Available online at www.sciencedirect.com
ScienceDirect
IFAC PapersOnLine 50-1 (2017) 7795–7800

Data-Driven Predictive Control with Nonlinear Compensation for Performance Management in Virtualized Software System

Dharma Aryani * **Liuping Wang** ** **Tharindu Patikirikorala** ***

* *School of Electrical and Computer Engineering, RMIT University, Australia (e-mail: s3351269@student.rmit.edu.au). Department of Electrical Engineering, The State Polytechnic of Ujung Pandang, Indonesia*
** *School of Electrical and Computer Engineering, RMIT University, Australia (e-mail: liuping.wang@rmit.edu.au)*
*** *Faculty of Information and Communication Technology, Swinburne University of Technology, Australia(e-mail: tpatikirikorala@swin.edu.au)*

Abstract: This paper deals with data-driven predictive control for relative performance management in virtualized software system. The system dynamics are characterized in Hammerstein-Wiener structure to capture nonlinear and linear characteristics. The proposed control approach is the implementation of Subspace-based Predictive Control with the integration of nonlinear compensation. The compensator functions are inverse static input and output nonlinearity models from the Hammerstein-Wiener system identification. The subspace predictors are formulated from the linear model input and output of Wiener block. The experimental results from three scenarios of performance objectives show the reliability of Subspace-based Predictive Control to manage the virtualized software system.

Keywords: data-driven control, subspace predictive control, Hammerstein-Wiener, nonlinear compensation, virtualized software system

1. INTRODUCTION

Data-driven predictive control is a synthesized technique of system identification with control system design which is also called as Subspace-based Predictive Control (SPC). This approach incorporates the subspace state space estimation into Model Predictive Control (MPC) structure. Series of earlier works in this class of control and identification approach have been conducted by Di Ruscio and Foss (1998); Favoreel et al. (1998); Kadali et al. (2003); Barry and Wang (2004); Mardi and Wang (2009). The novelty of SPC is the use of subspace linear predictor to estimate the future output of the system without performing explicit parameterization of the conventional model. Subspace identification method identifies certain matrices to capture the relationship between the process inputs and outputs in non-parametric coefficients form. Then, the future output of the system is predicted by a linear function of past input, past output, and future input values. In addition, SPC is numerically robust and very attractive for on-line implementation since it uses QR-decomposition to generate the subspace coefficients directly from I/O data.

Model Predictive Control (MPC) is a model-based control algorithm which objective is to find the future control input in a finite-time prediction horizon. The control algorithm is formulated upon a numerical minimization of a cost function in a receding horizon principle. The design of MPC use the system matrices obtained from subspace system identification with a guarantee of reducing design complexity for MPC gain calculations from the real experiment data. The system matrices of dynamic model are not explicitly composed since SPC only implementing the subspace predictor variable.

In a virtualized software systems, the provider serves multiple customers by managing a single physical environment to deliver the required performance properties. The main control objective is to perform dynamic resource management where the resources can be allocated efficiently among the clients during runtime. The performance management could be carried out for absolute or relative management objective. In the case of relative scheme, the preferences consideration of performance properties and resource provisioning between the client classes lead to a severe nonlinear dynamics which can be observed at the system input and output. Some constituted factors in software systems, namely demand changes and complex preferences for performance objectives, could provoke noisy characteristics to the environment in management implementation. For noisy system, block-oriented Hammerstein-Wiener structure is considered as a favorable and generic class for dynamic model estimation (Lennart (1999)) since the approach is reliable in system identification for a process with significant nonlinearity issues. To proceed control engineering technique in virtualized software systems, the linear and nonlinear dynamics should be characterized. A useful approach to compensate the nonlinear dynamics in Hammerstein-Wiener structure is by estimating the nonlinearities in their inversion functions (Kalafatis et al. (1997)). Studies by the authors in Patikirikorala et al. (2012); Aryani et al. (2014, 2016b) exhibited the efficacy of system identification in Hammerstein-Wiener manner to identify the linear and nonlinear characteristics of shared resources environment .

