Modeling of power electronics devices in EMTP-TACS

Serge Lefebvre
Hydro-Québec(IREQ)
1800 Montée Ste-Julie
Varennes, Québec, Canada

Ricardo D. Rangel
Cepel
Caixa Postal 2754, CEP 20001
Rio de Janeiro, RJ-Brazil

Abstract - This paper investigates the capabilities and limitations of the EMTP as applied to HVdc system simulations. New features of the EMTP are used to improve the quality and efficiency of the simulations. The representation of digital controls remains a difficult issue and possible solutions around the problem are indicated.

1.- INTRODUCTION

With EMTP, thyristor valves are normally modelled as switches (the v-i characteristic curve of the switching device is missing) while control systems are modelled with TACS, originally designed for the representation of analog controls. These characteristics have been criticized for power electronics simulation [1-4]. Aside from initialization issues, some of the complaints often made about the EMTP when simulating power electronics are 1) numerical oscillations at valve opening requiring numerical snubbers 2) the jitter introduced by the lack of interpolation of the firing instants, 3) the excessive computation time and 4) the difficulty to model digital control systems.

The new Critical Damping Adjustment (CDA) option of the EMTP does take care of the first problem, but CDA in itself does not eliminate the jitter on the switching instants. In section 3, it is shown how interpolation can reduce the jitter. Section 3 also discusses how the penalties on computation times can be alleviated. Section 4 describes development activities regarding a digital TACS, called DTACS, which will provide a program better suited to digital controls.

2.- BACKGROUND

First it is useful to describe main EMTP modeling and algorithm issues which are pertinent to the simulation of power electronics. New features of the EMTP (EPRI/DCG version 3.0) which are used in this work are also presented.

2.1 Main EMTP algorithms for power electronics applications

2.1.1 Control system modeling

TACS limitations arise when modeling digital controls. Free-format pseudo-fortran is a crude alternative providing some TACS flexibility towards digital control, but it is severely limited in terms of functionality and it is certainly too slow to deserve long term consideration in this perspective. Another problem is the presence of time delays between the TACS solution and the network solution, as well as within some TACS nonlinear loops. These internal TACS delays are the result of the non-iterative TACS solution. The TACS ordering scheme is designed to limit the number of time delays, and not to improve sparsity. Because of the complexity of the mechanisms, it occasionally happened in previous releases that time delays were incorrectly inserted or history terms incorrectly updated. A third limitation is the lack of efficient general initialization for nonlinear control systems.

2.1.2 Valve modeling

Thyristors are ideal switches controlled by voltage, time and current. They are modelled in the nodal admittance matrix of the discretized network. Consider a switch between nodes k and m. If the switch is closed, the EMTP adds the respective rows and columns of the nodal matrix (formed with all switches open) to form a new axis m, and axis k is discarded. If the switch is open, no modifications to the original matrix are necessary. Then the nodal matrix is triangularized.

If all nodes to which the thyristor switches are connected are kept in a separate partition to be re-triangularized when topological changes occur and if the time step is kept constant throughout the entire simulation, then partial re-factorization could be used. For a time this was part of the EMTP, but it is not used anymore because the option often

failed in terms of memory/time efficiency [5] due to destruction of the sparsity structure when there are several switches. In current EMTP versions, except the UBC version, the complete admittance matrix needs to be built and triangularized after each topological change. For a single HVdc converter, topological changes occur 12 times per cycle per six-pulse bridge. Hence, whether the sparsity structure is destroyed or not, there is a considerable computational burden imposed on the EMTP.

2.1.3 Trapezoidal integration

Sudden changes in current across inductors or voltages across capacitors can result in oscillations with EMTP trapezoidal method network simulations. These changes can be the result of switching actions or the variations in the operating point on nonlinear elements. Chopping exactly at current zero does not eliminate the numerical oscillation when re-initialization is not performed. (Numerical oscillations will also result from an excessively large time step.) Hence, in power electronics applications, there are numerical difficulties because of the need to integrate over a succession of sharp voltage discontinuities across the inductive branches of the network, caused by the interruption of the current in other branches. It is very important to remove the numerical oscillations so that they do not interfere with the firing controls of the power electronics device.

