RESEARCH ON VISUAL MODELING AND SIMULATION METHOD FOR OPERATIONAL FLOWSHEETS-ORIENTED CANDU CONTROL SYSTEMS

Z. Cui[1], Robyn Prime[2], K. Scott[1] and M. McInTyre[1]
[1] Atlantic Nuclear Services Ltd, Fredericton, New Brunswick, Canada
[2] Point Lepreau Generating Station, New Brunswick, Canada

Abstract

To flexibly model and accurately simulate CANDU control systems that are described by variety of operational flowsheets, a general visual modeling and simulation method has been researched, in which discrete-similarity algorithm, database and visually graphical modeling technologies are employed. And Parallel Virtual Machine (PVM) is adopted for interface to stand-alone simulation software of controlled objects. Test cases demonstrate that the presented method is effective and reliable, and the modeling and simulation program based on such method is generally applicable for all kinds of control systems in CANDU-type plants.

1. Introduction

Computer simulation is playing a very important role in control system design and analysis of nuclear power stations. Therefore some very useful and station or reactor-type specific programs have been developed, for example, some programs specific for CANDU 6 control system simulation. However, these programs are not flexible enough to deal with operational flowsheet-oriented control system modelling and simulation. When doing control system design and analysis by using such programs, what users have to prepare are complicated and voluminous input data files, hence it is very hard to visually connect the simulation models with the real control systems. Moreover, this traditional simulation method based on input data files has the following major shortages: (1) It is laborious and mistakable to convert the operational flowsheets into input data files in specific formats, especially for large-scale and complex control systems; (2) It is troublesome to find the corresponding data in the input data files and make modification when the parameters in the control systems are changed; (3) Without modifying the source codes, it is difficult or even impossible to setup new models and do the relevant simulations when configuration of the simulated control system is changed due to design update or additional simulation necessity, for example, control elements or input/output signals in the modelled operational flowsheet are changed.

To overcome the shortages of the modelling and simulation program developed by using traditional methodology, a general visual modelling and simulation methodology has been researched, in which discrete-similarity algorithm, database and visually graphical modeling technologies are employed. The modelling and simulation program based on the new methodology can flexibly model and accurately simulate CANDU control systems that are described by variety of operational flowsheets. And Parallel Virtual Machine (PVM) is adopted for interface to stand-alone simulation software of controlled objects. Test cases demonstrate that the presented methodology is effective and reliable, and the modeling and simulation program based on such method is generally applicable for all kinds of control systems in CANDU-type plants.

2. Operational flowsheet-oriented visual modelling

Operational flowsheet is graphic description for functions of each element and flow of each signal in the control system. Whatever the control system is complicated, its operational flowsheet consists of the below basic elements: input/output signal, linear element, non-linear element, computational element and logic element. These elements are connected by signal chains.

Typical linear elements include proportional controller [P] (k), proportional-integral [PI] controller (k_1+k_2/s), proportional-derivative controller [PD] (k_1+k_2s), proportional-integral-derivative controller [PID] $k(1+\frac{1}{\tau_I s})(\frac{1+\tau_D s}{1+\tau_D s/\lambda})$, lag element e^{-Ts} (expanded as $\frac{sT_1+1}{sT_2+1}$) and oscillator $\frac{k}{T^2s^2+2\rho Ts+1}$ (expanded as $\frac{k}{(T_1s+1)(T_2s+1)}$) and so on. Since transfer functions are capable of direct multiplication and addition, all linear elements can be treated as the general equation $\frac{c+ds}{a+bs}$. For instance, when a is zero, $\frac{c+ds}{a+bs}$ expresses the PI controller.

In operational flowsheets of CANDU control systems, there are usually many non-linear elements, for example RRS adjuster speed and on-off heater controls for heat transport system pressure in Figure 1. These non-linear elements can be categorized into two basic groups, one for general non-linear elements, the other for specific non-linear elements. The former can be represented by a set of independent and dependent variables. The later has to be expressed by a set of specific data and operation rules.

