JST: Smart Systems and Devices
Volume 32, Issue 1, January 2022, 042-050

Alternative Generalized Predictive Control for Output Disturbed Multi-Input Multi-Output Discrete-Time Systems

Tuan Anh Nguyen, Thu Ha Nguyen*, Lan Anh Dinh Thi, Doan Phuoc Nguyen
School of Electrical Engineering, Hanoi University of Science and Technology, Hanoi, Vietnam
**Email: ha.nguyenthu3@hust.edu.vn*

Abstract

This article proposes an option to execute conveniently the traditional Model Predictive Control (GPC), called the Alternative Generalized Predictive Control (AGPC). In this AGPC the disturbed discrete-time input-output mapping of controlled plant is utilized directly for the prediction of plant outputs, instead of its transfer function as being done in the conventional approach. Hence, solving Diophantine equations will be avoided. Within this AGPC all recorded values of plant inputs/outputs in the last time-horizon are matched into separate vectors for computing predictive control signals at the next control step, which helps therefore that its implementation becomes more manageable. To verify via virtually real simulation the control performance of this proposed AGPC a Simscape water tank model, which is chosen as the controlled plant, had been created among Thermal Fluids Toolbox. The simulation is carried out for two different circumstances, one by using AGPC and the other by applying conventional PID, for comparison purposes. The simulation demonstrates also how to realize this AGPC in practice.

Keywords: MPC, alternative GPC, PID, optimization, virtually real tank model.

1. Introduction

Model Predictive Control (MPC) is a control concept of using a dynamic model of processes to predict the signal control by minimizing an appropriate cost function [1-4]. But different to conventional optimal controller such as LQR (Linear Quadratic Regulator) for linear processes or DP (Dynamic Programming) for non-linear processes in general [5], with MPC concept it is allowed the tracking performance of processes in current control time instant to be optimized while keeping future tracking errors in account. This is achieved by optimizing an appropriate cost-function belonging to a fixed time-horizon. Afterward, only the control signals at the current time instant are implemented. This optimization will be executed repeatedly step by step for receiving control signals in the next timeslots [1]. The control concept of MPC had shown its superiority in process control, compared to other methods such as Minimum Variance (MV), Smith Predictor, Generalized Minimum Variance GMV… [1-10].

At present there are various MPC algorithms available for industrial applications, such as Model Algorithmic Control (MAC) [1], Dynamic Matrix Control (DMC) [3], Generalized Predictive Control (GPC) [4], Predictive Control in state space [2,11], Adaptive Horizon Model Predictive Control (AHMPC) [6] and Extended Prediction Self-Adaptive Control (EPSAC) [12]. However, in general, any control algorithm of MPC always consists of three main elements: the predictive model for forecasting the process outputs, the objective function for characterizing the process tracking performance, and the optimization algorithm for solving this objective function. All aforementioned algorithms are distinguished from each other mainly in how to realize these three elements in it, accordingly to the particular dynamic of controlled processes [7-12]. For example, both MAC [1] and DMC algorithm [3] are open-loop controllers, in which the time response is used as a process model. Hence, they are only applicable if the processes are linear and the models are sufficiently exact. In the contrary GPC [4] is a closed-loop controller and established based on the transfer function model. Therefore, it could be applied for linear processes with disturbances. For non-linear processes, the multi predictive linear model around operating points is used to forecast future outputs [8-10]. With these approaches and since the original nonlinear process model is approximated by a chain of linear, time-invariant sub-models around a series of operating points, the desired control performances could not hold anymore if these operating points are not determined sufficiently precisely online. Moreover, the oscillation would be happening inevitability there, if the control has to switch from one LTI subsystem to another one. This is known as the main disadvantage of multiple predictive LTI model-based methods.

ISSN: 2734-9373
https://doi.org/10.51316/jst.155.ssad.2022.32.1.6
Received: May 29, 2021; accepted: September 7, 2021

With the assumption that the controlled processes could be approximately linear with matched model error, this article will propose a substitutive option of conventional GPC, called the Alternative GPC (AGPC), in which the solving required Diophantine equations in GPC will be avoided. Furthermore, similarly with GPC, this proposed AGPC can be implemented also to both SISO and MIMO systems in an output feedback control scheme. But in contrary, the proposed AGPC method can be applied for processes modelled by uncertain input-output mapping with matched model errors, instead of converting it in such a structure, where input signals of which have to satisfy the requirement for being applicable of Diophantine equations [1,4]. Hence, with this proposed AGPC it is no need to solve Diophantine equations by applying the conventional GPC. Consequently, the real-time response of closed-loop controlled systems by using AGPC would be improved.

The main idea of AGPC is as below. Starting from the original uncertain input-output mapping, the future vectors of all system outputs $\underline{y}_{k+i}$ in the prediction horizon N will be represented as a matched disturbed function depending only on all input vectors at the present fixed time horizon $\underline{u}_k \div \underline{u}_{k+N-1}$ by applying an iterative calculation method. The remaining system error in the predictive model will be estimated optimality for compensating purposes. These representations of future system outputs will be then concatenated into objective function for determining general inputs at the present time instance by using an optimization algorithm.

For an effective verification of the control performance of this proposed AGPC, in this article, the simulation plant is a virtually real tank model. This model is created with Simscape Thermal Fluids Toolbox, which is available in MathWorks. So it can be implemented in Simulink as connected thermodynamics blocks. The dynamic of the model is nonlinear, with a strong influence between system state variables. So it would be suitable for authenticating the AGPC behaviour.

The control task is to keep tracking the level and the temperature of water in tank to the desired set points. For a true comparison purpose, besides using AGPC algorithm, this article implements also the traditional PID to control the same plant. Parameters of PID controllers are determined with the help of an optimization approach for minimizing tracking errors. The comparison is carried out by evaluating the response time, settling time, overshoot, and steady state error. Obtained simulation results will show visually the advantages and disadvantages of AGPC algorithm compared with traditional PID controller.

The rest of this article is organized as follows. In Section 2 all theoretical substances related to AGPC are presented. A virtually real water tank model is established in Section 3. Numerical simulations and discussions are shown in Section 4 to authenticate the performance of AGPC in comparison with PID. Final Section provides conclusions and future works.

2. Alternative GPC

Consider a linear time invariant, discrete time plant with q inputs and p output, described by input-output mapping as follows:

$$\begin{aligned}\underline{y}_k + A_{1,0}\underline{y}_{k-1} + \ldots + A_{n,0}\underline{y}_{k-n} + \underline{d}_k \\ = B_{0,0}\underline{u}_k + B_{1,0}\underline{u}_{k-1} + \ldots + B_{m,0}\underline{u}_{k-m}\end{aligned} \quad (1)$$

which is disturbed additionally in outputs. There are in this mapping:

$\underline{u}_k = \left[u_1(k),\ \ldots\ ,u_q(k)\right]^T \in \mathbb{R}^q$ is the vector of q inputs,

$\underline{y}_k = \left[y_1(k),\ \ldots\ ,y_p(k)\right]^T \in \mathbb{R}^p$ is the vector of p outputs,

$A_{1,0},\ \ldots\ ,A_{n,0}, B_{0,0},\ \ldots\ ,B_{m,0}$ represent all model parameters,

$\underline{d}_k$ is the vector of output disturbances,

and k denotes the current time instant.

