta U_k = \begin{bmatrix} \Delta u_k^T & \Delta u_{k+1}^T & \cdots & \Delta u_{k+N_c-1}^T \end{bmatrix}^T$$

G is a predicted matrix, which consists of unit step response from the system and f_k is a free response for calculating the predicted output using the past data.

$$G = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_c} & g_{N_c-1} & \cdots & g_1 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_p} & g_{N_p-1} & \cdots & g_{N_p-N_c+1} \end{bmatrix}, \qquad (5)$$

$$f_k = \begin{bmatrix} \bar{y}_k \\ \bar{y}_k \\ \vdots \\ \bar{y}_k \end{bmatrix} + \begin{bmatrix} g_2 - g_1 & g_3 - g_2 & \cdots & g_N - g_{N-1} \\ g_3 - g_1 & g_4 - g_2 & \cdots & g_{N+1} - g_{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_p+1} - g_1 & g_{N_p+2} - g_2 & \cdots & g_{N_p+N-1} - g_{N-1} \end{bmatrix}$$

$$\begin{bmatrix} \Delta u_{k-1} \\ \Delta u_{k-2} \\ \vdots \\ \Delta u_{k-(N-1)} \end{bmatrix} + \begin{bmatrix} h \\ 2h \\ \vdots \\ N_p h \end{bmatrix} u_{k-N} \qquad (6)$$

Where $\bar{y}_k$ is a measurement output, ΔU_k is a $N_c \times 1$ input adjustments vector and N_c is a control horizon. $\hat{Y}_k$ is a $N_p \times 1$ vector representing a prediction of future output trajectory when $k = i$, and N_p is a prediction horizon. Therefore, G becomes $N_p \times N_c$ dynamic matrix containing a step response coefficients as the (5) [6 7].

Consequently, the value of ΔU_k^* while satisfying the constraints condition, as shown in the (2) and minimize a cost function, as shown in the (3) is

$$P = G^T G + \Xi, \quad q = -2(W_k - f_k)^T G, \qquad (4)$$

$$A = \begin{bmatrix} I_d \\ -I_d \\ I_L \\ -I_L \end{bmatrix}, \quad b = \begin{bmatrix} I_V u_{max} \\ -I_v u_{min} \\ I_V(u_{max} - u_{k-1}) \\ -I_V(u_{min} - u_{k-1}) \end{bmatrix} \qquad (5)$$

Where

$$I_D = \begin{vmatrix} I_m & 0 & \cdots & 0 \\ 0 & I_m & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & I_m \end{vmatrix}, \quad I_L = \begin{vmatrix} I_m & 0 & \cdots & 0 \\ I_m & I_m & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I_m & I_m & \cdots & I_m \end{vmatrix}, \quad I_V = \begin{vmatrix} I_m \\ I_m \\ \vdots \\ I_m \end{vmatrix}$$

The solution of QP problem is not only solving the NNLS(Non-negative Least Squares). Furthermore, complex calculations are required, such as SVD(Singular Value Decomposition). However, in order to implement DMC to the linear time-invariant system, we have to solve QP problems for every step. P and A are always be in constant value. On the contrary, q and b have the characteristic of changing the value at every step. Therefore, using these characters, it is possible to fulfil the algorithm which can significantly increase the effective calculation.
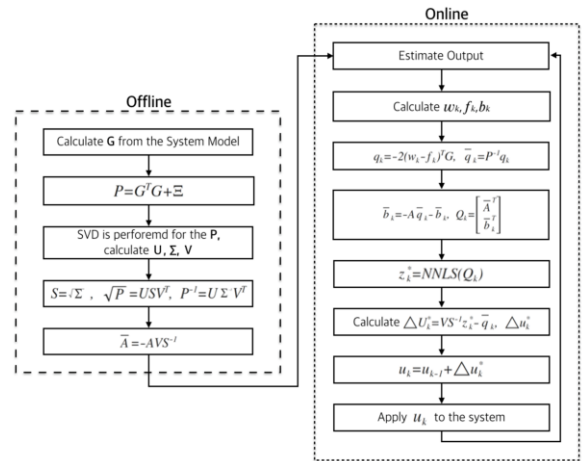

Figure 1.   The Proposed algorithm for the predictive control

The value P is fixed, therefore the SVD of P and the calculation of $\sqrt{P}$ and P^{-1} can be calculated in offline. DMC algorithms contains general arithmetic, multiplication of matrix, multiplication between matrix and vector, subtraction and addition of vector and NNLS. Those calculations must be calculated in online. Fig. 1 indicates the proposed flow chart of the model predictive control algorithm implementation. Assume that the offline calculation parts must be calculated via the micro-controller, then the SVD algorithm based on the c-code must be needed. Readers can find the SVD of a matrix based on c-code algorithm is on [8], and the solution algorithm of NNLS is on [9]. Also the c-code of NNLS is on internet [10].