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2017.08.1054

The main contribution of this paper is the implementation of control system design using Subspace Predictive Control with nonlinear compensation in Hammerstein-Wiener structure for virtualized software system. Input and output data set for this study are generated from the experimental testbed of two-clients virtualized software system. The linear model is represented by non-parametric Frequency Sampling Filter (FSF) model and the input and output nonlinearities are formulated in polynomial functions. The estimation of linear and nonlinear models parameters in Wiener block is performed in a straightforward manner. Nonlinear elements are estimated in inverse form since these functions are used as pre-input and post-output nonlinear compensators in the control system loop. This approach could reduce the impact of nonlinearities for relative performance management in virtualized software system.

Presentation in the paper is structured in seven sections. Section 2 covers the dynamic description of virtualized software system, Section 3 presents the Hammerstein Wiener system identification, then the formulation of Subspace-based Predictive Control is addressed in Section 4. The identification results and SPC design are shown in Section 5, followed by the feedback control results in section 6. Section 7 delivers conclusion based on the results from experiments in several runtime scenarios.

2. DYNAMIC SYSTEM DESCRIPTION

2.1 Virtualized software System

A real system of virtualized software environment is established using RUBiS application. It is a multi-tiers application of e-commerce website which channeling the dynamics of *ebay.com*. RUBiS has been a favourable application in the studies of software system management (eg. Patikirikorala et al. (2012)). The computing infrastructures are shared among the installed virtual machines. Three elements are operated and connected on an isolated network. A server machine, a database and a client simulator are set up to run the application. The virtualization using $Xen2.6$ hypervisor which comes with a credit-based scheduler for allocating the resources for the VMs proportionally. To support this scheduler, an actuator was installed to send the preferred ratio of resources to the system and a sensor component was added to each VM to calculate the response time of incoming requests. For this study purposes, the virtualized software system experiments are carried out in relative performance management scheme. Figure 1 shows the testbed structure.

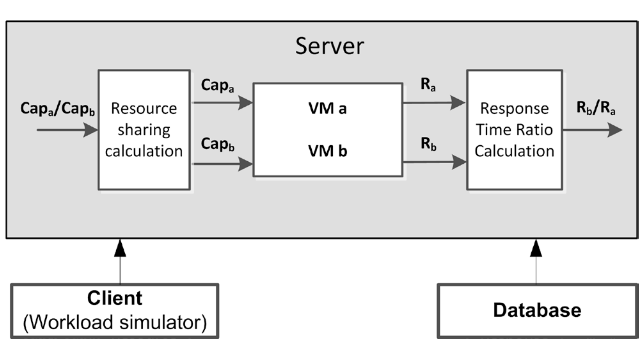


Fig. 1. Virtualized software system

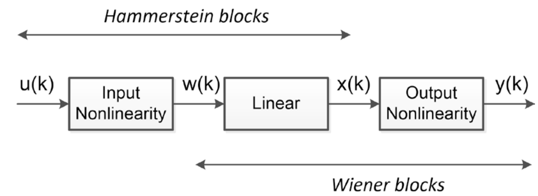


Fig. 2. Hammerstein-Wiener structure

2.2 Dynamic Nonlinearities

In relative guarantee management scheme, input and output variable setting are defined as the ratio values of the two VMs variables. $Cap_a(k)$ and $Cap_b(k)$ are the CPU allocation VM_a and VM_b respectively, where the total CPU capacity $Cap_{total} = Cap_a + Cap_b$. The input variable is $u(k) = \dfrac{Cap_a(k)}{Cap_b(k)}$. The portion of resource sharing is in the percentage of total CPU capacity where full CPU capacity equals to 100%. Therefore, to prevent a shortage of resources when workload requests suddenly increase in unpredictable condition, CPU share is constrained to a minimum capacity. In this experiment, $Cap_{a,min}(k), Cap_{b,min}(k) = 20$ and $Cap_{total} = 100$. Output variable $y(k)$ is the measured response time from each virtual machine. $RT_a(k)$, $RT_b(k)$ are response time to the workloads of VM_a and VM_b respectively. The output variable is $y(k) = \dfrac{RT_b(k)}{RT_a(k)}$.
It is clear that nonlinearities in input and output variables are caused by the ratio formulation between the VMs variable in relative scheme.

3. SYSTEM IDENTIFICATION

This section gives a summary of the Hammerstein-Wiener system identification for virtualized software system dynamics. Figure 2 shows the block structure where nonlinear memoryless blocks are sandwiched by a linear dynamic block. In the Hammerstein block, a nonlinear model is assigned to get the relationship between input signal u and intermediate input w in the form of inverse static nonlinearity function. The estimated model will be employed as compensator for the nonlinear characteristic of the input element. In Wiener block, the linear model is represented in Frequency Sampling Filters function. Moreover, the output nonlinear model is estimated in terms of inverse static nonlinearities by assigning a polynomial function as the predicted nonlinear model.