The paper aims to design an output feedback controller such that p system outputs $\underline{y}_k$ of (1) will convergence asymptotically to any desired references $\underline{r}_k$ and this tracking performance must not be affected by the disturbances $\underline{d}_k$.

2.1. Predictive Model

According to the system model (1) it is clear that all p system outputs $\underline{y}_k$ at the current time instant k can be established approximately from system data in the past and from current system inputs $\underline{u}_k$ as below:

$$\begin{aligned}\underline{y}_k = &\left(B_{0,0}\underline{u}_k + B_{1,0}\underline{u}_{k-1} + \cdots + B_{m,0}\underline{u}_{k-m}\right) \\ &-\left(A_{1,0}\underline{y}_{k-1} + \cdots + A_{n,0}\underline{y}_{k-n} + \underline{d}_k^{/}\right)\end{aligned} \quad (2)$$

where $\underline{d}_k^{/}$ denotes an estimation of $\underline{d}_k$.

Assume that $\underline{d}_k'$ can be considered as a constant during whole current control horizon $k,\ \ldots\ ,k+N$, then according to the approximation (1) all predictive system outputs $\underline{y}_{k+i},\ i=1,2,\ \ldots\ ,N$ in this horizon can be calculated as follows:

1) For $i=1$:

$$\begin{aligned}\underline{y}_{k+1} &= \left(B_{0,0}\underline{u}_{k+1} + B_{1,0}\underline{u}_k + \ldots + B_{m,0}\underline{u}_{k+1-m}\right) \\ &\quad -\left(A_{1,0}\underline{y}_k + A_{2,0}\underline{y}_{k-1} + \ldots + A_{n,0}\underline{y}_{k+1-n} + \underline{d}'_k\right) \\ &= \left(B_{0,0}\underline{u}_{k+1} + B_{1,0}\underline{u}_k + \ldots + B_{m,0}\underline{u}_{k+1-m}\right) \\ &\quad -\Big\{A_{1,0}\Big[\left(B_{0,0}\underline{u}_k + B_{1,0}\underline{u}_{k-1} + \ldots + B_{m,0}\underline{u}_{k-m}\right) \\ &\quad -\left(A_{1,0}\underline{y}_{k-1} + \ldots + A_{n,0}\underline{y}_{k-n} + \underline{d}'_k\right)\Big] \\ &\quad + A_{2,0}\underline{y}_{k-1} + \ldots + A_{n,0}\underline{y}_{k+1-n} + \underline{d}'_k\Big\} \\ &= (B_{0,1}\underline{u}_{k+1} + B_{1,1}\underline{u}_k + \ldots + B_{m+1,1}\underline{u}_{k-m}) \\ &\quad -(A_{1,1}\underline{y}_{k-1} + \ldots + A_{n,1}\underline{y}_{k-n} + D_1\underline{d}'_k)\end{aligned} \tag{3}$$

where

$$\begin{aligned}&B_{0,1} = B_{0,0},\ B_{1,1} = B_{1,0} - A_{1,0}B_{0,0},\ \ldots \\ &B_{m,1} = B_{m,0} - A_{1,0}B_{m-1,0},\ B_{m+1,1} = -A_{1,0}B_{m,0},\end{aligned} \tag{4}$$

and

$$\begin{aligned}&A_{1,1} = A_{2,0} - A_{1,0}A_{1,0},\ A_{2,1} = A_{3,0} - A_{1,0}A_{2,0},\ \ldots \\ &A_{n-1,1} = A_{n,0} - A_{1,0}A_{n-1,0},\ A_{n,1} = -A_{1,0}A_{n,0}, \\ &D_1 = I_p - A_{1,0}\end{aligned} \tag{5}$$

2) For $i = 2$:

$$\begin{aligned}\underline{y}_{k+2} &= \left(B_{0,1}\underline{u}_{k+2} + B_{1,1}\underline{u}_{k+1} + \ldots + B_{m+1,1}\underline{u}_{k+1-m}\right) \\ &\quad -\left(A_{1,1}\underline{y}_k + \ldots + A_{n,0}\underline{y}_{k+1-n} + D_1\underline{d}'_k\right) \\ &= \left(B_{0,1}\underline{u}_{k+2} + B_{1,1}\underline{u}_{k+1} + \ldots + B_{m+1,1}\underline{u}_{k+1-m}\right) \\ &\quad -\Big\{A_{1,1}\Big[\left(B_{0,0}\underline{u}_k + B_{1,0}\underline{u}_{k-1} + \ldots + B_{m,0}\underline{u}_{k-m}\right) \\ &\quad -\left(A_{1,0}\underline{y}_{k-1} + \ldots + A_{n,0}\underline{y}_{k-n} + \underline{d}'_k\right)\Big] \\ &\quad + A_{2,1}\underline{y}_{k-1} + \ldots + A_{n,1}\underline{y}_{k+1-n} + D_1\underline{d}'_k\Big\} \\ &= (B_{0,2}\underline{u}_{k+2} + B_{1,2}\underline{u}_{k+1} + \ldots + B_{m+2,2}\underline{u}_{k-m}) \\ &\quad -(A_{1,2}\underline{y}_{k-1} + \ldots + A_{n,2}\underline{y}_{k-n} + D_2\underline{d}'_k)\end{aligned} \tag{6}$$

where

$$\begin{aligned}&B_{0,2} = B_{0,1},\ B_{1,2} = B_{1,1},\ B_{2,2} = B_{2,1} - A_{1,1}B_{0,0},\ \ldots \\ &B_{m+1,2} = B_{m+1,1} - A_{1,1}B_{m-1,0},\ B_{m+2,2} = -A_{1,1}B_{m,0}, \\ &A_{1,2} = A_{2,1} - A_{1,1}A_{1,0},\ A_{2,2} = A_{3,1} - A_{1,0}A_{2,0},\ \ldots \\ &A_{n-1,2} = A_{n,0} - A_{1,1}A_{n-1,0},\ A_{n,2} = -A_{1,1}A_{n,0}, \\ &D_2 = D_1 - A_{1,1}\end{aligned} \tag{7}$$

3) For $i = 3,\ \ldots\ ,N$:

$$\begin{aligned}\underline{y}_{k+i} &= \left(B_{0,i}\underline{u}_{k+i} + B_{1,i}\underline{u}_{k+i-1} + \ldots + B_{m+i,i}\underline{u}_{k-m}\right) \\ &\quad -\left(A_{1,i}\underline{y}_{k-1} + \ldots + A_{n,i}\underline{y}_{k-n} + D_i\underline{d}'_k\right)\end{aligned} \tag{8}$$

where

$$\begin{aligned}&B_{0,i} = B_{0,i-1},\ B_{1,i} = B_{1,i-1},\ \ldots \\ &B_{i-1,i} = B_{i-1,i-1},\ B_{i,i} = B_{i,i-1} - A_{1,i-1}B_{0,0},\ \ldots \\ &B_{m+i-1,i} = B_{m+i-1,i-1} - A_{1,i-1}B_{m-1,0}, \\ &B_{m+i,i} = -A_{1,i-1}B_{m,0}, \\ &A_{1,i} = A_{2,i-1} - A_{1,i-1}A_{1,0}, \\ &A_{2,i} = A_{3,i-1} - A_{1,i-1}A_{2,0},\ \ldots \\ &A_{n-1,i} = A_{n,i-1} - A_{1,i-1}A_{n-1,0},\ A_{n,i} = -A_{1,i-1}A_{n,0}, \\ &D_i = D_{i-1} - A_{1,i-1}\end{aligned} \tag{9}$$

Obviously, both expressions (3), (6) for the prediction of $\underline{y}_{k+1}, \underline{y}_{k+2}$ as well as calculating recursively (5) and (7) for their parameters, are contained in (8) and (9). Therefore the recursive equation (9) can be also used to determine all parameters $B_{0,i},\ \ldots\ ,B_{m+i,i}, A_{1,i},\ \ldots\ ,A_{n,i}$ and D_i of all N predictive system outputs $\underline{y}_{k+1}$ given in (8).

