# TOEPLITZ/HANKEL MATRIX ALGEBRA IN MODEL PREDICTIVE CONTROL

# TOEPLITZ/HANKEL MATICOVÁ ALGEBRA V PREDIKTIVNÍM ŘÍZENÍ

Daniel Honc, Libor Havlíček[1]

*Summary: Model-based Predictive Control (MPC) algorithms popularity grows steadily in industry and academic area as well. Control actions are computed as an optimization problem. Future controlled system behaviour is taken into account - predicted over some horizon. Different methods and mathematical apparatus are utilized to get prediction equations. Diophantine equations and their recursive solving are very neat methods. On the other hand to compute predictions by matrix operations in state-space is easily to program. Paper is focussed on another insightful and compact method based on input-output process model representation and taking advantage of Toeplitz/Hankel matrix algebra.*

*Key words: model predictive control, CARIMA process model, Toeplitz/Hankel matrix algebra*

*Anotace: Popularita prediktivních algoritmů v průmyslu i akademické oblasti neustále roste. Akční zásahy jsou počítány jako optimalizační problém. Je bráno do úvahy budoucí chování řízeného procesu – je predikováno na konečném horizontu. Pro získání predikčních rovnic jsou používány různé metody a matematické aparáty. Rekurzivní řešení Diofantických rovnic jsou velmi elegantní metody. Na druhé straně výpočet predikce pomocí maticových operací ve stavovém popisu je jednoduše programovatelný. Článek je zaměřen na jinou průhlednou a kompaktní metodu založenou na vstupně-výstupní reprezentaci modelu a využívající výhody Toeplitz/Hankel maticové algebry.*

*Klíčová slova: prediktivní řízení, CARIMA model, Toeplitz/Hankel maticová algebra*

## 1. INTRODUCTION

Generalized Predictive Control (GPC) as it was published by Clarke and Mohtadi [1], [2] is based on CARIMA process model. Polynomial algebra is used to get prediction equations – it is necessary to solve Diophantine equations. This is very elegant and scientifically interesting problem but for beginners usually not very clear and intuitive. I is not easy to program general controller design algorithm because of the problem complexity for multi-input multi-output (MIMO) processes, measurable disturbances and coloured noise case. Methods based on state-space models are more easily to understand and to program [3]. Their main disadvantage is necessity to measure or estimate state variables and problems with steady-state control error. Alternative method how to compute prediction equation from input-

[1] Ing. Daniel Honc, Ph.D., Ing. Libor Havlíček, Univerzita Pardubice, Fakulta elektrotechniky a informatiky, Katedra řízení procesů, nám. Čs. legií 565, 532 10 Pardubice, tel.: +420 466 037 107, fax +420 466 037 068, e-mail: daniel.honc@upce.cz

output CARIMA process model is presented in the paper. This method is easy to understand and only knowledge of difference equations and matrix algebra is required [4].

## 2. MODEL PREDICTIVE CONTROL CONCEPT

If disturbances are non-stationary CARIMA process model is usually considered (resulting controller has integrating character)

$$\Delta A\left(z^{-1}\right)y(k)=B\left(z^{-1}\right)\Delta u(k-1)+C\left(z^{-1}\right)e(k) \tag{1}$$

where $A(z^{-1})$, $B(z^{-1})$ and $C(z^{-1})$ are polynomials, increment operator $\Delta$ is defined as $\Delta = 1 - z^{-1}$, $y(k)$ is plant output, $u(k)$ is plant input and $e(k)$ is zero-mean white noise.

Following finite horizon quadratic criterion (cost function) is minimized

$$J = r\sum_{j=N_1}^{N_2}\left[\hat{y}(k+j\,|\,k)-w(k+j)\right]^2 + q\sum_{j=1}^{N_3}\Delta u(k+j-1)^2 \tag{2}$$

where $\hat{y}(k+j\,|\,k)$ is an optimum j-step ahead prediction of the system output, $N_1$ and $N_2$ are the minimum and maximum prediction horizon and $w(k+j)$ is a future set-point. $N_3$ is the control horizon (after the first $N_3$ control moves is the control signal kept constant). $r$ and $q$ are weighting coefficients.

