Discrete State Simulation of Power System Dynamics

Lan Tang, and Hongchun Shu

***Abstract*--Conventional power system dynamic simulation is based on discretization of time. When distributed simulator is implemented, inherent continuity of time requires frequent communication among different solvers. Given the fact that state variables of system are usually loosely coupled, even independent, this paper introduces discrete state method into power system simulation, and presents how to imitate power system dynamics by discrete event simulation. The feasibility of the method is proved by a simple case. The potential advantages of the method are also summarized.**

***Index Terms*--Discrete state simulation，DEVS, Power system dynamics, Power system simulation.**

I. Introduction

MODELING and simulation are two important steps of power system analysis, design, and research. At the stage of modeling, dynamic elements in power system are generally described as a set of differential equations. In the simulation stage, the core of conventional power system dynamic simulation is the solution of differential equations with adaptive numerical integration methods. From the view of the system theory, each dynamic element is a continuous system. Simulation of continuous systems on digital computers requires discretization. There are two system aspects that can be made discrete; time and state.

Classical methods as Euler, Runge-Kutta, Adams, etc., are based on discretization of time, and the solution is calculated at fixed points in time. The essential feature of a discrete time approximation is that the resulting difference equations map a discrete time set to a continuous state set. With the expansion of modern interconnected power system, the orders of differential equations describing dynamic elements become very huge. Therefore many parallel and distributed algorithms are involved in this domain to improve the efficiency of solution. Inherent continuity of time requires frequent communication among different solvers. On the contrary, the relationship among majority of state variables in a system is loosely coupled, even mutually independent. Therefore, to discretize the state variables during the course of simulation would decrease the communication loads of parallel and distributed algorithms. Moreover, many attractive properties of the discrete state simulation have also been proved [1].

The results of discrete time simulation methods are trajectories of different power system state variables. Thus, the dynamic performance of power system can be judged by these trajectories. For example, the swing curves of synchronous generators are used to analyze the transient stability of a power system. That is to say, the changes of state determine the performance of power system. It proved that sudden transform of state variables is the root of blackout. Therefore, to discretize the state variables during the course of simulation will be a more direct way. The event scheduling provides a natural way to think about the evolution of the system.

When the states are discrete, the differential equations are approximated by a discrete event system. The method of using discrete event system to study simulation of continuous system has achieved tremendous progress since 70th last century. Based on the fruits mentioned above, this paper presents how to imitate power system dynamics by discrete event simulation, and a simple case is used to prove the feasibility of the method.

II. Discrete Event Simulation of Continuous System

A. A Brief Review of DEVS

The Discrete Event System Specification (DEVS) formalism was conceived by Zeigler, to provide a means of specifying a mathematical object called a system [2]. Basically, a system has a time base, inputs, states, and outputs, and functions for determining next states and outputs given current states and inputs. There are atomic models and coupled models in DEVS formalism.

A DEVS atomic model is defined by the following structure :

$$M = < X, Y, S, \delta_{\text{int}}, \delta_{ext}, \lambda, ta >$$

where:

X is the set of input values,

Y is the set of output values,

S is a set of states,

$\delta_{\text{int}}: S \rightarrow S$ is the internal state transition function,

$\delta_{ext}: Q \times X^b \rightarrow S$ is the external state transition function with $Q = \{(s,e) | s \in S, 0 \le e \le ta(s)\}$ and X^b denotes the collection of bags over X (sets in

This work was supported in part by National Natural Science Foundation of China(50347026,50467002).

L. Tang is with Department of Electrical Engineering, Harbin Institute of Technology, Harbin, Heilongjiang, P.R.China (e-mail:larry_km@163.com).

H. C. Shu is with the Department of Electrical Engineering, Kunming University of Science and Technology, Kunming, Yunna, P.R.China (e-mail: kmshc@sina.com).

which some elements may occur more than once),

$\lambda: S \rightarrow Y^b$ is the output function, and

$ta: S \rightarrow R^+_{0,\infty}$ is the time advance function.