The formal combination of all predictive outputs $\underline{y}_{k+i},\ i = 1,2,\ \ldots\ ,N$ given in (8) implies prediction model among current horizon as below

$$\underline{\mathbf{y}} = \mathbf{Y}\underline{\mathbf{u}} + \underline{\mathbf{g}} \tag{10}$$

with all elements $\underline{\mathbf{y}}, \mathbf{Y}, \underline{\mathbf{u}}$ and $\underline{\mathbf{g}}$ in it are defined precisely as below

$$\begin{aligned}\underline{\mathbf{g}} &= \begin{bmatrix} B_{1,0} & B_{2,0} & \cdots & B_{m,0} \\ B_{2,1} & B_{3,1} & \cdots & B_{m+1,1} \\ \vdots & \vdots & \ddots & \vdots \\ B_{N+1,N} & B_{N+2,N} & \cdots & B_{m+N,N}\end{bmatrix}\begin{bmatrix}\underline{u}_{k-1} \\ \underline{u}_{k-2} \\ \vdots \\ \underline{u}_{k-m}\end{bmatrix} \\ &\quad - \begin{bmatrix} A_{1,0} & A_{2,0} & \cdots & A_{n,0} \\ A_{1,1} & A_{2,1} & \cdots & A_{n,1} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1,N-1} & B_{2,N-1} & \cdots & B_{n,N-1}\end{bmatrix}\begin{bmatrix}\underline{y}_{k-1} \\ \underline{y}_{k-2} \\ \vdots \\ \underline{y}_{k-n}\end{bmatrix} \\ &\quad - \begin{bmatrix} I_p \\ D_1 \\ \vdots \\ D_N\end{bmatrix}\underline{d}'_k \\ &= \mathbf{B}\underline{\mathbf{u}}_b - \mathbf{A}\underline{\mathbf{y}}_b - \mathbf{D}\underline{d}'_k\end{aligned} \tag{11}$$

and

$$\underline{\mathbf{y}} = \begin{bmatrix}\underline{y}_k \\ \underline{y}_{k+1} \\ \cdots \\ \underline{y}_{k+N}\end{bmatrix},\ \underline{\mathbf{u}} = \begin{bmatrix}\underline{u}_{k+N} \\ \underline{u}_{k+N-1} \\ \vdots \\ \underline{u}_k\end{bmatrix}$$
$$\mathbf{Y} = \begin{bmatrix}\Theta & \cdots & \Theta & B_{0,0} \\ \Theta & \cdots & B_{0,1} & B_{1,1} \\ \vdots & \ddots & \vdots & \vdots \\ B_{0,N} & \cdots & B_{N-1,N} & B_{N,N}\end{bmatrix} \tag{12}$$

where

$$\mathbf{B}=\begin{bmatrix} B_{1,0} & B_{2,0} & \cdots & B_{m,0} \\ B_{2,1} & B_{3,1} & \cdots & B_{m+1,1} \\ \vdots & \vdots & \ddots & \vdots \\ B_{N+1,N} & B_{N+2,N} & \cdots & B_{m+N,N} \end{bmatrix},$$

$$\mathbf{A}=\begin{bmatrix} A_{1,0} & A_{2,0} & \cdots & A_{n,0} \\ A_{1,1} & A_{2,1} & \cdots & A_{n,1} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1,N-1} & B_{2,N-1} & \cdots & B_{n,N-1} \end{bmatrix}, \quad (13)$$

$$\mathbf{D}=\begin{bmatrix} I_p \\ D_1 \\ \vdots \\ D_N \end{bmatrix},\ \underline{u}_b=\begin{bmatrix} \underline{u}_{k-1} \\ \underline{u}_{k-2} \\ \vdots \\ \underline{u}_{k-m} \end{bmatrix},\ \underline{y}_b=\begin{bmatrix} \underline{y}_{k-1} \\ \underline{y}_{k-2} \\ \vdots \\ \underline{y}_{k-n} \end{bmatrix}.$$

In afore mentioned equations Θ denotes the matrix of all zero entries (zero matrix) and I_p is the identity matrix of dimension $p \times p$

The equation (10) is just the needed prediction model, which will be utilized hereafter for calculating the optimal inputs.

2.2. Disturbance Estimation for Rejecting

From the original input-output mapping (1) it is clear that:

$$\underline{d}_k=\left(B_{0,0}\underline{u}_k+\ \dots\ +B_{m,0}\underline{u}_{k-m}\right) -\left(\underline{y}_k+A_{1,0}\underline{y}_{k-1}+\ \dots\ +A_{n,0}\underline{u}_{k-n}\right)$$

Hence, under the assumption that disturbances are sufficiently slow, it could be approximated accordingly as

$$\begin{aligned}\underline{d}_k^{/} &\approx \underline{d}_{k-1} \\ &=\left(B_{0,0}\underline{u}_{k-1}+\ \cdots\ +B_{m,0}\underline{u}_{k-m-1}\right) \\ &\quad -\left(\underline{y}_{k-1}+A_{1,0}\underline{y}_{k-2}+\ \cdots\ +A_{n,0}\underline{u}_{k-n-1}\right) \\ &=\left[B_{0,0},\ \dots\ ,B_{m,0}\right]\underline{\mathbf{u}}_b^{/}- \\ &\quad -\left[I_p,A_{1,0},\ \dots\ ,A_{n,0}\right]\underline{\mathbf{y}}_b^{/}\end{aligned} \quad (14)$$

where

$$\underline{\mathbf{u}}_b^{/}=\begin{bmatrix} \underline{u}_{k-1} \\ \underline{u}_{k-2} \\ \vdots \\ \underline{u}_{k-m-1} \end{bmatrix}=\begin{bmatrix} \underline{\mathbf{u}}_b \\ \underline{u}_{k-m-1} \end{bmatrix},\ \underline{\mathbf{y}}_b^{/}=\begin{bmatrix} \underline{y}_{k-1} \\ \underline{y}_{k-2} \\ \vdots \\ \underline{y}_{k-n-1} \end{bmatrix}=\begin{bmatrix} \underline{\mathbf{y}}_b \\ \underline{y}_{k-n-1} \end{bmatrix}.$$

