



Missouri University of Science and Technology
Scholars' Mine

Electrical and Computer Engineering Faculty
Research & Creative Works

Electrical and Computer Engineering

01 Aug 1993

A Study of Numerically Efficient Algorithms for Power System Dynamic Analysis

J. G. Chen

Mariesa Crow

Missouri University of Science and Technology, crow@mst.edu

Follow this and additional works at: https://scholarsmine.mst.edu/ele_comeng_facwork

 Part of the [Electrical and Computer Engineering Commons](#)

Recommended Citation

J. G. Chen and M. Crow, "A Study of Numerically Efficient Algorithms for Power System Dynamic Analysis," *Proceedings of the 36th Midwest Symposium on Circuits and Systems (1993, Detroit, MI)*, vol. 1, pp. 712-715, Institute of Electrical and Electronics Engineers (IEEE), Aug 1993.

The definitive version is available at <https://doi.org/10.1109/MWSCAS.1993.342947>

This Article - Conference proceedings is brought to you for free and open access by Scholars' Mine. It has been accepted for inclusion in Electrical and Computer Engineering Faculty Research & Creative Works by an authorized administrator of Scholars' Mine. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

A Study of Numerically Efficient Algorithms for Power System Dynamic Analysis

James G. Chen Mariesa L. Crow
Department of Electrical Engineering
University of Missouri-Rolla
Rolla, Missouri 65401

Abstract: In this paper, the multirate method will be introduced to analyze power system behavior including linear and non-linear systems with widely varying time constants. The development and study of time domain simulation techniques and error detection will be discussed. The results, both in terms of accuracy and computation time, will be compared to traditional simulation methods in a small nonlinear power system example.

1. Introduction

Computational complexity is of timely concern in the assessment of dynamic security. Steady-state and transient stability computational methods have studied in depth and many robust and widely-used tools are available for analyses in these time frames. Unfortunately, the development of computational tools for dynamic (mid- to long-range) analysis lags far behind. The ability to develop such methods is further complicated by the lack of appropriate models for various system components. In response, the power engineering community has tried to incorporate more detailed models into simulators. The inclusion of increasingly detailed models has further increased the complexity of the numerical calculations.

One family of methods which has been used for power system simulation are variable-step methods [1][2]. Variable-step methods are integration techniques in which the time step may vary in accordance with the fastest varying state in the system. The variable-step method is well suited for simulating dynamic systems which are primarily slow response systems, but exhibit infrequent fast decaying transients. The method is not well suited for systems in which the fast response is sustained for a large portion of the simulation interval. Unfortunately, this is the case when the power system contains induction machines under continually changing loading levels. So, while only a small portion of the entire system state are affected by fast dynamics of the induction machine loads, the integration time step must remain small, thus computational efficiency is lost. This is again the case when the power system is modelled with fast switching devices, such as FACTS devices, or the converters necessary

to inter tie DC lines into an AC system.

Recently, multirate methods have been proposed to effectively simulate systems with this widely varying time response behavior[3][4]. Multirate methods are distinguished from variable-step methods in that the system states are aggregated into loosely coupled components which are then integrated individually with a time step dictated by the time response of the component. The coupling between components is either neglected or estimated in some way.

Computational speed-up is achieved if the number of rapidly varying components is small compared to the number of slowly varying components. High accuracy is achieved by retaining both the gross and specific behavior of the system. The potential of this method for power system simulation is great.

In this paper, a study of multirate methods has established the viability of this type of numerical method for efficient simulation of power system dynamics. As a dynamics. As a first approach, the multirate method has been applied to a generalized linear system which may encompass a separation into n distinct time scales. The results of this study will be used to ascertain the stability of the numerical method for any given time scale separation between states. One of the main results of the preliminary linear system study is the development of a formula to estimate the possible obtainable speed-up given any number of time scales and the separation between them. The multirate method is then extended to a small nonlinear power system example which exhibits a time scale separation into two and three distinct time scales.