The atomic models may be coupled in DEVS formalism to form a coupled model. The coupled model provides the hierarchical and modular structure necessary to describe system networks. It is formally defined by:

$$CM = < X, Y, D, \{M_i\}, \{I_i\}, \{Z_{ij}\}, select >$$

where:

D: is a set of component names; for each i in D,

M_i: is a basic DEVS component; for each i in D,

I_i: is a set, the influencees of i and for each j in I_i,

Z_{ij}: is the i -to- j output-input translation function

$select$: is a function, the tie-breaking selector.

The coupled models can be used as the components of more complicated models. Therefore, DEVS formalism provided a mechanism to model complex system with hierarchical construction.

B. State Quantization-Based Methods

Continuous time ODE systems with initial conditions have traditionally been simulated by discretizing the time domain, and solving the ODE over each discrete time interval. Quantized DEVS models [3] permitted to solve this problem using a different approach. Instead of advancing time in discrete steps, we can advance time based on discrete events, where an event is a significant change in an input, state or output variable. The key to handling the continuous variables is to determine the time when a significant event occurs in a component. A significant event detector, called a quantizer, monitors its input and uses a logical condition to decide when a significant change, such as crossing a threshold has occurred. The concept of quantization formalizes the operation of significant event detectors. A quantum is the measure of how big a change must be considered significant. Thus quantization of state variables is a method to obtain a discrete event approximation of a continuous system.

So far there are different quantizers presented. Among them Kofman's quantization function with hysteresis [1] shows more attractive properties. Figure 1 shows a typical quantization function $q(t)$ with uniform quantization intervals, obtained with a hysteresis window.

The Quantized State System (QSS) method can be defined as follow [4].

Given a time invariant states equation system:

$$\dot{x}(t) = f[x(t), u(t)] \quad (1)$$

where $x \in R^n, u \in R^m$ and $f: R^n \rightarrow R^n$, the QSS-method approximates it by a system:

$$\dot{x}(t) = f[q(t), u(t)] \quad (2)$$

where $q(t)$ and $x(t)$ are related components by hysteretic quantization functions (i.e. each quantized variable $q_i(t)$ is related to the corresponding state variable $x_i(t)$ by a hysteretic quantization function).

The resulting system (2) is called Quantized State System.

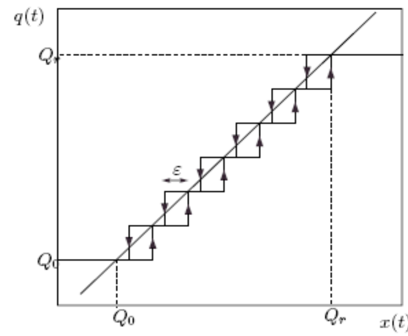

Fig. 1. Quantized function with hysteresis

The Quantized State System can be represented by a coupled DEVS model. So the behavior of a continuous system can be simulated by a discrete event system.

Recent years, quantization-based method has been used to simulate hybrid system [5]. The method itself has been studied more deeply. Some theory issues for discrete event method for simulating continuous systems have also been studied [6].

From the view of Modeling and Simulation (M&S), the DEVS formalism offers a new way to unify the computational representation of both continuous and discrete phenomena and simulate them with the greater efficiency and flexibility afforded by object-oriented discrete event environments.

III. Discrete State Model of Swing Equtions

Synchronous generator is the important element of power system. The motion of synchronous generator is usually modeled as a set of ordinary differential equations (3). For analysis of power system transient stability, (3) will be solved with know initial values.

$$\begin{cases} \dot{\delta} = \omega - \omega_0 \\ \dot{\omega} = \dfrac{\omega_0}{T_j}(P_T - P_E(\delta)) \end{cases} \quad (3)$$

As above mentioned, not only conventional numerical integration but also QSS is the method of solving (3). And the latter has never been used in power system simulation.

Using a uniform quantum $\Delta Q = Q_{k+1} - Q_k$ and hysteresis width ε in both state variables in (3), a quantized state system (4) is presented.

$$\begin{cases} \dot{\delta} = q_2(t) - \omega_0 \\ \dot{\omega} = \dfrac{\omega_0}{T_j}(P_T - P_E(q_1(t))) \end{cases} \quad (4)$$

It can be simulated by a coupled DEVS model, composed by atomic models as shown in Fig.2.