2.3. Objective Function

According to the control task of $\underline{y}_k \to \underline{r}_k$, i.e. of $\underline{\mathbf{e}}=\underline{\mathbf{y}}-\underline{\mathbf{r}}\to\underline{\mathbf{0}}$, where $\underline{\mathbf{0}}$ denotes the zeros vector of dimension $(N+1)p$, and

$$\underline{\mathbf{r}}=\mathrm{vec}\left[\underline{r}_k,\underline{r}_{k+1},\ \dots\ ,\underline{r}_{k+N}\right] \quad (15)$$

the following objective function will be used

$$J_k=\left(\underline{\mathbf{y}}-\underline{\mathbf{r}}\right)^T Q_k\left(\underline{\mathbf{y}}-\underline{\mathbf{r}}\right)+\underline{\mathbf{u}}^T R_k\underline{\mathbf{u}}\to \min . \quad (16)$$

Substituting (10) in (16) yields

$$J_k=\left(\mathbf{Y}\underline{\mathbf{u}}+\underline{\mathbf{g}}-\underline{\mathbf{r}}\right)^T Q_k\left(\mathbf{Y}\underline{\mathbf{u}}+\underline{\mathbf{g}}-\underline{\mathbf{r}}\right)+\underline{\mathbf{u}}^T R_k\underline{\mathbf{u}}\ ,$$

which is obviously equivalent with

$$J_k=\underline{\mathbf{u}}^T\left(\mathbf{Y}^T Q_k\mathbf{Y}+R_k\right)\underline{\mathbf{u}}-2\left(\underline{\mathbf{r}}-\underline{\mathbf{y}}\right)^T Q_k\mathbf{Y}\underline{\mathbf{u}}\ . \quad (17)$$

To ensure that the optimization problem (16) is solvable, in the aforementioned objective function both matrices Q_k, R_k must be positive definite. Except that, in principle, they are chosen arbitrarily.

2.4. Optimization

Because the optimization problem (16), i.e. (17) is an unconstrained and quadratic problem, it yields immediately that

$$\underline{\mathbf{u}}=\left(\mathbf{Y}^T Q_k\mathbf{Y}+R_k\right)^{-1}\mathbf{Y}^T Q_k\left(\underline{\mathbf{r}}-\underline{\mathbf{y}}\right) \quad (18)$$

which implies therefore the needed control value $\underline{u}_k$ at the current time instant k as follows

$$\underline{u}_k=\left[\Theta,\dots,\Theta,I_p\right]\underline{\mathbf{u}}\ . \quad (19)$$

2.5. Opportunity to Improve Control Performance

The control value $\underline{u}_k$ from (19) contains in it freely chosen symmetric positive definite matrices Q_k, R_k. Hence, it arises here some opportunities to select them appropriately so that the control performance of closed-loop system will be improved.

Some beneficial hints are in following:

- The bigger R_k is chosen, the smaller $|\underline{u}_k|$ will be.
- The smaller Q_k is chosen, the larger component of R_k will take part in objective function J defined in (16) and hence $|\underline{u}_k|$ will become also smaller. On the contrary, the slower tracking process $\underline{\mathbf{y}}\to\underline{\mathbf{r}}$ will be.
- The bigger Q_k is chosen, the smaller overshoot will be.

2.6. Control Algorithm

In order to facilitate the implementation of proposed controller the following algorithm is established.

1) Choose $N \geq 2$. Using (9) to determine all matrices $A_{j,i}, B_{l,s}, D_z$ for $1 \leq j \leq n$, $0 \leq i \leq N$ and $0 \leq s \leq N$.

Establish $\mathbf{Y}, \mathbf{A}, \mathbf{B}, \mathbf{D}$ accordingly to (12) and (13) respectively, Create two arrays $\underline{\mathbf{u}}_b^{/} \in \mathrm{R}^{(m+1)q}$ and $\underline{\mathbf{y}}_b^{/} \in \mathrm{R}^{(n+1)p}$. Set $k=0$ and $\underline{\mathbf{u}}_b^{/} = \underline{\mathbf{0}}$, $\underline{\mathbf{y}}_b^{/} = \underline{\mathbf{0}}$.

Determine two arrays $\underline{\mathbf{u}}_b = \left[I_{mq}, \Theta_{mq\times q}\right]\underline{\mathbf{u}}_b^{/}$ and $\underline{\mathbf{y}}_b = \left[I_{np}, \Theta_{np\times p}\right]\underline{\mathbf{y}}_b^{/}$, where $\Theta_{i\times j}$ is the zero matrix of dimension $i\times j$ and I_{ij} is $(ij\times ij)$ identity matrix. It means that $\underline{\mathbf{u}}_b, \underline{\mathbf{y}}_b$ are two arrays of mq and np elements, which are truncated from $\underline{\mathbf{u}}_b^{/}$, $\underline{\mathbf{y}}_b^{/}$, respectively. Choose arbitrarily two symmetric positive definite matrices Q, R and in scenario of constrained control two positive factors $0 < \mu \leq 1$, $1 < \eta$.

2) Estimate $\underline{\mathbf{d}}_k^{/}$ with (14). Calculate $\underline{\mathbf{g}}$ and $\underline{\mathbf{r}}$ accordingly to (11), (15), respectively.

3) Determine $\underline{u}_k$ by using (18) and (19).

4) In the circumstance of control problem with constraints, do this step, otherwise, skip it.

If $\underline{u}_k$ does not satisfy the required constraint $|\underline{u}_k| \leq \underline{U}$, then set $R := \eta R$ and go back to step 3.

5) Send $\underline{u}_k$ to the original plant (1) for a while of sampling time.

6) Measure the output $\underline{y}_k$ from system (1). Reorder two arrays $\underline{\mathbf{u}}_b^{/}$, $\underline{\mathbf{y}}_b^{/}$ as follows

$\underline{u}_i \leftarrow \underline{u}_{i-1}$, $i = 2, \ldots, m+1$

$\underline{y}_j \leftarrow \underline{y}_{j-1}$, $j = 2, \ldots, n+1$

$\underline{u}_1 \leftarrow \underline{u}_k$ and $\underline{y}_1 \leftarrow \underline{y}_k$

Determine $\underline{\mathbf{u}}_b, \underline{\mathbf{y}}_b$ from $\underline{\mathbf{u}}_b^{/}$, $\underline{\mathbf{y}}_b^{/}$ with

$\underline{\mathbf{u}}_b = \left[I_{mq}, \Theta_{mq\times q}\right]\underline{\mathbf{u}}_b^{/}$, $\underline{\mathbf{y}}_b = \left[I_{np}, \Theta_{np\times p}\right]\underline{\mathbf{y}}_b^{/}$.

Set $Q := \mu Q$ (in scenario of constrained control). Go back to step 2.

It is clear that based on minimizing the objective function (16) and since both Q_k, R_k are positive definite, the bigger horizon N is chosen, the smaller tracking error $\underline{\mathbf{e}}_k = \underline{\mathbf{y}} - \underline{\mathbf{r}}$ belong current time horizon will be. Furthermore, the established control algorithm produces monotonously decreasing sequences $\|\underline{\mathbf{e}}_k\|$ and $\|\underline{\mathbf{u}}\|$. Hence, by disregarding system disturbances the closed-loop system tends to an equilibrium point. Because the closed-loop system with $\underline{d} = \underline{0}$ is linear with a regular system matrix, it has only one equilibrium point at origin. Therefore, the control algorithm above satisfies $\|\underline{\mathbf{e}}_k\| \to 0$.