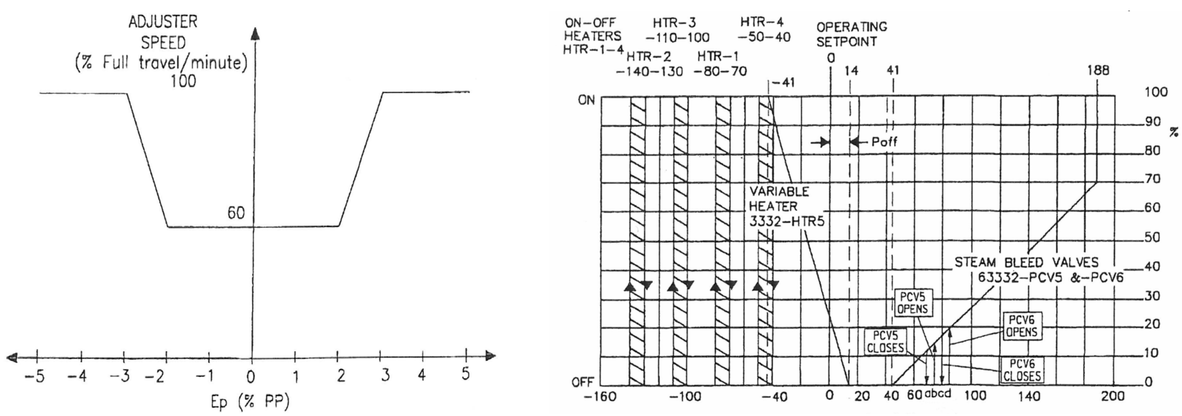

Figure 1 Characteristics of RRS adjuster speed and HTS pressure control

Operation elements are important ones in the operational flowsheets, which include addition, subtraction, multiplication, division, absolute value, reciprocal, average, maximum, minimum and square root, etc. Except for the common logic AND, OR and NO, there exist particular logic operation elements in the flowsheets of CANDU control systems. Shown in Figure 2 are two examples, RRS adjuster in/out criteria and load limiter in turbine control. Flexible modelling for logic operations in the operational flowsheets could be the most challenging aspect in visual modelling and simulation of CANDU control systems. Complex logic operations can be modelled by first being separated into simple ones and then being combined.

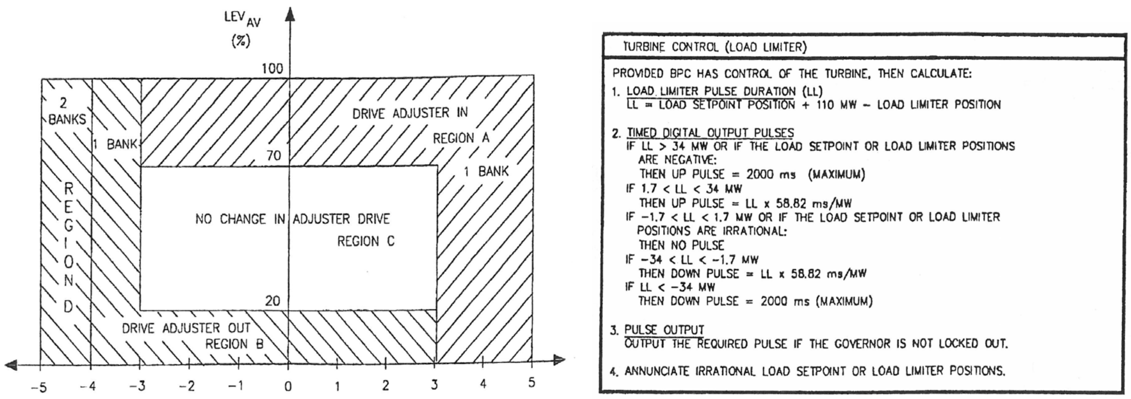

Figure 2 RRS adjuster in/out criteria and load limiter in turbine control

Based on the above analysis on CANDU control system operational flowsheets, the methodology of general visual modelling is described as follows: In Windows graphical mode, the basic elements of the control system are inserted by selecting the element through the element toolbar, then edit or change the parameters related to the selected elements, then connect the elements by using the signal chain to generate a control system or parts of a control system, finally modify the control system modelled in the graphic mode to make it consistent with the operational flowsheet to be simulated, by using the edit functions such as addition, deletion, copy and paste. From the point of visual modelling, this methodology is similar to that of SimuLink in MATLAB, or more popularly that of AutoCAD. But this methodology has more specific basic elements for CANDU control system modelling, moreover the effective simulation algorithm and PVM interface presented in Section 3.