2. The multirate method for linear systems

A multirate method for integrating ordinary differential equations is one in which different equations are integrated by using different step sizes. The multirate method combines the robustness of a variable-step method with independent step size capabilities. The principle of the multirate method is the integration of each variable with a steplength which is necessary and sufficient for the requested accuracy. Although multirate methods are conceptually simple, there are still many problems and open questions

regarding their theory, formula, and implementation. This section discusses the use of multirate methods for solving a linear system with n ordinary differential equations.

2.1. The multirate method for three time scales

Consider a 3×3 linear time-invariant system of differential equations, which may be integrated with 3 different step sizes. For simplicity, let $C_i!$ be defined as

$$C_i! = C_i \times C_{i-1} \times C_{i-2} \times \dots \times C_2 \times C_1$$

for $C_i, i \in Z^+$, and $C_i = h_i/h_{i-1}$. Note that this expression is different from the mathematical expression "C!". Also note that $C_1!$ is equal to 1. Without loss of generality, it can be assumed that $h_1 \leq h_2 \leq h_3 \dots \leq h_{n-1} \leq h_n$

$$\begin{aligned} \text{so that } h_2 &= C_2 \times C_1 \times h_1 \\ h_3 &= C_3 \times h_2 = C_3 \times C_2 \times C_1 \times h_1 \\ &\vdots \\ h_n &= C_n h_{n-1} = C_n \times C_{n-1} \times \dots \times C_2 \times C_1 \times h_1 = C_n! h_1 \end{aligned}$$

The three time scale case is illustrated in the following figure, where $y_1(t)$ is the fastest varying state and $y_3(t)$ is the slowest varying state. Note that $h_2=2h_1$ and $h_3=2h_2=4h_1$, thus $C_2=2, C_3=2$, and $C_3!=4$.

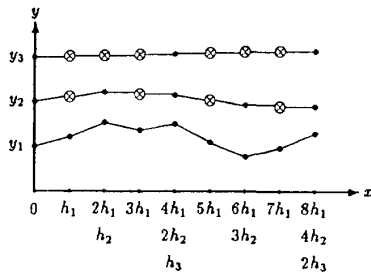


Figure--1: Three time scale example

Consider the calculation of the system states at time $t = 4h_1 = 2h_2 = h_3$:

$$\begin{aligned} y_3(t+h_3) &= y_3(t) + \frac{h_3}{2} \left[a_{31}(y_1(t+4h_1) + y_1(t)) \right. \\ &\quad \left. + a_{32}(y_2(t+2h_2) + y_2(t)) + \right. \\ &\quad \left. a_{33}(y_3(t+h_3) + y_3(t)) \right] \end{aligned} \quad <1>$$

$$\begin{aligned} y_2(t+2h_2) &= y_2(t+h_2) + \frac{h_2}{2} \left[a_{21}(y_1(t+4h_1) + \right. \\ &\quad \left. y_1(t+h_1)) + a_{22}(y_2(t+2h_2) + \right. \\ &\quad \left. y_2(t+h_2)) + a_{23}(y_3(t+h_3) + y_3(t+h_2)) \right] \end{aligned} \quad <2>$$

$$\begin{aligned} y_1(t+4h_1) &= y_1(t+3h_1) + \frac{h_1}{2} \left[a_{11}(y_1(t+4h_1) + y_1(t+3h_1)) \right. \\ &\quad \left. + a_{12}(y_2(t+2h_2) + y_2(t+3h_1)) + a_{13}(y_3(t+h_3) \right. \\ &\quad \left. + y_3(t+3h_1)) \right] \end{aligned} \quad <3>$$

Note that each variable is integrated with the step size which

is appropriate for its time response. Note also that not all states are available at the desired time (those marked by \otimes in Figure 1) and must be approximated. The simplest approximation is a linear interpolation between calculated values.