3. Virtually Real Water Tank Model

3.1. Physical Model and Its Linearization

The water tank system diagram set up in Simulink is exhibited in Fig.1, where the water level in the tank is $h[m]$ and the temperature is $T[°C]$. There are two streams of water flowing into the tank: hot water flow with flow rate is $F_h[m^3/s]$ and temperature is $T_h[°C]$; cooling water flow with flow rate is $F_c[m^3/s]$ and temperature is $T_c[°C]$. There is also an outflow flow with flow rate $F_o[m^3/s]$. The system dynamic model is described by the two balance equations: mass and energy balance. The mass balance equation is described as following

$$A\frac{dh}{dt} = F_h + F_c - F_o \tag{20}$$

here $A[m^2]$ is cross-sectional area of the tank.

The energy balance equation is

$$U = U_{in} - U_{out} \tag{21}$$

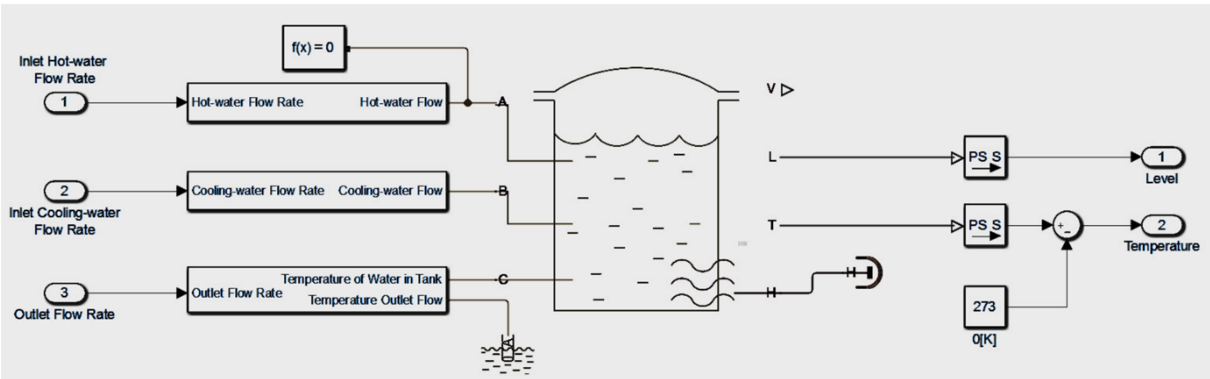

Fig.1. MATLAB Simulink Simscape Tank Model

Assuming that (i) there is no heat exchange between water tank and the environment, (ii) the temperature at anywhere inside the tank has the same value, we obtain

$$\begin{aligned} \frac{d\left(\rho Ah\times C\times T\right)}{dt} &= \rho_h F_h \times C_h \times T_h \\ &+ \rho_c F_c \times C_c \times T_c - \rho F_o \times C \times T \end{aligned} \tag{22}$$

where ρ, ρ_h, ρ_c is the density of water in the tank, hot water flow, and cooling water flow, C, C_h, C_c is the specific heat of water in the tank, hot water flow, and cooling water flow. Assume furthermore that $C = C_h = C_c$ and $\rho = \rho_c = \rho_h$, then (22) becomes

$$\begin{aligned} & A\frac{d\left(hT\right)}{dt} = F_h T_h + F_c T_c - F_o T \\ \Rightarrow\quad & AT\frac{dh}{dt} + Ah\frac{dT}{dt} = F_h T_h + F_c T_c - F_o T \end{aligned} \tag{23}$$

The substitution (20) in (23) yields

$$\begin{aligned} & \left(F_h + F_c - F_o\right)T + Ah\frac{dT}{dt} = F_h T_h + F_c T_c - F_o T \\ \Rightarrow\quad & Ah\frac{dT}{dt} = F_h\left(T_h - T\right) + F_c\left(T_c - T\right) \end{aligned} \tag{24}$$

From (20) and (24), the dynamics of tank system is described as follow

$$\begin{cases} A\times \dot{h} = F_h + F_c - F_o \\ Ah\times \dot{T} = F_h\left(T - T_h\right) + F_c\left(T - T_c\right) \end{cases} \tag{25}$$

The aim of control problem here is to keep the water level h and temperature T tracking to the set point. The simplest solution to this is to linearize the equation (25) then apply linear controls methods. The first step in linearizing the equations (25) is to approximate process variables

$$\begin{cases} h = \overline{h} + \tilde{h}; T = \overline{T} + \tilde{T} \\ F_h = \overline{F}_h + \widetilde{F}_h; F_c = \overline{F}_c + \widetilde{F}_c \end{cases} \tag{26}$$

where the bar $(\overline{\ })$ is the value at the operating point and $(\tilde{\ })$ is the term represents incremental variations around this point.

For the simulation, all parameters of system at operating point are given as

$\overline{h} = 0.1[m]$, $\overline{T} = 45[°C]$, $\overline{F}_c = 2\times 10^{-5}\,[m^3/s]$

and

$\overline{F}_h = 2\times 10^{-5}\,[m^3/s]$.

The second step is to find the matrix of transfer functions

$$\underline{\tilde{y}} = \mathbf{G}\,\underline{\tilde{u}} \tag{27}$$

where

$$\underline{\tilde{y}} = \begin{bmatrix} \tilde{h} \\ \tilde{T} \end{bmatrix},\ \underline{\tilde{u}} = \begin{bmatrix} \overline{F}_h \\ \overline{F}_c \end{bmatrix},\ \mathbf{G} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}. \tag{28}$$

The matrix $\mathbf{G}$ can be obtained by considering the effects of changes in hot and cool flow rate on the level and temperature of water in the tank, respectively. With the above parameters, the matrix $\mathbf{G}$ is described as following

$$\mathbf{G} = \begin{bmatrix} \dfrac{31}{s} & \dfrac{31}{s} \\ \dfrac{4.9\times 10^5}{71.5s+1} & \dfrac{-4.9\times 10^5}{71.5s+1} \end{bmatrix} \tag{29}$$

The discretion of this model (27) with sample time $T_s = 1s$ implies the input-output mapping

$$\underline{\tilde{y}}_k = \mathbf{G}_z(z^{-1})\underline{\tilde{u}}_k \tag{30}$$

with

$$\mathbf{G}_z(z^{-1}) = \begin{bmatrix} \dfrac{31z^{-1}}{1-z^{-1}} & \dfrac{31z^{-1}}{1-z^{-1}} \\ \dfrac{6805z^{-1}}{1-0.9861z^{-1}} & \dfrac{-6805z^{-1}}{1-0.9861z^{-1}} \end{bmatrix} \tag{31}$$

where $z^{-l}\underline{\tilde{u}}_k$, $z^{-r}\underline{\tilde{y}}_k$ denote $\underline{\tilde{u}}_{k-l}$, $\underline{\tilde{y}}_{k-r}$, respectively.