Sometimes, it is more practical to use more than one diagram to express a complex and large-scale control system. In this case, combination of the modelled operational flowsheets is completed through a workspace called project, of which the method is described in Reference [1].

By employing this visual modeling method, users do not have to prepare input data files anymore. That is, what seen on the screen is what to obtain. It becomes very easy to setup the simulation models for CANDU control systems.

3. Simulation algorithms

Simulation algorithm for non-linear elements is very simple, that is, direct calculation based on the characteristics of the elements. However, non-linear elements in the control system operational flowsheets make it more difficult to solve the linear elements in the systems, since the linear input (i.e. output of the non-linear elements) could experience abrupt and large numerical value variance. In this situation, discrete-similarity algorithm is much more effective than general numerical integral algorithm.

Presented below is the brief description of the discrete-similarity algorithm and its corresponding treating of basic linear elements in the control system. Assume state equation of continuous systems to be

$$\dot{X}(t) = AX(t) + BU(t) \tag{1}$$

Then its time domain solution is

$$X(t) = e^{At} X(0) + \int_o^t e^{A(t-\tau)} BU(\tau) d\tau \tag{2}$$

At time t=kT and t=(k+1)*T, the above solution can be written as

$$X(kT) = e^{AkT} X(0) + \int_o^{kT} e^{A(kT-\tau)} BU(\tau) d\tau \tag{3}$$

$$X[(k+1)T] = e^{A(k+1)T} X(0) + \int_o^{(k+1)T} e^{A[(k+1)T-\tau]} BU(\tau) d\tau \tag{4}$$

Subtract equation (3) multiplied by e^{-AT} from equation (4), get

$$X[(k+1)T] = e^{AT} X[kT] + \int_{kT}^{(k+1)T} e^{A[(k+1)T-\tau]} BU(\tau) d\tau \tag{5}$$

Where, T is sampling period and k is the number of T. In fact, the integral result of equation (5) is independent of k, thus the equation can be re-written as

$$X[(k+1)T] = e^{AT} X(kT) + \int_o^{T} e^{A[T-\tau]} BU(\tau) d\tau \tag{6}$$

Equation (6) is the discrete format of equation (1) in which U(τ) is the input function. If the sampling holder is one-order, then input function U(τ) varies linearly between two neighbouring sampling times, therefore [2,3]

$$U(\tau) = U(kT) + \frac{U[(k+1)T - U(kT)]}{T} \tau \tag{7}$$

That is,

$$U(\tau) = U(kT) + \dot{U}(kT)\tau \tag{8}$$

Introduce equation (8) into equation (6), have

$$X[(k+1)T] = e^{AT} X(kT) + \left[\int_0^T e^{A(T-\tau)} B d\tau\right] U(kT) + \left[\int_0^T e^{A(T-\tau)} B \tau d\tau\right] \dot{U}(kT) \tag{9}$$

Given

$$\phi(T) = e^{AT} \tag{10}$$

$$\phi_m(T) = \int_0^T e^{A(T-\tau)} B d\tau \tag{11}$$

$$\phi_p(T) = \int_0^T e^{A(T-\tau)} B \tau d\tau \tag{12}$$

Then, equation (9) can be expressed as

$$X[(k+1)T] = \phi(T)X(kT) + \phi_m(T)U(kT) + \phi_p(T)\dot{U}(kT) \tag{13}$$

Further, the above equation can be re-written in recursive format as follows

$$X_{n+1} = \phi X_n + \phi_m U_n + \phi_p \dot{U}_n \tag{14}$$

Since the coefficients A and B in equation (1) are known, coefficients $\phi(T)$, $\phi_m(T)$ and $\phi_p(T)$ can be calculated by using equations (10) to (12). When sampling holder is zero order, $\phi_p(T)=0$. Besides, if sampling period is constant, the values of coefficients $\phi(T)$, $\phi_m(T)$ and $\phi_p(T)$ do not change. Thus no repeat calculations of coefficients $\phi(T)$, $\phi_m(T)$ and $\phi_p(T)$ are needed during transient simulation, hence the simulation results can be obtained by using the recursive algorithm in equation (14). Compared with traditional numerical integral algorithms such as Runge-Kutta method, the discrete-similarity algorithm described above can significant reduce calculation burden, increase simulation speed, and improve numerical calculation stability.