3.1 Linear model

The linear model is estimated in Frequency Sampling Filters (FSF) function. This model is used to deal with high dimensionality estimation issue when using Finite Impulse Response (FIR). FSF coefficients of the linear model are captured in frequency domain which is acquired from a linear transformation of FIR and composed of narrow bandpass filters (Wang and Cluett (1997)).

Let $f_l(k) = (\dfrac{1}{M}\dfrac{1-z^{-M}}{1-e^{j\omega_l}z^{-1}})w(k)$,

$$x(k) = \sum_{l=-\frac{m-1}{2}}^{\frac{m-1}{2}} G(e^{j\omega_l}) f_l(k) \qquad (1)$$

Equation 1 represents the j-th FSF with $\omega_l = \frac{2\pi l}{M}$ as the center frequency in $l = 0, \pm 1, \pm 2, \cdots, \pm \frac{m-1}{2}$. m is the effective order which indicates the significant parameters of FSF model and M represents the order of individual FSF filters corresponding to the process settling time $M = T_s/\delta t$ and δt is a sampling interval. m should be much smaller than M and in odd number.

3.2 Input and output nonlinearities

The estimated function of nonlinear model in Hammerstein block is formulated in an inverse relationship of input u and intermediate input w ($u(k) = f^{-1}(w(k))$. For this study, the operating points is configured to be $n = 61$ points ($u = u_1, u_2, u_3, \cdots, u_{n-1}, u_n$)

$$u = \frac{20}{80}, \frac{21}{79}, \ldots, \frac{50}{50}, \ldots, \frac{79}{21}, \frac{80}{20}$$

. Firstly, the operating points u are transformed to a range of evenly spaced operating points. The technique was formulated by defining an intermediate variable w where $w_{min} \leq w \leq w_{max}$. To have an equally spaced operating points, a fixed deviation is determined by $\delta w = \dfrac{w_{max} - w_{min}}{n-1}$.

$$w = w_1, w_2, w_3, \cdots, w_n$$

where $w_1 = w_{min}, w_2 = w_1 + \delta w, w_3 = w_2 + \delta w$ and $w_n = w_{max}$. Secondly, the respective operating points of input u_i is mapped to the w_i points. Furthermore, the relationship between data pairs from the mapping is estimated in a polynomial function $u(k) = f^{-1}(w(k))$,

$$u(k) = \alpha_0 + \alpha_1 w(k) + \alpha_2 w(k)^2 + \cdots + \alpha_a w(k)^a$$

Accordingly, the coefficients of $u(k)$ function are approximated by using least squares method, see (Aryani et al. (2016b)).

Nonlinear element in Wiener block represents the relationship of output y and intermediate output x in its inverse function $x(k) = g^{-1}(y(k))$. This function is used as a post-output nonlinear compensator in the feedback control structure. In a similar way with the pre-input compensator, the post-output compensator is estimated in a polynomial function

$$x(k) = \beta_0 + \beta_1 y(k) + \beta_2 y(k)^2 + \cdots + \beta_b y(k)^b \tag{2}$$

Furthermore, model parameters of linear and inverse static output nonlinearity models are approximated in a one-step manner by equating the $x(k)$ functions from (1) and (2). It leads to a process output function. With an assumption that inverse static function is a single-valued smooth function and $\beta_1 = 1$, the output function is formulated as follows

$$\hat{y}(k) = \sum_{l=-\frac{m-1}{2}}^{\frac{m-1}{2}} G(e^{j\omega_l}) f_l(k) w(k) - \beta_0 - \beta_2 y(k)^2 - \cdots - \beta_b y(k)^b$$

Afterwards, the model parameters of linear and inverse static output nonlinearity are obtained by implementing least squares estimation.

4. SUBSPACE-BASED PREDICTIVE CONTROL

This section provides the algorithms of Subspace-based Predictive Control. Starts with a state-space description of a linear time invariant model :

$$x_{(k+1)} = Ax_k + Bu_k + Ke_k$$

$$y_k = Cx_k + Du_k + e_k$$

where $u_k \in \Re^m$ is the input variable, $y_k \in \Re^l$ is the output, $x_k \in \Re^n$ is the state variable of the system and $e_k \in \Re^l$ is white noise disturbance.