3.2. Initializing AGPC for Water Tank

Rewrite input-output mapping (30) with matrix $\mathbf{G}_z(z^{-1})$ given in (31) in the standard form, which is already defined in (1)

$$\begin{aligned} & \underline{\tilde{y}}_k + A_{1,0}\underline{\tilde{y}}_{k-1} + \cdots + A_{n,0}\underline{\tilde{y}}_{k-1} + \underline{d}_k \\ & = B_{0,0}\underline{\tilde{u}}_k + B_{1,0}\underline{\tilde{u}}_{k-1} + \cdots + B_{m,0}\underline{\tilde{u}}_{k-m} \end{aligned} \tag{32}$$

then there are obtained $n = 2,\ m = 2$ and starting matrices $A_{1,0},\ A_{2,0},\ B_{0,0},\ B_{1,0},\ B_{2,0}$ as follows

$$\begin{aligned} A_{1,0} &= \begin{bmatrix} -1.9861 & 0 \\ 0 & -1.9861 \end{bmatrix} \\ A_{2,0} &= \begin{bmatrix} 0.9861 & 0 \\ 0 & 0.9861 \end{bmatrix} \end{aligned} \tag{33}$$

and

$$\begin{aligned} & B_{0,0} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}; B_{1,0} = \begin{bmatrix} 31 & 31 \\ 6805.45 & -6805.45 \end{bmatrix} \\ & B_{2,0} = \begin{bmatrix} -30.57 & -30.57 \\ -6805.45 & 6805.45 \end{bmatrix} \end{aligned} \tag{34}$$

Since (32) is the linearization model at operation point, so the predictive model (10) should be described concretely with

$$\begin{aligned}\underline{\mathbf{y}} &= \underline{\bar{\mathbf{y}}} + \underline{\tilde{\mathbf{y}}} = \underline{\bar{\mathbf{y}}} + \mathbf{Y}\underline{\tilde{\mathbf{u}}} + \underline{\mathbf{g}} \\ &= \underline{\bar{\mathbf{y}}} + \mathbf{Y}\left(\underline{\mathbf{u}} - \underline{\bar{\mathbf{u}}}\right) + \underline{\mathbf{g}}\end{aligned} \tag{35}$$

where

$$\underline{\mathbf{y}} = \begin{bmatrix} \underline{y}_k \\ \underline{y}_{k+1} \\ \vdots \\ \underline{y}_{k+N} \end{bmatrix};\ \underline{\bar{\mathbf{y}}} = \begin{bmatrix} \underline{\bar{y}} \\ \underline{\bar{y}} \\ \vdots \\ \underline{\bar{y}} \end{bmatrix};\ \underline{\bar{y}} = \begin{bmatrix} \bar{h} \\ \bar{T} \end{bmatrix};\ \underline{\tilde{\mathbf{y}}} = \begin{bmatrix} \underline{\tilde{y}}_k \\ \underline{\tilde{y}}_{k+1} \\ \vdots \\ \underline{\tilde{y}}_{k+N} \end{bmatrix}$$

$$\underline{\mathbf{u}} = \begin{bmatrix} \underline{u}_{k+N} \\ \underline{u}_{k+N-1} \\ \vdots \\ \underline{u}_k \end{bmatrix};\ \underline{\bar{\mathbf{u}}} = \begin{bmatrix} \underline{\bar{u}} \\ \underline{\bar{u}} \\ \vdots \\ \underline{\bar{u}} \end{bmatrix};\ \underline{\bar{u}} = \begin{bmatrix} \bar{\omega}_h \\ \bar{\omega}_c \end{bmatrix};\ \underline{\tilde{\mathbf{u}}} = \begin{bmatrix} \underline{\tilde{u}}_{k+N} \\ \underline{\tilde{u}}_{k+N-1} \\ \vdots \\ \underline{\tilde{u}}_k \end{bmatrix}$$

with $\underline{\bar{\mathbf{u}}}$, $\underline{\bar{\mathbf{y}}}$ are the vector of dimension $q \times (N+1)$ and $p \times (N+1)$, respectively.

The vector $\underline{\mathbf{g}}$ is defined as

$$\begin{aligned}\underline{\mathbf{g}} &= \mathbf{B}\underline{\tilde{\mathbf{u}}}_b - \mathbf{A}\underline{\tilde{\mathbf{y}}}_b - \mathbf{D}\underline{d}_k^{/} \\ &= \mathbf{B}\left(\underline{\mathbf{u}}_b - \underline{\bar{\mathbf{u}}}_1\right) - \mathbf{A}\left(\underline{\mathbf{y}}_b - \underline{\bar{\mathbf{y}}}_1\right) - \mathbf{D}\underline{d}_k^{/}\end{aligned} \tag{36}$$

where:

$$\underline{\mathbf{u}}_b = \begin{bmatrix} \underline{u}_{k-1} \\ \underline{u}_{k-2} \\ \vdots \\ \underline{u}_{k-m} \end{bmatrix};\ \underline{\bar{\mathbf{u}}}_1 = \begin{bmatrix} \underline{\bar{u}} \\ \underline{\bar{u}} \\ \vdots \\ \underline{\bar{u}} \end{bmatrix};\ \underline{\tilde{\mathbf{u}}}_b = \begin{bmatrix} \underline{\tilde{u}}_{k-1} \\ \underline{\tilde{u}}_{k-2} \\ \vdots \\ \underline{\tilde{u}}_{k-m} \end{bmatrix}$$

$$\underline{\mathbf{y}}_b = \begin{bmatrix} \underline{y}_{k-1} \\ \underline{y}_{k-2} \\ \vdots \\ \underline{y}_{k-n} \end{bmatrix};\ \underline{\bar{\mathbf{y}}}_1 = \begin{bmatrix} \underline{\bar{y}} \\ \underline{\bar{y}} \\ \vdots \\ \underline{\bar{y}} \end{bmatrix};\ \underline{\tilde{\mathbf{y}}}_b = \begin{bmatrix} \underline{\tilde{y}}_{k-1} \\ \underline{\tilde{y}}_{k-2} \\ \vdots \\ \underline{\tilde{y}}_{k-n} \end{bmatrix}$$

with $\underline{\bar{\mathbf{u}}}_1$, $\underline{\bar{\mathbf{y}}}_1$ are the vector of dimension $q \times (m+1)$ and $p \times (n+1)$, respectively.

The vector of disturbances $\underline{d}_k^{/}$ is defined as below:

$$\begin{aligned}\underline{d}_k^{/} &= \left[B_{0,0}, \ldots, B_{m,0}\right]\underline{\mathbf{u}}_b^{/} - \left[I_p, A_{1,0}, \ldots, A_{n,0}\right]\underline{\mathbf{y}}_b^{/} \\ &= \left[B_{0,0}, \ldots, B_{m,0}\right]\left(\underline{\mathbf{u}}_b^{/} - \underline{\bar{\mathbf{u}}}_2\right) \\ &\quad - \left[I_p, A_{1,0}, \ldots, A_{n,0}\right]\left(\underline{\mathbf{y}}_b^{/} - \underline{\bar{\mathbf{y}}}_2\right)\end{aligned} \tag{37}$$

where:

$$\underline{\mathbf{u}}_b^{/} = \begin{bmatrix} \underline{\mathbf{u}}_b \\ \underline{u}_{k-m-1} \end{bmatrix};\ \underline{\bar{\mathbf{u}}}_2 = \begin{bmatrix} \underline{\bar{\mathbf{u}}}_1 \\ \underline{\bar{u}} \end{bmatrix};\ \underline{\tilde{\mathbf{u}}}_b^{/} = \begin{bmatrix} \underline{\tilde{\mathbf{u}}}_b \\ \underline{\tilde{\mathbf{u}}}_{k-m-1} \end{bmatrix}$$

$$\underline{\mathbf{y}}_b^{/} = \begin{bmatrix} \underline{\mathbf{y}}_b \\ \underline{y}_{k-n-1} \end{bmatrix};\ \underline{\bar{\mathbf{y}}}_2 = \begin{bmatrix} \underline{\bar{\mathbf{y}}}_1 \\ \underline{\bar{y}} \end{bmatrix};\ \underline{\tilde{\mathbf{y}}}_b^{/} = \begin{bmatrix} \underline{\tilde{\mathbf{y}}}_b \\ \underline{\tilde{\mathbf{y}}}_{k-n-1} \end{bmatrix}$$