As per the discrete-similarity algorithm, the linear elements in the control systems can be discretized one by one to obtain the coefficients $\phi(T)$, $\phi_m(T)$ and $\phi_p(T)$, then calculate the state variables by using equation (14) and finally get the output of the linear element. Here is description for discretization of a linear element that has the transfer function of $G(s) = \frac{c+ds}{a+bs}$, where a, b, c, and d are non-zeros.

$$G(s) = \frac{c+ds}{a+bs} = \frac{d}{b} + \frac{d}{b}\,\frac{\frac{c}{d} - \frac{a}{b}}{\frac{a}{b} + s} \tag{15}$$

Its time domain state equation and output equation are

$$\begin{cases} \dot{X}(t) = -\frac{a}{b}X(t) + \frac{d}{b}U(t) \\ Y(t) = \left(\frac{c}{d} - \frac{a}{b}\right)X(t) + \frac{d}{b}U(t) \end{cases} \tag{16}$$

In comparison with continuous state equation (1), have $A = -\frac{a}{b}$ $\mathrm{B} = \frac{\mathrm{d}}{\mathrm{b}}$, and according to equations (10) to (12), obtain

$$\phi(T) = e^{-\left(\frac{a}{b}\right)T} \tag{17}$$

$$\phi_m(T) = \frac{d}{a}\left(1 - e^{-\left(\frac{a}{b}T\right)}\right) \tag{18}$$

$$\phi_p(T) = \frac{d}{a}T - \frac{bd}{a^2}\left(1 - e^{-\left(\frac{a}{b}T\right)}\right) \tag{19}$$

Therefore, the discrete equation of the transfer function $G(s) = \dfrac{c + ds}{a + bs}$ is

$$\begin{cases} X_{n+1} = \phi(T)X_n + \phi_m(T)U_n + \phi_p(T)\dot{U}_n \\ Y_{n+1} = \left(\dfrac{c}{d} - \dfrac{a}{b}\right)X_{n+1} + \dfrac{d}{b}U_n + \dfrac{d}{b}T\dot{U}_n \end{cases} \tag{20}$$

Given $\psi(T) = \dfrac{c}{d} - \dfrac{a}{b}$, $\psi_{\mathrm{m}}(T) = \dfrac{d}{b}$, $\psi_{\mathrm{p}}(T) = \dfrac{d}{b}T$, then equation (20) can be re-written as

$$\begin{cases} X_{n+1} = \phi(T)X_n + \phi_m(T) + \phi_p(T)\dot{U}_n \\ Y_{n+1} = \psi(T)X_{n+1} + \psi(T)U_n + \psi_p(T)\dot{U}_n \end{cases} \tag{21}$$

For different a, b, c and d values in the transfer function $G(s) = \dfrac{c + ds}{a + bs}$, the calculations of $\phi(T)$, $\phi_m(T)$, $\phi_p(T)$, $\Psi(T)$, $\Psi_m(T)$ and $\Psi_p(T)$ are listed in Table 1. Based on these calculated coefficients, the state variable and output of the linear elements at different sampling times can be obtained by equation (21).

Table 1 $\phi(T)$, $\phi_m(T)$, $\phi_p(T)$, $\Psi(T)$, $\Psi_m(T)$ and $\Psi_p(T)$ for transfer function $G(s) = \dfrac{c + ds}{a + bs}$,