The subspace I/O matrix equations in the field of subspace system identification are constructed by recursively substituting the system equations.

$$Y_p = \Gamma_M X_p + H_M^d U_p + H_N^s E_p \tag{3}$$

$$Y_f = \Gamma_M X_f + H_M^d U_f + H_N^s E_f \tag{4}$$

$$X_f = A^M X_p + \Delta_M^d U_p + \Delta_M^s E_p \tag{5}$$

The subscripts 'p' and 'f' denote the past and future matrices. Based on Equation (3 - 5), data matrices are calculated as shown below:

$$U_p = \begin{bmatrix} u_1 & u_2 & \cdots & u_{N-f-p+1} \\ u_2 & u_3 & \cdots & u_{N-f-p+2} \\ \vdots & \vdots & \ddots & \vdots \\ u_p & u_{p+1} & \cdots & u_{N-f} \end{bmatrix}$$

$$U_f = \begin{bmatrix} u_{p+1} & u_{p+2} & \cdots & u_{N-f+1} \\ u_{p+2} & u_{p+3} & \cdots & u_{N-f+2} \\ \vdots & \vdots & \ddots & \vdots \\ u_{p+f} & u_{p+f+1} & \cdots & u_N \end{bmatrix}$$

As the reference, these Hankel matrices have particular composition where the off-diagonal elements are similar, and the dimensions of the matrices are $Y_p, Y_f \in \Re^{MlxN-2M+1}, U_p, U_f \in \Re^{MmxN-2M+1}$ Furthermore, the state sequences are defined as

$$X_p = [x_o \; x_1 \; \cdots \; x_{j-1}]$$

$$X_f = [x_N \; x_{N+1} \; \cdots \; x_{N+j-1}]$$

By concatenating matrices Y_p and U_p in $W_p = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}$ The formulation for optimal future output prediction can be written in subspace predictor equations as follows,

$$\hat{Y}_f = L_w W_p + L_u U_f \tag{6}$$

The prediction of $\hat{Y}_f$ can be solved from its corresponding least squares problem formulation below

$$\min_{L_w, L_u} \left\| Y_f - [L_w \; L_u] \begin{bmatrix} W_p \\ U_f \end{bmatrix} \right\|_F^2 \tag{7}$$

Furthermore, the solution of minimization problem in Eq. 7 is the orthogonal projection of the row space of Y_f into the row space of matrix $\begin{bmatrix} W_p \\ U_f \end{bmatrix}$.

$$\hat{Y}_f = Y_f \Big/ \begin{bmatrix} W_p \\ U_f \end{bmatrix}$$

$$Y_f \Big/ \begin{bmatrix} W_p \\ U_f \end{bmatrix} = Y_f \begin{bmatrix} W_p \\ U_f \end{bmatrix}^{\dagger} \begin{bmatrix} W_p \\ U_f \end{bmatrix} \tag{8}$$

where $\dagger$ is Moore-Pendrose pseudoinverse. This projection can be implemented in a mathematically powerful approach by undertaking a QR-decomposition of matrix $\left[W_p^T U_f^T Y_f^T\right]^T$ to generate the lower triangular matrix R and the orthogonal matrix Q (Favoreel et al. (1998)).

$$\begin{bmatrix} W_p \\ U_f \\ Y_f \end{bmatrix} = \underbrace{\begin{bmatrix} R_{11} & 0 & 0 \\ R_{21} & R_{22} & 0 \\ R_{31} & R_{32} & R_{33} \end{bmatrix}}_{R} \underbrace{\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix}}_{Q}$$

Then, by applying the decomposed matrix above into Eq. 8, the subspace linear predictor can be calculated as follows,

$$L = [L_w \ L_u] = [R_{31} \ R_{32}] \begin{bmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{bmatrix}^{\dagger}$$

For the MPC algorithm, the subspace predictor in Eq. 6 is transformed into receding function breaking down the value of $w_p = w_p^{z^{-1}} + \Delta w_p$ and $u_f = u_f^{z^{-1}} + \Delta u_f$.

$$\hat{y}_f = L_w \Delta w_p + L_u \Delta u_f + L_w w_p^{z^{-1}} + L_u u_f^{z^{-1}} \quad (9)$$