After repeated interpolations and substitutions, the following expression for $y_1(t+h_3)$ may be obtained:

$$\begin{aligned} y_1(t+h_3) &= \beta_1^{C_3!} y_1(t) + \sum_{j=1}^{C_3} \sum_{k=1}^{C_2} \frac{h_1 a_{12}}{2-h_1 a_{11}} \beta_1^{C_2(C_3-j)} \times \beta_1^{C_2-k} \\ &\quad \left[\left(2 - \frac{2k-1}{C_2!} \right) y_2(t+(j-1)h_2) + \left(\frac{2k-1}{C_2!} \right) y_2(t+jh_2) \right] \\ &\quad + \sum_{i=1}^{C_3!} \frac{h_1 a_{13}}{2-h_1 a_{11}} \beta_1^{C_3+i} \left[\left(2 - \frac{2k-1}{C_3!} \right) y_3(t) + \left(\frac{2k-1}{C_3!} \right) \times \right. \\ &\quad \left. y_3(t+h_3) \right] \end{aligned} \quad <4>$$

$$\text{where } \beta_i = \frac{2+h_i a_{ii}}{2-h_i a_{ii}} \quad \forall i \in Z^+$$

Similarly, expressions for $y_2(t+h_3)$ and $y_3(t+h_3)$ may be found. Note that since the trapezoidal method is an implicit method, there is an implicit dependence on the variables at both previous and current time steps in addition to the dependence introduced by the interpolation. This reduction process will be discussed in the next section for a generalized linear system of n distinct time scales.

2.2 The multirate method for n time scales

Consider a system of n linear functions:

$$\begin{aligned} \dot{y}_1 &= a_{11}y_1 + a_{12}y_2 + \dots + a_{1n}y_n \\ \dot{y}_2 &= a_{21}y_1 + a_{22}y_2 + \dots + a_{2n}y_n \\ &\vdots \\ \dot{y}_n &= a_{n1}y_1 + a_{n2}y_2 + \dots + a_{nn}y_n \end{aligned} \quad <5>$$

Here, $a_{ij} \in \mathfrak{R}$ for $1 \leq i \leq n$ and $1 \leq j \leq n$. Following the same approach discussed for the three time scale system, the following expressions may be obtained for any $1 \leq i \leq n$:

$$\begin{aligned} y_i(t+h_n) &= \beta_i^{C_n!} y_i(t) + \sum_{m=1}^{i-1} \sum_{k=1}^{C_m!} \beta_i^{C_m!k} \frac{h_1 a_{im}}{2-h_1 a_{ii}} \times \\ &\quad \left[y_m(t+(k-1)h_i) + y_m(t+kh_i) \right] + \\ &\quad \sum_{m=i+1}^n \sum_{j=1}^{C_m!} \sum_{k=1}^{C_m!} \left\{ \beta_i^{C_m!(C_m!-j)} \beta_i^{C_m!k} \frac{h_1 a_{im}}{2-h_1 a_{ii}} \right. \\ &\quad \left[\left(2 - \frac{C_m!(2k-1)}{C_m!} \right) y_m(t+(j-1)h_m) \right. \\ &\quad \left. \left. + \left(\frac{C_m!(2k-1)}{C_m!} \right) y_m(t+jh_m) \right] \right\} \end{aligned} \quad <6>$$

As in the previous discussion, it is possible to apply a reduction process to find a closed form for the matrix M which relates

$$y(t+h_n) = M y(t) \quad <7>$$

The form of the matrix M is important for a variety of reasons. Firstly, the numerical stability of any integration

method can be ascertained by computing the eigenvalues of M . This aspect of the matrix M is not considered in this paper. The matrix M can be used to estimate the potential savings gained from the multirate method. Note from the discussion of Section 2.1 that at any point in time, only a portion of the entire system is calculated at any given time. Thus rather than solving an $n \times n$ system at each step (requiring on the order of n^2 multiplication and divisions if the system is sparse), considerably less computation is involved.