Plant output prediction can be expressed in matrix form as

$$\mathbf{Y}_{N12} = \mathbf{G}_{N123}\cdot\Delta\mathbf{U}_{N3} + \mathbf{G}'_{N12}\cdot\Delta\mathbf{U}_{\mathrm{p}} + \mathbf{F}_{N12}\cdot\mathbf{Y}_{\mathrm{p}} + \mathbf{E}_{N12}\cdot\mathbf{E}_{\mathrm{p}} = \mathbf{G}_{N123}\cdot\Delta\mathbf{U}_{N3} + \mathbf{f}_{N12} \tag{3}$$

where

$$\mathbf{Y}_{N12} = \left[\hat{y}(k+N_1\,|\,k)\ldots\hat{y}(k+N_2\,|\,k)\right]^T,\ \Delta\mathbf{U}_{N3} = \left[\Delta u(k)\ldots\Delta u(k+N_3-1)\right]^T,$$
$$\Delta\mathbf{U}_{\mathrm{p}} = \left[\Delta u(k-1)\ldots\Delta u(k-n_b)\right]^T,\ \mathbf{Y}_{\mathrm{p}} = \left[y(k)\ldots y(k-n_a)\right]^T,\ \mathbf{E}_{\mathrm{p}} = \left[e(k)\ldots e(k-n_c+1)\right]^T$$

and matrices $\mathbf{G}_{N123}, \mathbf{G}'_{N12}, \mathbf{F}_{N12}$ and $\mathbf{E}_{N12}$ are calculated from the process model.

Cost function can be rewritten in matrix form and the optimum future control moves can be expressed analytically for unconstrained case. Receding strategy means that only the first element of the sequence $\Delta\mathbf{U}_{N3}$ is actually sent to the process and the control action is then

$$\Delta u(k) = \mathbf{K}\left(\mathbf{W}_{N12} - \mathbf{f}_{N12}\right) \tag{4}$$

where $\mathbf{K}$ is controller gain matrix and $\mathbf{W}_{N12}$ are future set-points $\mathbf{W}_{N12} = \left[w(k+N_1)\ldots w(k+N_2)\right]^T$.

## 3. PREDICTION EQUATIONS

CARIMA process model has a form (we consider white noise case for the sake of simplicity)

$$\widetilde{A}\left(z^{-1}\right)\cdot y(k)=B\left(z^{-1}\right)\cdot \Delta u(k-1),\quad \widetilde{A}=\Delta A,\quad C\left(z^{-1}\right)=1 \tag{5}$$

We can rewrite process model equations (5) for N future sampling times as

$$\begin{aligned}
&y(k+1)+\widetilde{a}_1 y(k)+\widetilde{a}_2 y(k-1)+\ldots+\widetilde{a}_{n_a+1} y(k-n_a)=b_0\Delta u(k)+b_1\Delta u(k-1)+\ldots+b_{n_b}\Delta u(k-n_b)\\
&y(k+2)+\widetilde{a}_1 y(k+1)+\widetilde{a}_2 y(k)+\ldots+\widetilde{a}_{n_a+1} y(k-n_a+1)=b_0\Delta u(k+1)+b_1\Delta u(k)+\ldots+b_{n_b}\Delta u(k-n_b+1)\\
&\vdots\\
&y(k+N)+\widetilde{a}_1 y(k+N-1)+\widetilde{a}_2 y(k+N-2)+\ldots+\widetilde{a}_{n_a+1} y(k-n_a+N-1)=b_0\Delta u(k+N-1)+\\
&\quad +b_1\Delta u(k+N-2)+\ldots+b_{n_b}\Delta u(k-n_b+N-1)
\end{aligned} \tag{6}$$

Set of equations (6) written in matrix form is

$$\begin{aligned}
&\begin{bmatrix} 1 & 0 & & & \cdots & 0\\ \widetilde{a}_1 & 1 & 0 & & \cdots & 0\\ & & & \vdots & & \\ 0 & \cdots & 0 & \widetilde{a}_{n_a+1} & \cdots\ \widetilde{a}_1 & 1 \end{bmatrix}\cdot\begin{bmatrix} y(k+1)\\ y(k+2)\\ \vdots\\ y(k+N)\end{bmatrix}=\begin{bmatrix} b_0 & 0 & & & \cdots & 0\\ b_1 & b_0 & 0 & & \cdots & 0\\ & & & \vdots & & \\ 0 & \cdots & 0 & b_{n_b} & \cdots\ b_1 & b_0\end{bmatrix}\cdot\begin{bmatrix}\Delta u(k)\\ \Delta u(k+1)\\ \vdots\\ \Delta u(k+N-1)\end{bmatrix}+\\
&+\begin{bmatrix} b_1 & b_2 & \cdots & & b_{nb}\\ b_2 & b_3 & \cdots & b_{n_b} & 0\\ & & \vdots & & \\ 0 & & \cdots & & 0\end{bmatrix}\cdot\begin{bmatrix}\Delta u(k-1)\\ \Delta u(k-2)\\ \vdots\\ \Delta u(k-n_b)\end{bmatrix}+\begin{bmatrix} -\widetilde{a}_1 & -\widetilde{a}_2 & \cdots & & -\widetilde{a}_{n_a+1}\\ -\widetilde{a}_2 & -\widetilde{a}_3 & \cdots & -\widetilde{a}_{n_a+1} & 0\\ & & \vdots & & \\ 0 & & \cdots & & 0\end{bmatrix}\cdot\begin{bmatrix} y(k)\\ y(k-1)\\ \vdots\\ y(k-n_a)\end{bmatrix}
\end{aligned} \tag{7}$$

To get vector of predicted process outputs $\mathbf{Y}_N=\left[y(k+1)\ldots y(k+N)\right]^T$ we have to multiply equation (7) by inverse of matrix on the left side or we can use Toeplitz/Hankel matrix algebra and compute prediction equations iteratively (as analogy to Diophantine equations solving).