Fig. 2 shows the flow chart for the implementation of the predictive control. The offline calculation parts should be working on the PC and the online calculation parts should be working on the micro-controller.
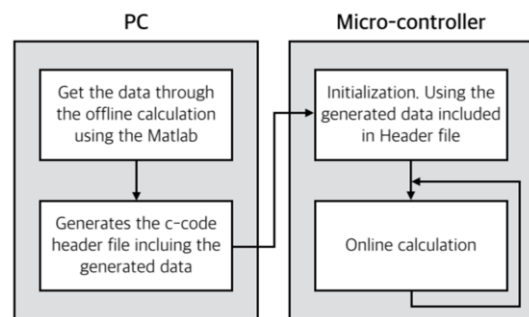

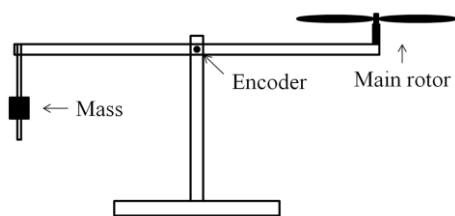Figure 2.   The flow chart for the implementation of the predictive control

Figure 3.   The concept diagram of the 1 DOF rotor system

III. 1 DOF rotor system

The 1 DOF rotor system is a simplified helicopter model, shown in Fig. 3, which is a concept diagram for the 1 DOF rotor system. It is highly non-linear system with a complex flight dynamics. The main DC motor produces a lifting force allowing the beam to rise vertically making a rotation around the pitch axis.

A. *Implementation of DMC block*

Through the proposed method, DMC algorithm can be easily fulfilled in Matlab/Simulink. Matlab/Simulink provides the C MEX S-function for easy access from block diagram. It is possible to define the initialization function block, the input/output relation functions and the termination function. Fig. 4 describes a concept diagram of DMC simulation. The input data of DMC block is the reference with the output data calculated from DMC block as controlled output. The process is working as feedback control. The input/output equations are implemented based on the proposed algorithm.
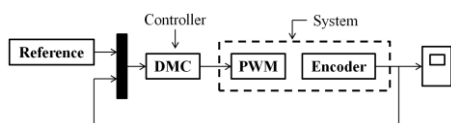

Figure 4.   The concept diagram of DMC simulation

IV. Simulation

This section indicates the possibility of using proposed DMC algorithm applied to the 1 DOF rotor system, and also it shows that it can be implemented in the faster system. The Maxon DC motor with 20 Watts was used and 1000 PPR encoder was used for control the pitch angle. Fig. 5 shows the laboratory platform designed for the TRMS simulation. The tail rotor is for the stability when the machine is not in progress, like mass shown in Fig. 3.
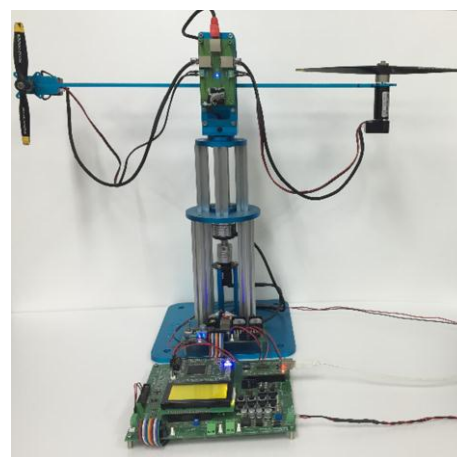

Figure 5.   The lab-built TRMS used for the simulation

A. *The unit step response data*

It is necessary to obtain the unit step response before running DMC simulator. First, supply 5 Volts step input to the main rotor to get the pitch data. After that, divide the pitch data by 5 for the normalization. Fig. 6 is a Simulink block for the unit step response experiment. The 1 DOF system block is a subsystem and the inside of the block is shown in Fig. 7.
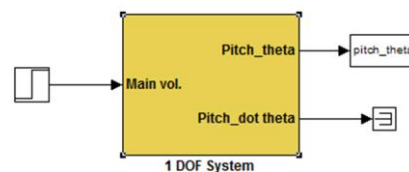

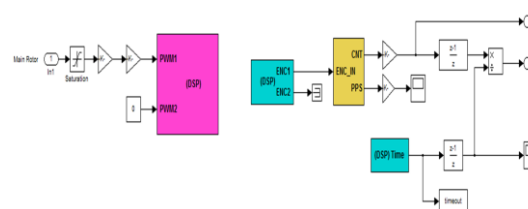Figure 6.   The simulation model for unit step response