Correspondingly, the objective function (17) becomes

$$J_k = (\underline{\mathbf{y}} - \underline{\mathbf{r}})^T Q_k (\underline{\mathbf{y}} - \underline{\mathbf{r}}) + (\underline{\mathbf{u}} - \triangleleft\, \underline{\mathbf{u}})^T R_k (\underline{\mathbf{u}} - \triangleleft\, \underline{\mathbf{u}}) \tag{38}$$

with

$$\triangleleft\, \underline{\mathbf{u}} = \operatorname{vec}\left(\underline{u}_{k+N-1}\ ,\ \underline{u}_{k+N-2}\ ,\ \ldots\ ,\ \underline{u}_{k-1}\right).$$

4. Simulations and Discussions

As mentioned before, the control performance of the proposed AGPC will be verified through simulating and comparing with conventional PID.

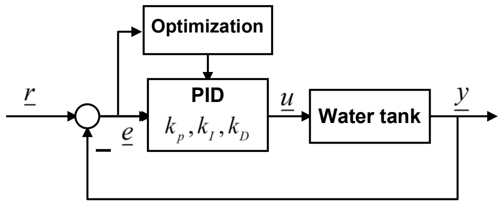

Fig. 2. PID optimizing framework.

For the scenario of control with PID, the control will be carried out in decentralized mode. In this control mode, the control variable – controlled variable pairs are selected as follows

- hot flow rate F_h & level h,
- cooling flow rate F_c & temperature T.

The decentralized controller used in the simulation is the PI controller.

$$C(s) = k_p + \frac{k_I}{s} \tag{39}$$

The PI controller parameters are selected based on minimizing the tracking errors $\underline{e} = \underline{r} - \underline{y}$ as illustrated in Fig. 2, with the cost function

$$J(\underline{p}) = \int_0^T \left|\underline{e}(\underline{p}, t)\right|^2 dt \underset{\underline{p}}{\rightarrow} \min \text{ with } \underline{p} = \begin{bmatrix} k_p \\ k_I \end{bmatrix} \tag{40}$$

Calculating parameters of the PI controller with the transfer function of the controlled variable - the control variable loop, it is obtained

- For the loop level h & hot flow rate F_h loop

$$\frac{\widetilde{H}(s)}{\widetilde{F}_h(s)} = \frac{31}{s} \Rightarrow \begin{cases} k_{p1} = 0.0052 \\ k_{I1} = 2.3432 \times 10^{-5} \end{cases}$$

- For temperature T and cooling flow rate F_c loop

$$\frac{\widetilde{T}(s)}{\widetilde{F}c(s)} = \frac{-4.9 \times 10^5}{71.5s + 1} \Rightarrow \begin{cases} k_{p2} = -2.2403 \times 10^{-5} \\ k_{i2} = -3.1540 \times 10^{-7} \end{cases}$$

By implementing AGPC controller there are many parameters to select. After some trials, the following design parameters are assigned to AGPC

– Sampling time $T_s = 1[s]$.

– Prediction horizon $N = 4$.

– Output weight

$$Q_k = \text{diag}\underbrace{\begin{bmatrix} Q & Q & \cdots & Q \end{bmatrix}}_{\text{repeat } Q \text{ for } (N+1) \text{ times}}$$

with

$$Q = \begin{pmatrix} 7\times 10^6 & 0 \\ 0 & 300 \end{pmatrix}$$

– Input weight

$$R_k = \text{diag}\underbrace{\begin{bmatrix} R & R & \ldots & R \end{bmatrix}}_{\text{repeat } R \text{ for } (N+1) \text{ times}}$$

with

$$R = \begin{pmatrix} 9\times 10^{12} & 0 \\ 0 & 8\times 10^{12} \end{pmatrix}$$

Initial parameters of plant (water tank) are

$h(0) = 0.1[m]$, $T(0) = 20[°C]$ and

$F_o(0) = 4\times 10^{-5}[m^3/s]$.

The set point signal for the level and temperature are stepwise functions.

The simulation has been carried out for two separate cases with different simulation times.

1) *Case 1*: The set points

$$h_{sp_1}(t) = \begin{cases} 0.15m & \text{khi } 0 \le t < 75s \\ 0.2m & \text{khi } 75s \le t \le 250s \end{cases}$$

$$T_{sp_1}(t) = \begin{cases} 20°C & \text{khi } 0 \le t < 75s \\ 50°C & \text{khi } 75s \le t \le 250s \end{cases}$$

are for level h and temperature T, respectively.

Obtained simulation results for this circumstance are exhibited in Fig. 3 and Fig. 4. They confirm that AGPC produces a more smooth tracking performance with smaller overshoot than the conventional PID, as already inferred theoretically before.

2) *Case 2*: The set point for output water flow is

$$F_0(t) = \begin{cases} 4\times 10^{-5}\, m^3/s & \text{khi } 0 \le t < 75s \\ 10^{-4}\, m^3/s & \text{khi } 75s \le t \le 150s \end{cases}$$

Again, all obtained simulation results, by applying both controllers PI and AGPC separately, are illustrated in Fig. 5 and Fig. 6. They authenticated the effectiveness of both controllers that system outputs had reached the set points as expected. However, the proposed AGPC produced a more precise tracking performance with shorter tracking time than by the PI, i.e., the AGPC drives system to steady state closer and faster.

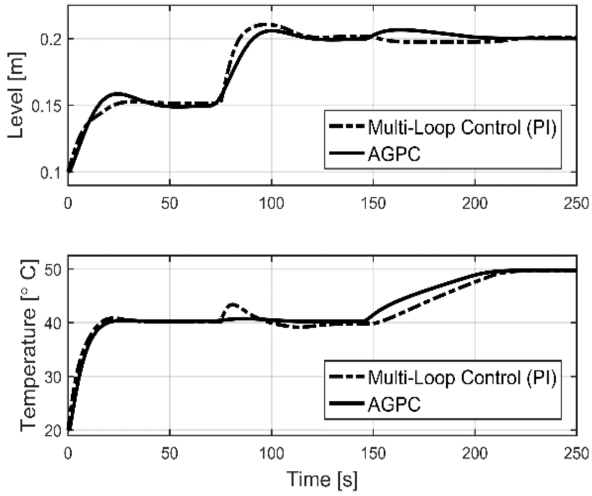

Fig. 3. Case 1 - Output signals.