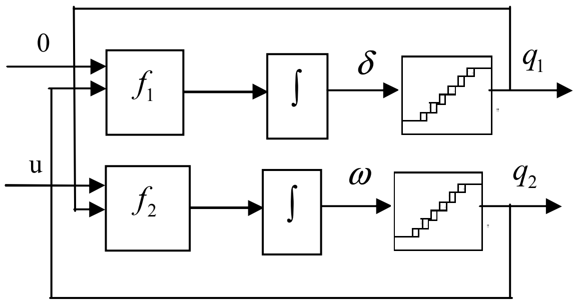

Fig. 2. Block diagram of (4)

where:

$$\begin{cases} f_1(q_1,q_2)=q_2-\omega_0 \\ f_2(q_1,q_2)=\dfrac{\omega_0}{T_j}(P_T-P_E(q_1)) \end{cases} \tag{5}$$

They can be represented as an atomic model, and the Integrator with hysteresis Quantization function(IQ) is another atomic model. The coupling of the four atomic models can be expressed by the connections: $[(IQ_1,1)(f_2,1)]$, $[(IQ_2,1)(f_1,1)]$, $[(f_1,1)(IQ_1,1)]$ and $[(f_2,1)(IQ_2,1)]$. The coupled model also can be used as a component to form larger coupled model.

In a word, when a quantization function was applied to a continuous system, continuous variables were mapped to a sequence of events. Thus a set of differential equations was approximated by a discrete event system. And then, the discrete event system was modeled based on DEVS, which kept the uniform structure of models and could form hierarchical construction.

IV. A Simulation Case

A. DEVS simulation environment

DEVS is a formal framework of M&S. There are many simulation environments which implemented the framework, such as DEVSJAVA[7], DEVS-C++[8], CD++[9], PowerDEVS[10] etc. Among them, PowerDEVS provides a SIMULINK like graphical user interface and discrete state methods for numerical integration of ordinary differential equations. The "continuous library" of PowerDEVS is shown in Fig.3. For primary users, the simulation task can be done by connecting the predefined blocks; for advanced users, any DEVS model can be constructed by defining functions with C++ language. PowerDEVS itself exhibited the flexibility of DEVS framework.

In this paper, PowerDEVS was used as a tool to prove the validity of QSS method in power system dynamic simulation. Therefore, many attractive functions of the simulation environment were not mentioned. The more details of PowerDEVS can be found in [10].

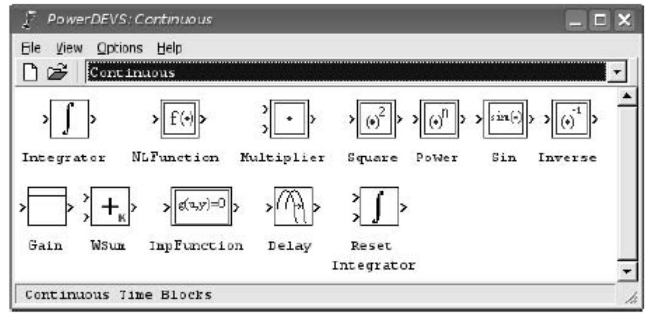

Fig. 3. The continuous library of PowerDEVS

B. test case

The test case is a Single Machine Infinite Bus (SMIB) system as shown in fig.4. All the parameters can be found in ref. [11]. When one of the transmission circuits experiences a solid three-phase fault and the fault is cleared by isolating the faulted circuit, the stability of the synchronous generator can be exhibited by solving the swing equation with the three system conditions. They are (i) prefault, (ii) during fault, and (iii) postfault.

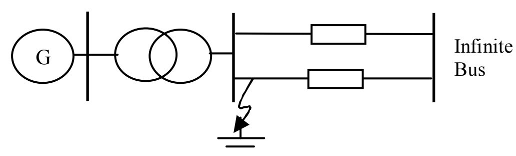

Fig. 4. Test case of SMIB

The PowerDEVS implemented continuous system simulator based on QSS method. The swing equation can be built by the blocks in continuous library as shown in fig.5. The simulation diagram shielded the complexity of DEVS models for us.