3. An Illustrative Example

The multirate method can be well illustrated with a small synchronous machine model, which is given in [5]

$$\frac{1}{w_s} \frac{d\Psi_d}{dt} = -\frac{R_s}{L'_d} \Psi_d + \frac{R_s}{L'_q} E'_q + \frac{w}{w_s} \Psi_q + V \sin \delta \quad <8>$$

$$\frac{1}{w_s} \frac{d\Psi_q}{dt} = -\frac{R_s}{L'_q} \Psi_q - \frac{R_s}{L'_d} E'_d - \frac{w}{w_s} \Psi_d + V \cos \delta \quad <9>$$

$$T'_d \frac{dE'_d}{dt} = -\frac{L_q}{L'_q} E'_d - \frac{(L_q - L'_q)}{L'_q} \Psi_q \quad <10>$$

$$T'_q \frac{dE'_q}{dt} = -\frac{L_d}{L'_d} E'_q - \frac{(L_d - L'_d)}{L'_d} \Psi_d + E_{ic} \quad <11>$$

$$\frac{d\delta}{dt} = w - w_s \quad <12>$$

$$\begin{aligned} \frac{2H}{w_s} \frac{dw}{dt} = T_m + \left(\frac{1}{L'_d} - \frac{1}{L'_q} \right) \Psi_d \Psi_q \\ + \left(\frac{1}{L'_q} \right) \Psi_d \Psi_q + \left(\frac{1}{L'_d} \right) \Psi_q E'_q \quad <13> \end{aligned}$$

The model contains the two stator / network flux linkages Ψ_d and Ψ_q . The voltage proportional to the field flux linkage E'_q , the voltage proportional to the damper winding flux linkage E'_d , and the electromechanical pair δ, w . The constant infinite bus voltage magnitude is given as V . The data for this example is given in [5]. The applied disturbance is a reduction of the infinite bus voltage from 1.0 to 0.8pu at time $t=0.2$ sec.

The flux linkage variables Ψ_d and Ψ_q both exhibit highly oscillatory, negligibly damped responses with a frequency close to 60Hz. These are the "fast" variables. In order to effectively capture the dynamics of these oscillatory variables, the integration step size must be small enough to accurately reproduce the shape of the sinusoid. A sinusoid waveform can be nominally reconstructed from 8 points per cycle, but 16 points per cycle is preferable. For this reason, in this example, the integration step size is chosen to be

$$h = \frac{(\text{frequency})^{-1}}{16} = \frac{1/60}{16} \cong 0.001s$$

If a variable step integration method were used to simulate this system, all the system variables would be discretized using the same time step $h=0.001$ seconds. This time step

could not be increased for better computational efficiency, because the oscillations are only negligibly damped, and thus would not decay in the time frame of interest, thus all advantages to using a variable step method are lost. The multirate method, however, is well suited to this type of problem. This system has a well-defined separation of time responses.

3.1 Two time scales

In the first example, a two-time scale separation will be considered, that is, the variables $[E'_d, E'_q, \delta, \omega]$ will be "slow" variables compared to $[\Psi_d, \Psi_q]$, and will be integrated with a step size $h_2 = C_2 h_1$, where $1 \leq C_2$. The results of this comparison are summarized in Table 1 which gives the maximum percent error over the simulation interval for each variable using the multirate method as compared to a constant step size method with $h_1 = 0.001s$. The computation time required as a function of C_2 is shown in Figure 2.