Because the last two terms of Eq. 9 are the predictor that shifted backwards by single time step, it yields the $\hat{y}_f$ equation to be

$$\hat{y}_f = \begin{bmatrix} \hat{y}_{t+1} \\ y_{t+2} \\ \vdots \\ y_{t+f} \end{bmatrix} = L_w \Delta w_p + L_u \Delta u_f + \begin{bmatrix} y_t \\ y_t \\ \vdots \\ y_{t+f-1} \end{bmatrix}$$

Finally, the prediction of future output in Subspace-based Predictive Control approach is formulated in the function below,

$$\hat{y_f} = F_l y_t + \Gamma_l L_w \Delta w_p + \Gamma_l L_u \Delta u_f \quad (10)$$

where

$$F_l = \begin{bmatrix} I_l \\ I_l \\ \vdots \\ I_l \end{bmatrix}, \Gamma_l = \begin{bmatrix} I_l & 0 & \cdots & 0 \\ Il & I_l & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I_l & I_l & \vdots & I_l \end{bmatrix}$$

However, since the signature algorithm of MPC is that the only input values in the first N_c time steps will affect the output of the system, so the coefficient L_u is truncated to first $N_c xm$ columns before it is applied in Eq. 10. N_c is the control horizon and can be equal or less than the prediction horizon (N_p). The cost function in MPC is written in a quadratic form:

$$J = \frac{1}{2} \Delta u_{N_c}^T H \Delta u_f + \Delta u_{N_c}^T \eta \quad (11)$$

with,

$$H = (\Gamma_l L_{N_c})^T W_Q (\Gamma_l L_{N_c}) + W_R$$
$$\eta = (\Gamma_l L_{N_c})^T W_Q \Gamma_l L_w \Delta w_p + F_l (y_t - r_{t+1})$$

If there are no constraints on the system, the control signal is calculated by differentiating the cost function (Eq. 11) with regards to Δu_{N_c}. It yields to

$$\Delta u_{N_c} = -H^{-1} \eta$$
$$\Delta u_{t+1} = -K_{\Delta w_p} \Delta w_p - K_e (y_t - rt + 1)$$

where $K_{\Delta w_p}$ and K_e are the SPC control gains

$$K_{\Delta w_{p,N_c}} = ((\Gamma_l L_{N_c})^T W_Q (\Gamma_l L_{N_c}) + W_R)^{-1} (\Gamma_l L_{N_c})^T W_Q \Gamma_l L_w)$$
$$K_{e,N_c} = ((\Gamma_l L_{N_c})^T W_Q (\Gamma_l L_{N_c}) + W_R)^{-1} (\Gamma_l L_{N_c})^T W_Q F_l \quad (12)$$

Calculation of the next control input u_{t+1} only use the first value of Δu_{N_c}.

5. SYSTEM IDENTIFICATION RESULT AND SPC IMPLEMENTATION

Input and output data pairs for Hammerstein-Wiener identification are measured from the experimental test-bed in Section II.A. 400 data samples are collected from the testbed nominal workloads condition using multi-level sinusoidal signal as input variable. The input signal represents three operating conditions, $\frac{20}{80}$, $\frac{50}{50}$, and $\frac{80}{20}$ where each input level is held for 4 samples, and each client sent 100 requests/s. The input and output data are plotted in Figure 3. The evaluation of model prediction accuracy refers to the Mean Squared Error (MSE) value. MSE measures the average of squared error between the actual data and data from estimated model.

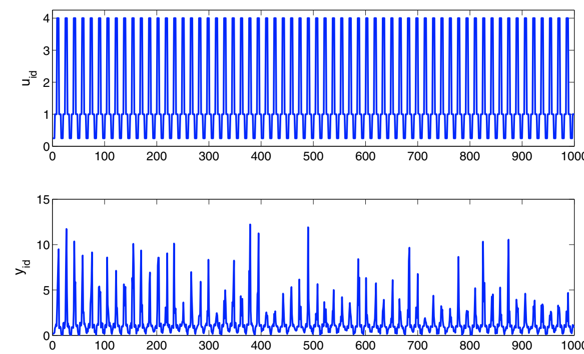


Fig. 3. Data set for system identification

5.1 Hammerstein Block

The inverse static input nonlinearity model $u(k) = f^{-1}(w(k))$ are approximated by setting $w(k)$ values with $w_{min} = -15$, $w_{max} = 15$ and $\delta w = 0.5$. Using least squares method, the polynomial function is written in (13) with MSE value 0.000068.