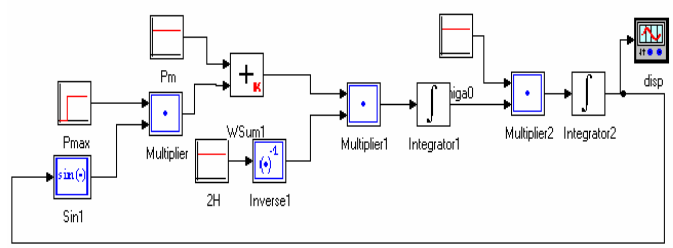

Fig. 5. The simulation diagram of SMIB system

To simplify the presentation of the problem, the simulation case was reduced to the solving of the swing equation, and the time of fault was set as $t_0=0$, the fault was cleared at $t=Tstep$. So a step source was used as input to denote different system conditions. The parameters of each block can be set from pop-up dialog-box. During the cause of simulation, we found that the results were sensitive to the setting of integrator. In the test case, each integrator used QSS2 method and the quantum $dq=0.001$ .The results of discrete state simulation with different fault clearing time are shown in Fig. 6.

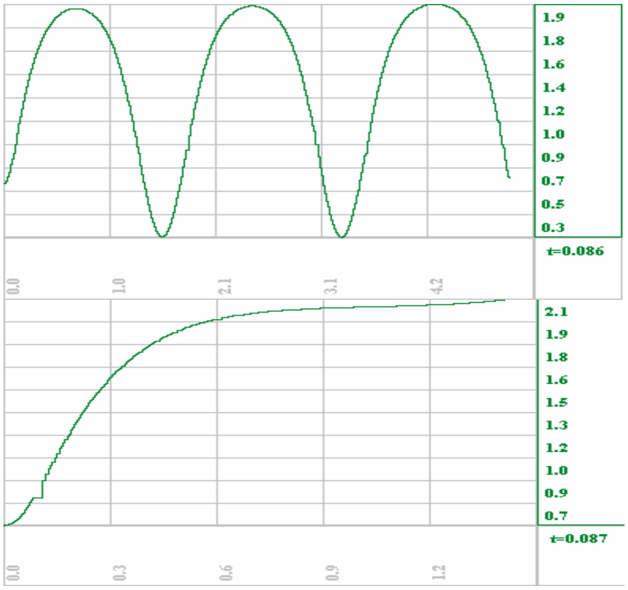

Fig. 6. Rotor angle response for different values of fault-clearing time

In order to compare with the results by discrete time integration method, a new block was added to Fig.4 to save event data to disk. And then all the data was plotted in MATLAB with appropriate axis as Fig. 7. Moreover, function ode45 in MATLAB was used to solve the same swing equations; the result was shown in Fig. 8.

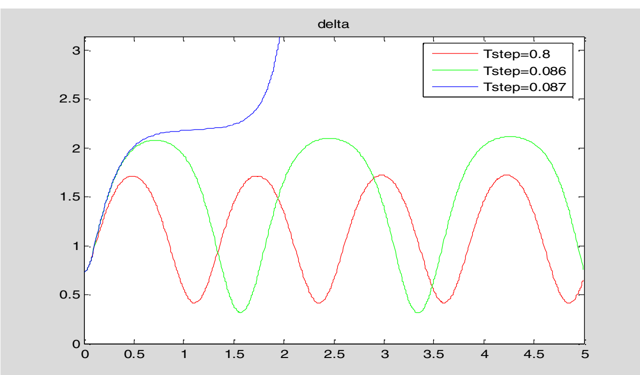

Fig. 7. Rotor angle response using QSS method

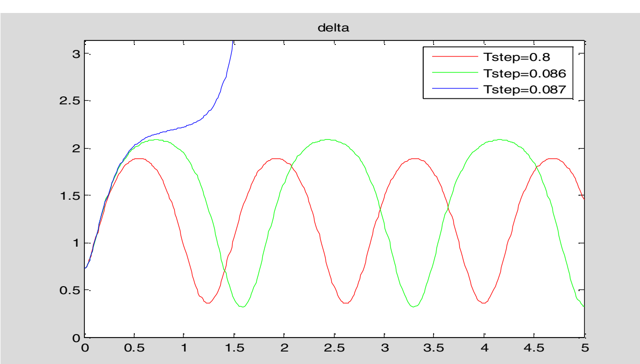

Fig. 8. Rotor angle response using discrete time method

It is obviously that the differences of results by different simulation method are very tiny. That is to say, QSS based method is an optional integrator to simulate power system dynamics.