Figure 7.   1 DOF subsystem

The sampling time is 5ms. The incremental encoder with the resolution of 1000 pulses/rev was used. Therefore quantization error occurs. Fig. 8 shows the single experiment data. The result is not good enough to use in the simulator. Therefore, to improve the data, the unit step response experiment was done for 30 times and calculate the average data. After that, Matlab command 'filtfilt' was used for smoothing the data. Fig. 9 indicates smoother unit step response data than Fig. 8.
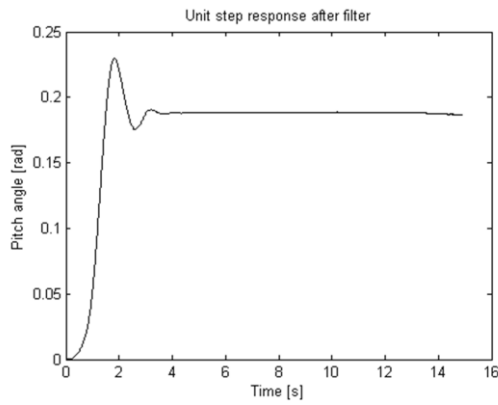
Figure 9.   Unit step response after filter

B. *The position control of DMC*

From the (6), considering the parameter N, the value should be at the stable boundary. Shown in Fig. 9, the pitch angle is approached on stable situation in about 5 seconds after the voltage supplied. Therefore, $t = 6$ and that makes $N = 6/0.005 = 1200$. Set up the control horizon and the prediction horizon as $N_c = 5$ and $N_p = 42$. Fig. 10 is proposed DMC simulator for controlling the 1 DOF system. The step input(LPF) was used as a reference for protect the plant. Fig. 11 shows the performance of DMC and generally the real data looks smooth at stable boundary. After supplying the voltage, in about 6 seconds later, the system approached at the stable boundary. In about 20 seconds later, a disturbance occurred, but the system gradually approached the stable boundary and it takes about 10 seconds.
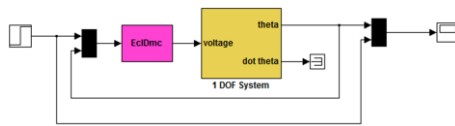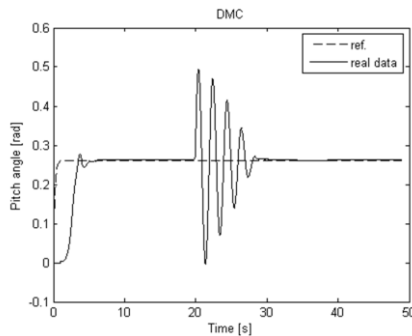

Figure 10.   1 DOF system using DMC

Figure 11.   Tracking performance using DMC

C. *The position control of PI*

PI(Proportional-Integral) control is commonly used in industrial control systems for a long time. PI control can provide control action designed for specific process requirements. However, PI controller has its weakness, in terms of considering the gain tuning and constraints. It cannot display a satisfactory performance because of its monotonous control mechanism and it could be more confused when the gains and constraints are considered in

MIMO system [11]. On the other hands, DMC has the advantage of being able to easily control in consideration of the constraints. This might be the reason why MPC is concerned in industrial processes.

This section is for comparing between DMC and PI control. Fig. 12 shows PI controller for 1 DOF system using Matlab/Simulink. Additionally, the anti-windup compensation was considered for preventing the degradation of performance [12]. For a fair comparison, PI gains are appropriately considered.

Fig. 13 is the result of the tracking performance using PI control. The reference input was considered for same reason as DMC simulation. After supplying the voltage, the real data approached at the stable boundary of the reference in about 5 seconds. As shown in Fig. 13, the real data looks little bit rough at stable boundary. In about 20 seconds later, same disturbance occurred. As a result, PI controller takes about 15 seconds for approaching at the stable boundary and more unstable than DMC. This compare experiment shows DMC is more stable and better response for a disturbance.
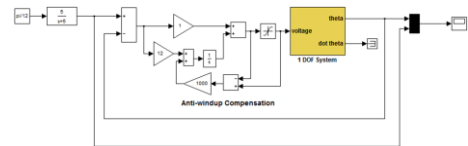