$$u(k) = 4.17e^{-7} w(k)^5 + 9.34e^{-6} w(k)^4 + 1.018e^{-4} w(k)^3 + 0.0028 w(k)^2 + 0.08 w(k) + 1.0045 \quad (13)$$

5.2 Wiener Block

Model parameters estimation is carried out based on least squares formulation in Section *III.C*. The initialized parameters for FSF model estimation are $M = 30$ and $m = 13$. The inverse static output nonlinearity function is in a 5^{th} order polynomial (Eq. 14) with MSE value 0.0082.

$$x(k) = 2.665e^{-4} y(k)^5 - 0.0082 y(k)^4 + 0.091 y(k)^3 - 0.4516 y(k)^2 + y(k) - 0.5956 \quad (14)$$

5.3 SPC Implementation

In the objective to reduce the impact of dynamic nonlinearities, the inverse static nonlinearity functions are set as pre-input and post-output compensators, and integrated in the control system implementation Aryani et al. (2016a). Consequently, the SPC is designed from data pairs of $w(k)$ and $x(k)$ as the input and output data, see Figure 2. Along with the nonlinear compesators, this approach is called HW-SPC where the structure is described in Figure 4. For further analysis of the SPC performance, the algorithm also implemented without nonlinear compensation which is called Linear SPC. In this approach, SPC is designed by utilizing the data set of $u(k)$ and $y(k)$. In this study, the first half of data pairs is used to calculate the subspace predictor coefficients (L_w, L_{N_c}), while the second half

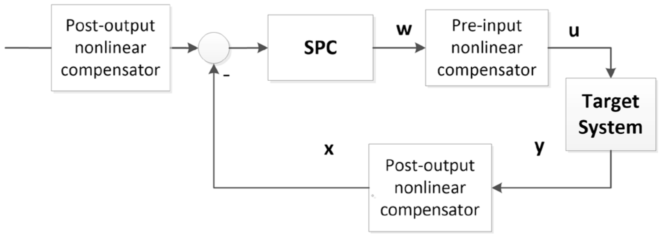


Fig. 4. Feedback control loop for HW-SPC

is used to validate the coefficients. The coefficients of subspace predictors are obtained as follows,

$$L_{N_c} = \begin{bmatrix} 0.029 & 0.002 & -0.0001 & -0.002 & -0.002 \\ 0.03 & 0.032 & 0.002 & -0.001 & -0.004 \\ 0.031 & 0.033 & 0.032 & 0.001 & -0.003 \\ 0.029 & 0.034 & 0.033 & 0.017 & -0.002 \\ 0.028 & 0.031 & 0.034 & 0.019 & 0.014 \end{bmatrix}$$

$$L_w = \begin{bmatrix} -0.07 & -0.09 & -0.01 & 0.04 & 0.21 & 0.001 & 0.001 & -0.002 & -0.001 & -0.005 \\ -0.007 & -0.21 & -0.11 & 0.04 & 0.31 & 0.001 & 0.004 & 0.001 & -0.003 & -0.007 \\ 0.02 & -0.15 & -0.22 & -0.05 & 0.34 & -0.001 & 0.003 & 0.005 & 0.0001 & -0.010 \\ 0.12 & -0.15 & -0.18 & -0.16 & 0.25 & -0.01 & 0.002 & 0.005 & 0.004 & -0.007 \\ 0.21 & -0.06 & -0.18 & -0.12 & 0.13 & -0.02 & -0.01 & 0.002 & 0.004 & -0.003 \end{bmatrix}$$

and the SPC gains are,

$$K_e = \begin{bmatrix} 0.933388 \end{bmatrix}$$

$$K_{\Delta w_p} = \begin{bmatrix} -0.005 & 0.009 & 0.001 & -0.02 & 0.11 & 0.05 & -0.02 & -0.11 & 0.031 & 0.03 \end{bmatrix}$$

Initially, subspace linear predictor is calculated using the $w(k)$ and $x(k)$ data from HW system identification. The first 500 data are used to calculate the predictor coefficients and the rest 500 to validate the prediction. The data fitting is plotted in Fig. 5 with MSE value 0.06. It can be confirmed that subspace predictors perform good data estimation. Accordingly, the HW-SPC is formulated and incorporated with the nonlinear compensators to form a feedback control.