## 4. EXAMPLE

Let us consider continuous-time process model $G(s)=\dfrac{1}{(s+1)^2}$ and GPC controller parameters - see Table 1.

Table 1 GPC parameters

| $T_S$ | $N_1$ | $N_2 = N_3$ | $r$ | $q$ | $C(z^{-1}) = T(z^{-1})$ |
|---|---|---|---|---|---|
| 1 s | 1 | 5 | 1 | 0.05 | $(1\text{-}0.8\,z^{-1})^2$ |

After discretization we get discrete-time transfer function and difference equation

$$G\left(z^{-1}\right)=\frac{0.2642z^{-1}+0.1353z^{-2}}{1-0.7358z^{-1}+0.1353z^{-2}} \quad \rightarrow$$

$$y(k)-0.7358y(k-1)+0.1353y(k-2)=0.2642u(k-1)+0.1353u(k-2) \tag{8}$$

$$A\left(z^{-1}\right)=1-0.7358z^{-1}+0.1353z^{-2}, \quad B\left(z^{-1}\right)=0.2642+0.1353z^{-1}$$

If we multiply the difference equation by Δ term we get CARIMA process model

$$y(k)-1.7358\,y(k-1)+0.8711\,y(k-2)-0.1353\,y(k-3)=0.2642\,\Delta u(k-1)+0.1353\,\Delta u(k-2) \tag{9}$$

$$\Delta A\left(z^{-1}\right)=\left(1-z^{-1}\right)\left(1-0.7358z^{-1}+0.1353z^{-2}\right)=1-1.7358z^{-1}+0.8711z^{-2}-0.1353z^{-3}$$

We use filtering polynomial (let us consider coloured noise case)

$$C\left(z^{-1}\right)=T\left(z^{-1}\right)=\left(1-0.8z^{-1}\right)^2=1-1.6z^{-1}+0.64z^{-2} \tag{10}$$

One-step ahead optimal plant output prediction is

$$\begin{aligned}\hat{y}(k+1)&=z\left[1-\Delta A\left(z^{-1}\right)\right]y_m(k)+B\left(z^{-1}\right)\Delta u(k)+z\left[T\left(z^{-1}\right)-1\right]e(k)=\\&=1.7358y_m(k)-0.8711y_m(k-1)+0.1353y_m(k-2)+0.2642\Delta u(k)+0.1353\Delta u(k-1)-\\&-1.6e(k)+0.64e(k-1)\end{aligned} \tag{11}$$

where $e$(k) is prediction error $e(k)=y_m(k)-\hat{y}(k)$ and $y_m(k)$ is measured plant output. We suppose zero future prediction errors, that means $e(k+1)=e(k+2)=\ldots=e(k+N)=0$

Two-step ahead optimal plant output prediction is

$$\begin{aligned}\hat{y}(k+2)&=1.7358\hat{y}(k+1)-0.8711y_m(k)+0.1353y_m(k-1)+0.2642\Delta u(k+1)+0.1353\Delta u(k)+\\&+0.64e(k)\end{aligned} \tag{12}$$

Three-step ahead optimal plant output prediction is

$$\hat{y}(k+3)=1.7358\hat{y}(k+2)-0.8711\hat{y}(k+1)+0.1353y_m(k)+0.2642\Delta u(k+2)+0.1353\Delta u(k+1) \tag{13}$$

Prediction equations rewritten in matrix form are

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1.7358 & 1 & 0 & 0 & 0 \\ 0.8711 & -1.7358 & 1 & 0 & 0 \\ -0.1353 & 0.8711 & -1.7358 & 1 & 0 \\ 0 & -0.1353 & 0.8711 & -1.7358 & 1 \end{bmatrix}}_{\mathbf{A}} \cdot \begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+3) \\ \hat{y}(k+4) \\ \hat{y}(k+5) \end{bmatrix} =$$