V. Conclusion and Future Work

The continuous system can be approximated to the discrete event system by the QSS method. Therefore, DEVS frame which has been widely used in discrete event system M&S can also be used to model and simulate continuous system. Generally, power systems are modeled as the continuous system, and the dynamics of the power system elements are represented by different differential equations. So the numerical integration algorithm is the key of any power system dynamic simulation software. When DEVS frame is applied to power system M&S, it is necessary to validate the validity of QSS based integration method firstly.

In this paper, the transient stability problem of a SMIB system was simplified to the solving of swing equations of synchronous generator with initial conditions. After the states of swing equations had been quantized, the rotor response curve constituted by events sequences were got by QSS based integration method with PowerDEVS. This result was compared with the discrete time response curve produced by classical numerical integration algorithm. The conclusion is that the accuracy of QSS base integration method is enough to simulate the dynamic of power system elements as long as appropriate parameters of quantizer are selected.

This paper is the first step of study on DEVS based power system M&S. In future work, all kinds of power system elements could be modeled as uniform DEVS structure. The whole system could be a hierarchical construction. Under the DEVS framework, bran-new power system simulator could be developed, which would have flowing characters: the separation of models from simulators, more scalability, more adaptability to the hybrid essential and distribution of power system, etc. Moreover, the dynamics of power system could be exhibited from the view of transition of states by the new simulator. It would be helpful to understand the evolvement of the system.

VI. References

[1] Kofman, E., Lee, J.S. and Zeigler, B . "DEVS representation of differential equation systems: review of recent advances." In *Proceedings of the 2001 European Simulation Symposium*, Marseille, France, Oct 2001

[2] Zeigler,B. *Theory of Modeling and Simulation*, New York: John Wiley Editor, 1984

[3] Zeigler,B. "DEVS theory of quantized systems," ECE Dep., University of Arizona, June 1998

[4] Kofman, E., "Discrete event based simulation and control of continuous system ," Ph.D. dissertation, Rosario National University. Argentina, 2003.

[5] Mariana C. and Gabriel A., "models for continuous and hybrid system simulation", *in Proc.2003 Winter Simulation Conf., pp. 641-649.*

[6] Nutaro,J. and Zeigler, B., "on the stability and performance of discrete event methods for simulating continuous systems " , *Journal of Computational Physics*, Vol. 227(1),2007

[7] Zeigler,B. and Sarjoughian, H. "Introduction to DEVS Modeling and Simulation with JAVA: A Simplified Approach to HLA-Compliant Distributed Simulations". Arizona Center for Integrative Modeling and Simulation, 2000.

[8] Hyup Cho and Young Cho. "DEVS–C++ Reference Guide". The Universityof Arizona, 1997.

[9] G. Wainer, G. Christen, and A. Dobniewski. "Defining DEVS Models with the CD++ Toolkit". In *Proc. ESS2001*, pp. 633 - 637, 2001.

[10] E.Kofman, M.Lapadula, and E.Pagliero, "PowerDEVS: A DEVS-based environment for hybrid system modeling and simulation," tech. rep., Rosario National University, 2003. http://fceia.unr.edu.ar/~kofman/pubs.html.

[11] P. Kundur, *Power System Stability and Control* ,McGraw-Hill Inc., New York, 1994

VII. Biographies

Lan Tang was born in 1977. He graduated form Zhejiang University, China with BEng in 1999 and received his M.S. degree from Kunming University of Science and Technology, Yunnan, China, in 2004. Currently he is pursuing the Ph.D. degree in school of electrical engineering and automation at Harbin Institute Technology, China.

Shu HongChun was born in 1961. He received the Ph.D. degree in Harbin Institute of Technology, China in 1997 and is post doctor of Xi'an Jiaotong Univercity. He is currently a professor of electrical engineering at Kunming University of Science and Technology (KMUST), Yunnan, China. His recent interests are novel relay of power system, fault location and stability control.