2.1.4 Fixed time step

For simplicity, the EMTP uses a fixed and single time step for all network and control system components. In addition of restricting the time step to that of the component with the fastest dynamics, there is the disadvantage of solution jittering in power electronics devices. The reason is the following. A firing pulse generator controls valve closing points which are typically referenced to a zero crossing point of an ac voltage. Neither the reference point nor the firing point falls on a time step. Firing point errors are of the order of half a time step, which is thus maintained at relatively low values (25 to 50 μsec) in the interest of accuracy.

2.2 Some new EMTP features

Here is a list of new EMTP features useful for improving the efficiency and quality of power electronics simulation.

2.2.1 Revised TACS

The first step in upgrading TACS has been to improve the ordering algorithm and correct miscellaneous coding bugs. Initialization of nonlinear systems has been improved (at the expense of an initial iterative loop) even though not all control systems can be automatically initialized. For example, some supplemental devices are ignored at initialization. However, supplemental variables (e.g. defined by pseudo-fortran statements) are now part of the initialization.

Contrary to MODELS [6], the revised TACS does not yet have a much larger library of special functions than its TACS predecessor. Important features have been provided though. IF-THEN-ELSE structures of supplemental variables are now allowed. The user also has the possibility to incorporate calls to his/her subroutines in the data file. Each call to a subroutine is interpreted as usage of a device, and the subroutine is ordered in the solution process. Thus it is possible to alleviate many, but not all, problems associated with digital control modeling with this approach. Dynamic linking of user-supplied control routines can be accepted by the new TACS program interface (UNIX). Working directly with compiled add-on routines has proven much more effective than interpreting the pseudo-FORTRAN code of TACS. A compensation interface between the network and TACS [7] can be used when numerical difficulties may arise due to time delays.

Other changes to TACS which are being implemented to make it even more suited to digital controls are listed in section 4.

2.2.2 Valve modeling

If so desired, it is now possible to include any valve model in the EMTP as a user-supplied device in

TACS having connectivity with TACS firing signals, and network voltages and currents. In particular, it is possible to define a 6-pulse thyristor converter module which can be used as a building block for converters. Details of this are given in section 3. It will also be shown how solution jittering can be minimized in a simple manner.

2.2.3 Critical Damping Adjustment (CDA)

CDA [8] is an efficient and reliable way to eliminate numerical oscillations in converter bridges [9]. The CDA procedure is applied throughout the network whenever there is a topological change in a converter or a nonlinear element.

3.- APPLICATION OF NEW FEATURES TO POWER ELECTRONICS SIMULATION

To demonstrate improvements achievable, an HVdc application is selected as shown in Fig. 1. The dc side is modelled by a resistance and an inductance, plus a constant dc voltage representing an inverter connected to a very large ac system. The rectifier is a 12-pulse converter with the associated transformers. The controls generate firing pulses for constant angle operation (18.2 degrees). Filters are not used and the rectifier short-circuit ratio is very strong. This system has been extracted from [10] and we have added a large model of the Hydro-Québec ac system to make the system size representative. The intent is to present potential improvements in the simulation of power electronics and not to examine ac/dc interaction problems.

3.1 Converter module

By incorporating a converter module as a power-building-block in the EMTP (TACS), more efficiency is achievable in the EMTP. This idea is simple and has been exploited in different forms. A consequence of modeling the converter directly in TACS is that time-step errors between EMTP and TACS are minimized (and eliminated if compensation is used). More importantly, the user directly takes advantage of the partitioned structure

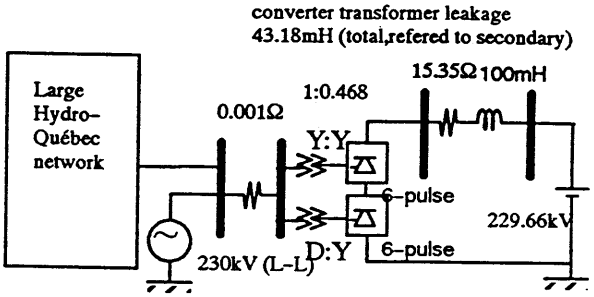

Fig. 1 Single-line diagram of the simulation model for a HVdc system

which is key in reducing the computation time. Nodal matrix re-factorizations of the entire system are not performed when a valve changes status, changes are limited to the converter module.