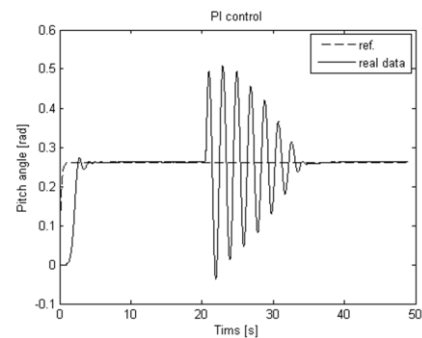Figure 12.   PI controller for 1 DOF system

Figure 13.   Tracking performance using PI control

V.   **Conclusion**

This paper presents the implementation of 1 DOF system using proposed DMC simulator based on Matlab/Simulink. QP (Quadratic Programming) is one of the most important topics in DMC. Therefore, QP solver and DMC algorithm were created based on c-code with optimal control algorithm and C MEX S-function was used to create the DMC block based on the created c-code for applying to Matlab/Simulink. The proposed method used to create a real-time DMC application in a single block. This makes the simulation possible to run the real-time controller of 1 DOF system with a lab-built RCP system. Through the experiment, the proposed DMC algorithm can apply to the fast dynamics from the results between PI control and DMC. The purpose of this paper is to fulfil the real-time application using DMC theory by creating C MEX S-function to generate a single DMC block for DMC controller in Matlab/Simulink. Additionally, this experiment shows the possibility of the DMC that can be applied to TRMS model (MIMO system).

Acknowledgment

References

[1]   J. H. Lee, M. Morari, and C. E. Garcia, "State-space interpretation of model predictive control," Automatica, vol. 30, no. 10, pp. 707-717, 1994.

[2]   D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control-Part I. The basic algorithm," Automatic, vol. 23, no. 2, pp. 137-148, 1987.

[3]   D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control-Part II. Extensions and interpretations," Automatic, Vol 23, no. 2, pp. 149-160, 1987.

[4]   C. E. Garcia and A. M. Morshedi, "Quadratic programming solution of dynamic matrix control(QDMC)," Chem. Eng. Commun., vol. 46, pp. 73-87, 1986.

[5]   Y. S. Lee, G. Y. Gyeong, and J. H. Park, "QP Solution for the implementation of the predictive control on micocontroller systems and its application method," Journal of Institute of Control, Robotics, and Systems (in Korean), vol. 20, no. 9, pp. 908-913, 2014.

[6]   Un-Chul Moon, Seung-Chul Lee, and Kwang Y. Lee, "An Adaptive Dynamic Matrix Control of a Boiler-Turbine System Using Fuzzy Inference," International Conference on Intelligent Systems Application to Power System, pp. 1-6, 2007

[7]   P. Lundstrom, J. H. Lee, M. Morari, and S. Skogestad, "Limitations of dynamic matrix control," Computers Chemical Engineering, vol. 19, no. 4, pp. 409–421, 1995

[8]   W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Plannery, "Numerical recipies in C," Cambridge University Press, Cambridge, 1988.

[9]   C. L. Lawson and R. J. Hanson, "Solving least squares problems, Prentice-Hall," Englewood Cliffs, New Jersey, 1974.

[10] http://hesperia.gsfc.nasa.gov/~schmahl/nnls/nnls.c.

[11] P. J. Gawthrop and P. E. Nomikos, "Automatic tuning of commercial PID controllers for singleloop and multiloop application," IEEE Control System Magazine, vol. 10, no. 1, pp. 34-42, 1990.

[12] K. J. Åström and L. Rundqwist, "Integrator windup and how to avoid it," in Proc. of the 1989 American Control Conference, pp. 1693–1698, 1989.

About Author (s):

Jae-Heon Park received his B.S. degree in Electrical Engineering from Inha University, Incheon, Korea, in 2014, respectively. He is studying M.S. degree in Electrical Engineering from Inha University. Interests in MPC

Yeong-Sang Park received his B.S. degree in Electronic Engineering from Inha University, Incheon, Korea, in 2015, respectively. He is studying M.S. degree in Electrical Engineering from Inha University. Interests in Control, Machine learning

Young Sam Lee received his B.S. and M.S. degrees in Electrical Engineering from Inha University, Inchon, Korea, in 1997 and 1999 and Ph.D. at the School of Electrical Engineering &Computer Science from Seoul National University, Seoul, Korea, in 2003, respectively. His research interests time delay systems, embedded systems. He is currently with the School of Electrical Engineering, Inha University, Incheon, Korea.

Oh-Kyu Kwon received his B.S., M.S., and Ph.D. degrees in Electrical Engineering from Seoul National University, Seoul, Korea, in 1978, 1980 and 1985, respectively. His research interests include receding horizon control, robust control, and signal processing. He is currently with the School of Electrical Engineering, Inha University, Incheon, Korea.