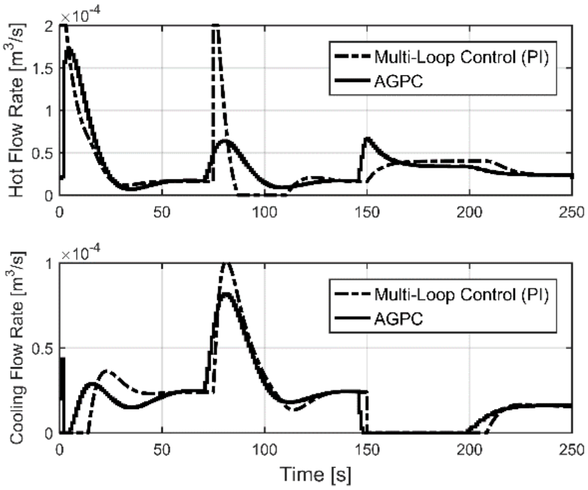

Fig. 4. Case 1 - Input signals.

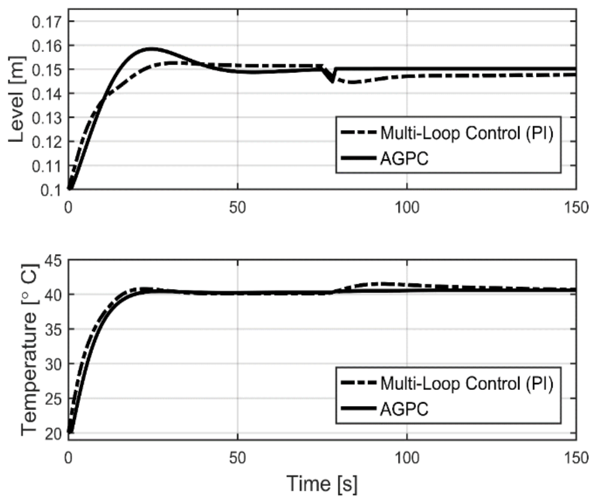

Fig. 5. Case 2 - Output signals.

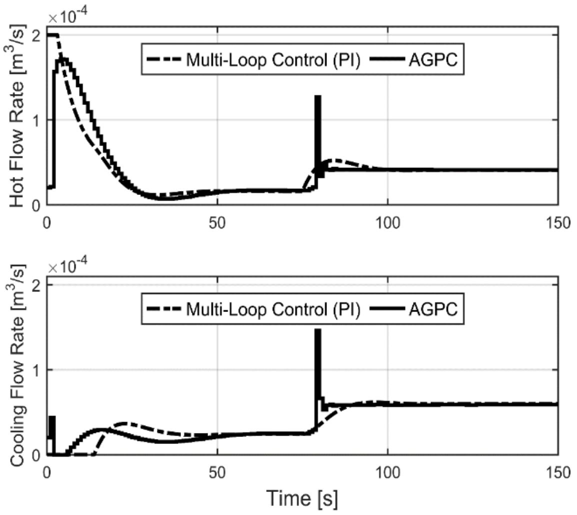

Fig. 6. Case 2 - Input signals.

Last but not least, all these in Fig. 3 to Fig. 6 displayed simulation results above with their quick reaction of proposed AGPC against system disturbances show also that this approach has the ability to anticipate future events and can take control actions suitably. PID controllers do not have this predictive ability. Furthermore, in comparison with PID control, the AGPC is more robust and be able to account for set bounded disturbance while still ensuring state constraints are met.

5. Conclusion

This paper presents the comparison between the performance of the AGPC controller and the optimal-parameters PI controller in the two case studies. In case study 1, it can be seen that the influence of the control channels on each other in use of AGPC controller is smaller than when applying conventional PI controller. In the case of study 2, the AGPC controller drives system output to steady state faster than by the PI controller when dynamics of system changes. In both cases, simulation results also authenticate that with the AGPC controller, the output responses follow the set point value with small overshooting and slightly steady state error. However, the AGPC controller includes too many parameters that need to be carefully initialized to archive the best performance.

Acknowledgments

This work was supported by the Ministry of Education and Training (MOET) under the Grant number B2021-BKA-10.

References

[1] J. M. Maciejowski, Predictive Control: With Constraints. Hoboken, NJ, USA: Prentice Hall, 2002, pp.108-144.

[2] J. Richalet, A. Rault, J. L. Testud and J. Papon, Model predictive heuristic control: Applications to industrial processes, Automatica, vol. 14, no 5, pp. 413-428, Sept.1978, https://doi.org/10.1016/0005-098(78)90001-8.

[3] A. Ramdani and S. Grouni, Dynamic matrix control and generalized predictive control, comparison study with IMC-PID, International Journal of Hydrogen Energy, vol. 42, no. 28, p. 17561-17570, July 2017, https://doi.org/10.1016/j.ijhydene.2017.04.015.

[4] J-P.Corriou, Generalized Predictive Control. In Process Control: Theory and Applications, 2nd ed., Gewerbestrasse, Cham, Switzerland: Springer International Publishing AG, 2018, pp. 611-630.

[5] C. K. Chui, G. Chen, Linear Systems and Optimal Control, Berlin, Heidelberg, Germany: Springer-Verlag Berlin Heidelberg, 1989.

[6] A. J. Krener, Adaptive horizon model predictive control, IFAC-PapersOnLine, vol. 51, no.13, pp. 31-36, Aug. 2018 https://doi.org/10.1016/j.ifacol. 2018.07.250.

[7] J.D. Hedengren, R. Asgharzadeh Shishavan, K.M. Powell and T.F. Edgar, Nonlinear modeling, estimation and predictive control in APMonitor, Computers & Chemical Engineering, vol. 70, no.5, pp.133-148, Nov. 2014, https://doi.org/10.1016/j.compchemeng.2014.04.013.

[8] S. Subramanian, Y. Abdelsalam, S. Lucia and S. Engell, Robust tube-enhanced multi-stage nmpc with stability guarantees, IEEE Control Systems Letters, vol. 6, pp.1112–1117, June 2021 https://doi.org/10.1109/LCSYS.2021.3089502.

[9] S. Subramanian, S. Lucia, R. Paulen and S. Engell, Tube-enhanced multi-stage model predictive control for flexible robust control of constrained linear systems, International Journal of Robust and Nonlinear Control, vol. 31, no 9, pp. 4458–4487, Mar. 2021, https://doi.org/10.1002/rnc.5486.

[10] R. Kamyar and E. Taheri, Aircraft optimal terrain/threat-based trajectory planning and control, Journal of Guidance, Control, and Dynamics, vol. 37, no 2, pp. 466–483, Feb. 2014 https://doi.org/10.2514/1.61339.

[11] J. G. Balchen, D. Ljungquist and S. Strand, State-space predictive control, Chemical Engineering Science, vol. 47, no. 4, pp.787-807, 1992 https://doi.org/ 10.1016/0009-2509(92)80268-H.

[12] Q-L. Su, M. W. Hermanto, R. D. Braatz and M-S. Chiu, Just-in-time-learning based extended prediction self-adaptive control for batch processes, Journal of Process Control, vol. 43, pp. 1-9, July 2016, https://doi.org/10.1016/j.jprocont.2016.04.009.