Fig. 2 shows the converter module interfaces. The building blocks for HVdc network modeling are the converter and the converter transformer. Each module is connected to two HVdc buses and three ac buses (the three-phase commutation bus). The converter can be modeled as a pseudo-nonlinear (disconnected subnetwork) or a true-nonlinear (connected subnetwork) power device. A 12-pulse module (using two separate converter transformer banks) is obtained by series connecting two 6-pulse modules. The converter transformer is part of 6-pulse converter module. When two 6-pulse converter transformers share a common primary winding, then a dedicated 12-pulse unit must be used. On the network side voltage and current slave sources are used to model the converter.

A converter module exchanges information with the ac and dc networks, as well as with the control system and possibly other converter modules. The simplest pseudo-nonlinear input interface is based on prediction of the ac commutating bus voltages and of the dc current through the smoothing reactor, such as in [11], control signals being readily accessible and ordered for the TACS time-step solution. By using prediction, there is no need to calculate Thévenin equivalents of the ac and dc networks, hence the solution is typically faster than with compensation (true-nonlinear), possibly at the expense of accuracy and numerical stability. For the test system, due to the dc line and converter

impedances parameters, compensation for the dc current interface is a requirement. Prediction is satisfactory for the ac side because the ac voltage is stiff. Of course, as the short-circuit ratio is decreased, this may no longer be valid, in which case compensation is required. Hence, a mixed pseudo-nonlinear/true nonlinear representation of the converter is used. To make compensation efficient, impedances of Thévenin equivalents are computed only upon request otherwise in EMTP this is done every time step.

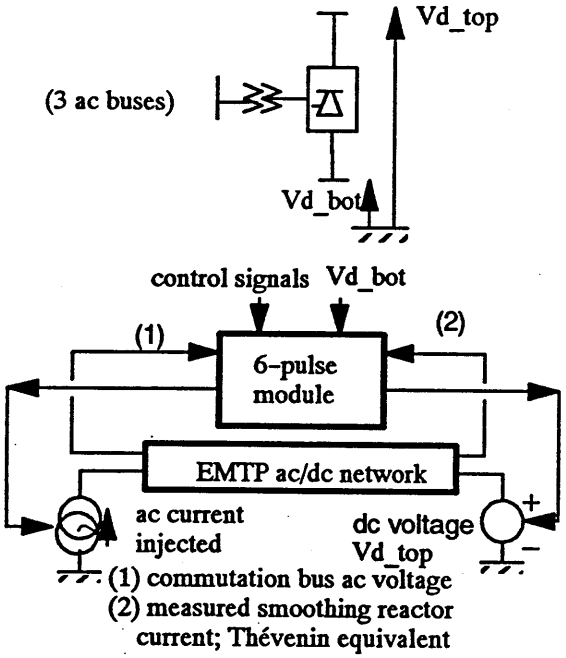

Fig 2 Converter module representation with mixed pseudo-nonlinear and true nonlinear interfaces

The inputs of a converter module are: commutation voltages, measured smoothing rector current, firing pulses to each valve, voltage at bus Vd_bot. The outputs are the injected current in each ac phase, and the dc voltage (with respect to Vd_bot) applied on the smoothing reactor.

The large ac network includes 963 nodes and 1753 branches. The HVdc link of Fig. 1 adds only a few nodes to the nodal matrix. A simulation of 0.500sec using a rather small time-step of 20 μsec is performed to evaluate the time in the time loop with different options. All cases in Table 1 are run without CDA. Table 1 compares the cpu time to solve the ac network to the same accuracy. Only the relative times are important, but let us mention that for the base case (a) the time is 513.5 sec. This is with the valves modelled in the EMTP, but with firing inhibited. Thus, it includes the overhead of control system simulation in TACS. The first three cases are for the system in Fig.1, while in the two last cases an additional dc tie is included. The modular simulations illustrated in cases (c) and (e) are more efficient than the standard EMTP simulations of (b) and (d).