$$\begin{bmatrix} 0.2642 & 0 & 0 & 0 & 0 \\ 0.1353 & 0.2642 & 0 & 0 & 0 \\ 0 & 0.1353 & 0.2642 & 0 & 0 \\ 0 & 0 & 0.1353 & 0.2642 & 0 \\ 0 & 0 & 0 & 0.1353 & 0.2642 \end{bmatrix} \cdot \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \Delta u(k+3) \\ \Delta u(k+4) \end{bmatrix} + \begin{bmatrix} 0.1353 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \cdot \Delta u(k-1) +$$

$$+ \begin{bmatrix} 1.7358 & -0.8711 & 0.1353 \\ -0.8711 & 0.1353 & 0 \\ 0.1353 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} y_m(k) \\ y_m(k-1) \\ y_m(k-2) \end{bmatrix} + \begin{bmatrix} -1.6 & 0.64 \\ 0.64 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} e(k) \\ e(k-1) \end{bmatrix} \quad (14)$$

After multiplying equation by $\mathbf{A}^{-1}$ we get plant output prediction in the for of equation (3)

$$\underbrace{\begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+3) \\ \hat{y}(k+4) \\ \hat{y}(k+5) \end{bmatrix}}_{\mathbf{Y}_{N12}} = \underbrace{\begin{bmatrix} 0.2642 & 0 & 0 & 0 & 0 \\ 0.5940 & 0.2642 & 0 & 0 & 0 \\ 0.8009 & 0.5940 & 0.2642 & 0 & 0 \\ 0.9084 & 0.8009 & 0.5940 & 0.2642 & 0 \\ 0.9596 & 0.9084 & 0.8009 & 0.5940 & 0.2642 \end{bmatrix}}_{\mathbf{G}_{N123}} \cdot \underbrace{\begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \Delta u(k+3) \\ \Delta u(k+4) \end{bmatrix}}_{\Delta\mathbf{U}_{N3}} +$$

$$+ \underbrace{\begin{bmatrix} 0.1353 \\ 0.2349 \\ 0.2899 \\ 0.3168 \\ 0.3292 \end{bmatrix}}_{\mathbf{G'}_{N12}} \cdot \underbrace{\Delta u(k-1)}_{\Delta\mathbf{U}_p} + \underbrace{\begin{bmatrix} 1.7358 & -0.8711 & 0.1353 \\ 2.1418 & -1.3767 & 0.2349 \\ 2.3409 & -1.6308 & 0.2899 \\ 2.4325 & -1.7493 & 0.3168 \\ 2.4729 & -1.8021 & 0.3292 \end{bmatrix}}_{\mathbf{F}_{N12}} \cdot \underbrace{\begin{bmatrix} y_m(k) \\ y_m(k-1) \\ y_m(k-2) \end{bmatrix}}_{\mathbf{Y}_p} +$$

$$+ \underbrace{\begin{bmatrix} -1.6000 & 0.6400 \\ -2.1372 & 1.1109 \\ -2.3159 & 1.3707 \\ -2.3747 & 1.4982 \\ -2.3938 & 1.5568 \end{bmatrix}}_{\mathbf{E}_{N12}} \cdot \underbrace{\begin{bmatrix} e(k) \\ e(k-1) \end{bmatrix}}_{\mathbf{E}_p} \quad (15)$$

## 5. CONCLUSION

Simple method how to calculate prediction equations from CARIMA input-output process model is presented in the paper. If the process is identical we must get identical prediction equations but the way can how to reach them can differ – we can use different mathematical apparatus.

Discussed approach is very illustrative and easy to understand. From this point of view it is suitable for "MPC beginners" and students. Algorithm of this method is easy to develop and program as well as for multidimensional systems and coloured noise case. The solution is possible by matrix inversion (to solve whole problem at once) or in an iterative way by use of Toeplitz/Hankel matrix algebra (as an analogy to Diophantine equations solving).

*The research has been supported in the program of Czech Republic MSM 0021627505. This supports is very gratefully acknowledged.*

## REFERENCES

[1] CLARKE, D.W., MOHTADI, C., TUFFS, P.S. Generalized Predictive Control – Part I. The Basic Algorithm. *Automatica*, vol. 23, no. 2, p. 137-148, 1987.

[2] CAMACHO, E.F., BORDONS, C. *Model Predictive Control*. London: Springer-Verlag London Limited, Great Britain, 280 pp., 2002.

[3] HONC, D., HABER, R. Multivariable predictive control in state space without observer. In: *15th International Conference on Process Control '05*, Štrbské pleso, Slovak Republic, June 7 – 10 2005, Slovak University of Technology, Bratislava, 2005, p. 207 (CD ROM 028-1-6).

[4] ROSSITER, J.A. Model-Based Predictive Control, A Practical Approach. CRC Press LLC, United States, 318 pp., 2003.

Recenzent: doc. Ing. František Dušek, CSc.

Univerzita Pardubice, FEI, Katedra řízení procesů