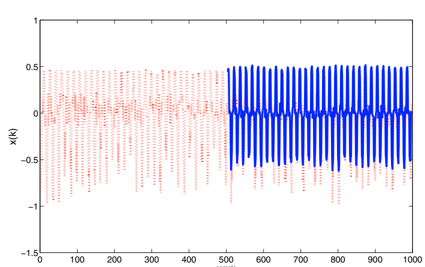


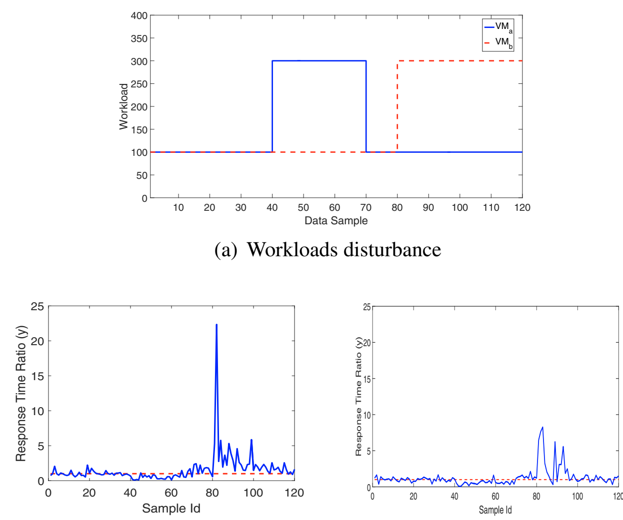
Fig. 5. Validation of subspace predictor, real (–), estimation (solid)

6. FEEDBACK CONTROL RESULTS

In feedback control loop, the target system is a virtualized software system which requires optimization for performance management and resource provisioning. Controller gains are calculated using Equation (12). The feedback control performances are evaluated in three experiments. The investigations are carried out in the testbed runtime mode with different performance differentiation ratios. The disturbances occur from the workload changes of the VMs. The order of past and future matrices are set to $p = f = 5$, and the control horizon is $N_c = 5$.

6.1 Experiment A. Performance Differentiation Ratio = 1

In this experiment, both of the VMs are equally important which means that the level of priority for the VMs to access the resources are similar. Thus, the reference value is set to 1. During runtime, the workloads are applied with similar capacity for both VMs before they are changed alternatively to the higher capacity (see Fig. 6). The objective of this experiment is to investigate system performance when the resource demands suddenly increased away from the nominal workload. The output signals show that the performance of HW SPC is more satisfactory than Linear SPC considering the steady-state behaviour of the system response. The Linear SPC shows significant oscillation with a high steady state error after the workload drastically increased, while the HW SPC provide a better steady state performance.



(a) Workloads disturbance

(b) Linear SPC (c) HW SPC

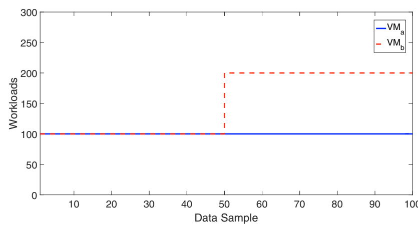
Fig. 6. Experiment A. Performance Differentiation Ratio = 1

6.2 Experiment B. Performance Differentiation Ratio > 1

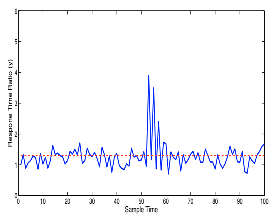
In this case, VM_a is more important than VM_b and the reference point is set to 1.3. This setting leads the control system to allocate fewer resources to VM_b as the less important VM. The workloads for both VMs are at the nominal capacities in the first half of the experiment then workload for VM_b is increased for the rest of experiment. This is the scenario when VM_b has fewer resources but its workloads are much higher than VM_a. The workload changes and output responses from the experiment are presented in Figure 7. The MSE values of Linear SPC and HW SPC are 0.1928 and 0.0882, respectively. The control output signals for this case show that HW SPC performance management outperforms the Linear SPC.