TABLE 1: Efficiency of different simulations

Case	Description	time-step time
a	one dc tie, valves in EMTP but blocked,(basecase)	100%
b	one dc tie, valves in EMTP firing enabled	145% (41% more than case (c))
c	one dc tie, converter module firing enabled	103%
d	with two dc ties, valves in EMTP firing enabled	171% (55% more than case (e))
e	with two dc ties, converter modules firing enabled	110%

3.2 Elimination of jitter

Linear interpolation can be used to locate a zero crossing between time steps and adjust the valves delay time from this point [12]. This reduces the power electronics jitter by making it dependent on interpolation errors rather than on the step size. For the test system, Fig. 3 shows the steady-state voltages and currents for two different step-size values with/without interpolation. All these cases have been run with EMTP, including CDA, with a fixed firing angle setting. With a small step size, the dc current is correctly established at 1kA. Without interpolation, at 100usec step size, the dc current is reduced and modulated. With interpolation, the result is improved.

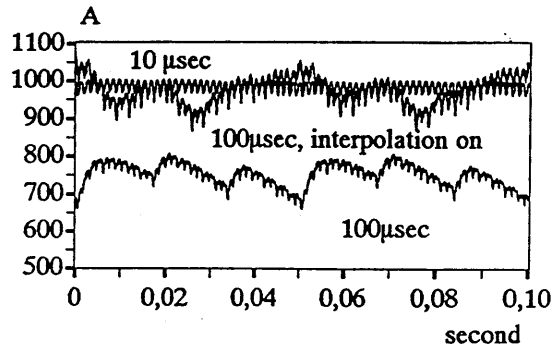

Fig 3: DC current at fixed firing angle

4. MODELING OF DIGITAL CONTROLS

TACS can be used quite satisfactorily to simulate analog controls, but when digital controls are the main issue a more specific tool is needed. Generally we need a control that would be solved only upon request and not every EMTP time step. Some aspects of the development of this tool will be treated here.

4.1 Suitability of analog TACS to digital control

First, some comments about aspects of analog TACS which may be related to Digital TACS (DTACS) characteristics:

- Representation of continuous systems by transfer functions in the Laplace transform ⇒ In digital controls Z transform transfer functions should be used instead.

- The non-iterative block-by-block TACS sequential solution requires block ordering to minimize inserted time delays ⇒ In DTACS no ordering will be necessary.

- Inefficiency in the solution of free-format pseudo-FORTRAN expressions ⇒ Comparisons between the performance of user-defined FORTRAN subroutines, old-fixed format pseudo-FORTRAN expressions and free-format pseudo-FORTRAN expressions showed that the last ones are the most inefficient due to the overhead in the solution caused by their interpretation. Since they will be certainly used widely in DTACS controls it is important to improve their performance: it is possible to eliminate many tests in the solution subroutine just by manipulating some pointers. This approach can be used to the present code but perhaps for the DTACS code a new data structure (ex: binary trees) should be considered for improving free-format pseudo-FORTRAN performance.

- IF-THEN-ELSE structures ⇒ Although it aimed to turn the data input of supplemental variables closer to real FORTRAN it does not quite match the solution philosophy. Any instruction for execution flux control in a programming language only makes sense because the natural processing order is sequential and fixed. Since in TACS all elements are ordered by the program (IF-THEN-ELSE structures are treated like super-blocks), the user does not know beforehand which IF-THEN-ELSE will be solved first and therefore it is not possible to define the same supplemental variable inside two different IF-THEN-ELSE structures because the result would be unpredictable. These structures can only be regarded as switches that provide alternative definitions for supplemental variables and not true flux control statements.