Element Characteristic	a	b	c	d	Transfer Function	$\phi(T)$	$\phi_m(T)$	$\phi_p(T)$	$\psi(T)$	$\psi_m(T)$	$\psi_p(T)$
Proportional (P)		0		0	$\frac{c}{a}$	0	0	0	0	$\frac{c}{a}$	0
Integral (I)	0			0	$\frac{c}{bs}$	1	$\frac{c}{b}T$	$\frac{c}{2b}T^2$	1	0	0
PI	0				$\frac{c+ds}{bs}$	1	$\frac{c}{b}T$	$\frac{c}{2b}T^2$	1	$\frac{d}{b}$	$\frac{d}{b}T$
Inertia				0	$\frac{c}{a+bs}$	$e^{-\frac{a}{b}T}$	$\frac{c}{a}\left(1-e^{-\frac{a}{b}T}\right)$	$\frac{c}{a}T-\frac{bc}{a^2}\left(1-e^{-\frac{a}{b}T}\right)$	1	0	0
Lead-lag					$\frac{c+ds}{a+bs}$	$e^{-\frac{a}{b}T}$	$\frac{d}{a}\left(1-e^{-\frac{a}{b}T}\right)$	$\frac{d}{a}T-\frac{bd}{a^2}\left(1-e^{-\frac{a}{b}T}\right)$	$\frac{c}{d}-\frac{a}{b}$	$\frac{d}{b}$	$\frac{d}{b}T$

In this way, the operational flowsheet-oriented transient simulation can be performed element by element by taking linear element $G(s) = \dfrac{c + ds}{a + bs}$ as the basic simulation unit. The simulation process can be briefed below:

(a) If input of the current linear element to be calculated is from non-linear, calculation or logic elements, continue to search the upstream signals against the signal flow direction, until another linear element or external input signals are met.

(b) Starting from the output of the met linear element or external input signals, calculate outputs of each elements in the same direction as the signal flow direction, until the current linear element is reached.

(c) Calculate input change rate of the current linear element as per the new value and the value at previous sampling time.

(d) Based on the obtained inputs and input change rate, calculate the state variable values and output of the current linear element.

(e) Repeat (a) to (d) for each linear element in operational flowsheets to perform the simulation for all control systems modelled through the visual modelling method presented in Section 2.

Data regarding elements and signal chains are stored in a database. Thus database search technique is used in the above process to find out the relevant elements.

Simulation of the controlled objects such as reactor, pressurizer and boilers in the CANDU systems can be carried out by specific system analysis codes, for example CATHENA. The data interface between the control system simulation program and the system analysis codes is set up by using PVM (Parallel Virtual Machine) technology [4].

4. Applications and conclusions

Except for the special non-linear elements and logic elements in CANDU control system operational flowsheets, the presented visual modeling methodology and simulation algorithms have been successfully applied in modeling and simulation for overall control systems of pressurized water reactors (PWR) [5], which are described by control system design block diagrams. Applications in CANDU control systems in form of operational flowsheets have been tested for some specific cases, for instance reactor control systems. Test cases demonstrate that the presented method is effective and reliable to set up the required simulation models. Development of visual modeling and simulation program for overall CANDU control systems is in process and expected to be done soon. There is no doubt that developed program based on the presented methodologies is generally applicable for all kinds of control systems in CANDU-type plants. The benefits of the visual modelling and simulation method presented are ease of simulation model setup and maintenance, control system data encapsulation and modification, configuration management, as well as signal traceability.

5. References

[1] Z. Cui, M. McIntyre, et al, "VIMSCAN – Visual Modeling Software for Assembling Nodalization Networks of Thermalhydraulics Analysis Codes", 28th Annual Conference of the Canadian Nuclear Society, St John, 2007.

[2] Q. Gu, System Design and Simulation (Chinese), Tsinghua University Press, Beijing, 1995.

[3] G. Xiang, Control System Simulation and Modelling (Chinese), Science Press, Beijing, 1993.

[4] Al Geist, Adam Beguelin, Jack Dongarra, et al, "PVM: Parallel Virtual Machine A Users' Guide and Tutorial for Networked Parallel Computing", the MIT Press Cambridge, Massachusetts London, England, 1994. Electronic version derived on January 15, 2007 from website http://www.netlib.org/pvm3/book/pvm-book.html.

[5] Z. Cui, NCS-A Software for Visual Modeling and Simulating of PWR Nuclear Power Control System, China Nuclear Science and Technology Report, China Nuclear Information Centre, Atomic Energy Press, CNIC-01308, TSHU-0084, Dec. 1998.

Other references used in the paper are unpublished research reports.