6.3 Experiment C. Performance Differentiation Ratio < 1

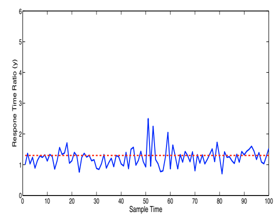
This experiment evaluates the control performance when VM_b is more important than VM_a. The reference point is 0.7 and the workload of VM_a is increased in the middle of runtime. The output response in Fig. 8 reveals that the Linear SPC has stability issues when disturbance are given to the system. The MSE values of Linear SPC and HW SPC are 0.1307 and 0.0584, respectively. The output response from all experiments indicate that the integration of nonlinear compensator in feedback control loop provides significantly better system stability to deal with performance differentiation ratio setting and the



(a) Workloads disturbance

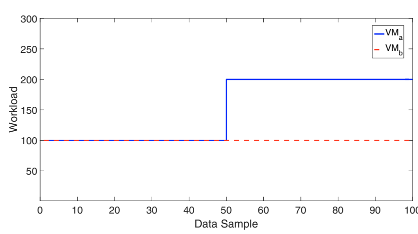


(b) Linear SPC

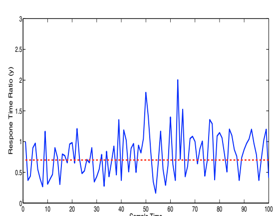


(c) HW SPC

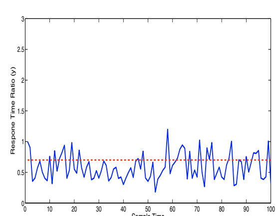
Fig. 7. Experiment B. Performance Differentiation Ratio > 1



(a) Workloads disturbance



(b) Linear SPC



(c) HW SPC

Fig. 8. Experiment C. Performance Differentiation Ratio < 1

workloads disturbance. This control approach can be implemented in other scenarios of shared resources environment with more virtual machines in the system. An extended amount of virtual machines will lead to a more complex differentiation ratio of performance objectives.

7. CONCLUSION

Data-driven predictive control has been implemented in virtualized software system with an advancement of nonlinearities compensation from the Hammerstein-Wiener system identification. The compensator reduces the impact of nonlinearities in relative performance management scheme. The output response evaluations from different scenarios have proven that HW SPC approach provides better performance management and disturbance rejections capability compared to Linear SPC. It can be concluded that the proposed approach contributes to the performance stability of a data-driven control class in application to virtualized software system.

REFERENCES

Aryani, D., Wang, L., and Patikirikorala, T. (2016a). Block-oriented nonlinear model based control for relative performance and resource management in virtualized software system. In *2016 Australian Control Conference (AuCC)*, 71–76. doi:10.1109/AUCC.2016.7868005.

Aryani, D., Wang, L., and Patikirikorala, T. (2014). Control oriented system identification for performance management in virtualized software system. *IFAC Proceedings Volumes*, 47(3), 4122–4127.

Aryani, D., Wang, L., and Patikirikorala, T. (2016b). On identification of hammerstein and wiener model with application to virtualised software system. *International Journal of Systems Science*, 48(6), 1146–1161.

Barry, T. and Wang, L. (2004). A model-free predictive controller with laguerre polynomials. In *Control Conference, 2004. 5th Asian*, volume 1, 177–184. IEEE.

Di Ruscio, D. and Foss, B. (1998). On state space model based predictive control. *def*, 2, 6664.

Favoreel, W., De Moor, B., and Gevers, M. (1998). Spc: Subspace predictive control. In *Proceedings of the IFAC World Congress, Beijing, China (July 1999)*. Citeseer.

Kadali, R., Huang, B., and Rossiter, A. (2003). A data driven subspace approach to predictive controller design. *Control engineering practice*, 11(3), 261–278.

Kalafatis, A., Wang, L., and Cluett, W. (1997). Identification of wiener-type nonlinear systems in a noisy environment. *International Journal of Control*, 66(6), 923–941.

Lennart, L. (1999). System identification: theory for the user. *PTR Prentice Hall, Upper Saddle River, NJ*, 1–14.

Mardi, N.A. and Wang, L. (2009). Subspace-based model predictive control of time-varying systems. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, 4005–4010. IEEE.

Patikirikorala, T., Wang, L., Colman, A., and Han, J. (2012). Hammerstein–wiener nonlinear model based predictive control for relative qos performance and resource management of software systems. *Control Engineering Practice*, 20(1), 49–61.

Wang, L. and Cluett, W.R. (1997). Frequency-sampling filters: an improved model structure for step-response identification. *Automatica*, 33(5), 939–944.