4.2 Basic requirements of Digital TACS

The basic requirements for DTACS are:

- The digital systems to be represented by DTACS will have intervals of solution multiple of the EMTP time step (as long as the EMTP does not provide interpolation of variables)

- A digital control does not respond instantly. ⇒ This makes it possible to perform data acquisition of TACS or network variables at the beginning of the time step in which the digital control is activated. The digital control can be solved independently from TACS since changes in TACS variables will only affect them when a new data acquisition takes place. Each DTACS control must have a certain execution time specified. Before this time (counted after activation) has elapsed new values of calculated output variables of a DTACS control will be kept in buffers until they can be made available to TACS.

- DTACS controls should be solved before analog TACS. ⇒ This allows TACS to use the most recently output values of DTACS controls that have completed their execution cycle.

- DTACS controls ⇒ They will communicate with TACS by a list of input variables and a list of output variables. All other variables will be considered internal (local) to the control and will not be accessible for plotting. An instruction like "PRINT" or "DUMP" can be provided to output internal variables for debugging purpose.

- DTACS activation ⇒ Each DTACS control will have a logical activation expression associated with it. This feature will make it possible to activate the control only in certain integration steps. The activation expression can contain output variables from other DTACS controls.

- Sequential solution ⇒ If description language of DTACS is to be similar to a programming language like FORTRAN, the execution inside each control is to be sequential (except for certain flux control instructions). This means that no ordering will be imposed to the macro-instructions that describe the control. IF-THEN-ELSE, DO and DO-WHILE structures would be recommendable

- Indexed variables ⇒ Looping structures are only useful with indexed variables. These however would be internal to DTACS since TACS does not allow them.

4.3 Digital TACS developments

The work in progress is the definition of data structures to be used in the programming of the new tool and its interface with present TACS.

5.- CONCLUSION

The paper has shown how to alleviate in EMTP the problem of jittering and computation burden in power electronics simulation. The difficulty to model digital control systems should be overcome by DTACS. DTACS will not replace TACS. It will simply provide digital controls as a kind of specific programmable devices.

6.- REFERENCES

[1] P. Lehn, J. Rittiger, B. Kulicke, "Comparison of the ATP version of the EMTP and the NETOMAC program for simulation of HVdc Systems," paper 95 WM 276-6 PWRD

[2] T.L. Maguire, A.M. Gole, "Digital simulation of flexible topology power electronic apparatus in power systems,", paper 91 SM 414-3 PWRD

[3] T. Ino, R.M. Mathur, M.R. Iravani, S. Sasaki, "Validation of digital simulation of DC links," IEEE Trans. PAS, 1985, pp. 2596-2602.

[4] J.G. Campos Barros, R. Rangel, "Computer simulation of modern power systems: The elimination of numerical oscillations caused by valve action," 4th Int Conf on AC/DC Power Trans, Sept 1985, #255, pp. 254-259.

[5] EMTP Theory Book, Microtran Power System Analysis Corporation, May 1992

[6] L. Dubé, I. Bonfanti, "MODELS: A new simulation tool in the EMTP," European Trans. on Elec. Power Eng., Vol 2, No 1, pp.45-50, Jan/Feb 1992.

[7] S. Lefebvre, J. Mahseredjian. "Improved control systems simulation in the EMTP through compensation", IEEE Trans. PWRD, Vol 9, pp 1654-1662, July 1994.

[8] J.R. Marti, J. Lin, "Suppression of numerical oscillations in the EMTP," IEEE Trans. PWDR, Vol 4, No 2, pp. 739-747, 1989.

[9] S. Lefebvre, L. Gérin-Lajoie, G. Beaulieu, "Description and typical EMTP applications of static compensator models", IMACS, TC1'93, pp. 213-219.

[10] Ned Mohan, Computer exercises for power electronics education, EE Dept, University of Minnesota. (Exercise 54)

[11] D.A. Woodford, A.M. Gole, R.W. Menzies,"Digital simulation of dc links and ac machines," IEEE Trans. PAS, 1983, pp. 1616-1623.

[12] R. H. Lasseter, K.H. Kruger, "HVDC simulation using NETOMAC," IEEE Montech'86, Sept 1986.