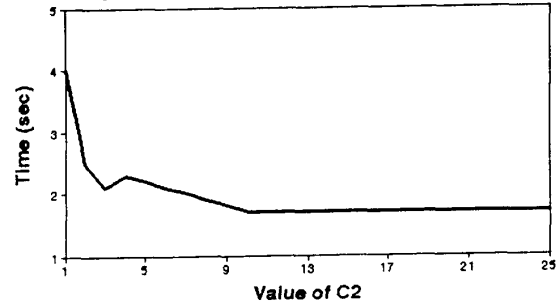


Figure 2: Computation time vs. C_2

Table 1: C_2 vs. Percent Error and Computation time

| C_2 | Ψ_d | Ψ_q | E'_d | E'_q | δ | ω | cpu |
|-------|----------|----------|--------|--------|----------|----------|-----|
| 1 | | | | | | | 4.0 |
| 2 | 0.34 | 0.64 | 0.48 | 0.01 | 0.06 | 0.00 | 2.5 |
| 3 | 0.80 | 1.55 | 1.05 | 0.03 | 0.17 | 0.00 | 2.1 |
| 4 | 1.27 | 2.49 | 1.92 | 0.05 | 0.32 | 0.01 | 2.3 |
| 5 | 1.58 | 3.32 | 2.80 | 0.08 | 0.52 | 0.02 | 2.2 |
| 6 | 1.69 | 3.52 | 3.77 | 0.11 | 0.77 | 0.03 | 2.1 |
| 7 | 1.49 | 3.15 | 4.66 | 0.15 | 1.08 | 0.03 | 2.0 |
| 8 | 0.87 | 3.20 | 5.29 | 0.19 | 1.47 | 0.04 | 1.9 |
| 9 | 1.11 | 3.98 | 6.39 | 0.25 | 1.96 | 0.05 | 1.8 |
| 10 | 2.87 | 5.04 | 8.54 | 0.33 | 3.59 | 0.06 | 1.7 |

The relationship of time vs. C_2 is not unexpected. When $C_2=1$, the computational burden will be dominated by the solution of the full 6×6 system. As C_2 increase, the dominance will shift to the 2×2 fast system, until the point where the infrequent computation of the full system is a small portion of the overall computation. The slight increase in computation time at $C_2=4$ is due to the increase of required Newton-Raphson iterations to achieve the required convergence accuracy. Far better speed-ups would be expected from a larger example where sparsity could be exploited or where the ratio of fast to total variables is

smaller than one-third, as in this example.

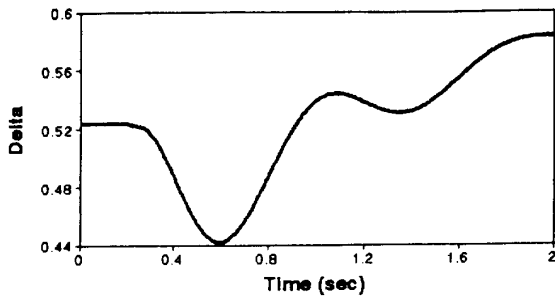


Figure 3: Standard output waveform for δ ($C_2=1$)

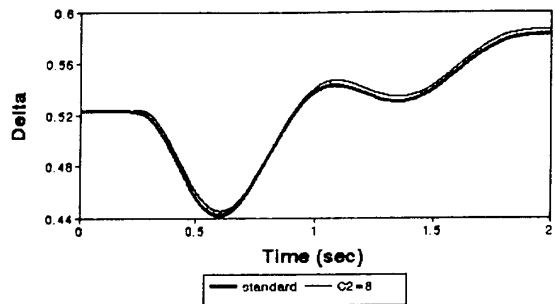


Figure 4: Output waveform for δ ($C_2=8$)

3.2 Three time scales

In the two time scale case, note that the variable E'_d has the largest error of all variables. This is due to the designation that E'_d is a "slow" variable, when in fact it maybe considered a "medium" variable, not quite fast, but requiring more frequent updating than the slow variables. This necessitates the introduction of a third time scale. A selection of three time scale results are presented in Table 2. Recall that the medium step size $h_2 = C_2 h_1$ and slow step size $h_3 = C_3 C_2 h_1 = C_3 h_2$. consider the two time scale example for $C_2=8$, where the variables $[E'_d, E'_q, \delta, \omega]$ are only integrated every $8h_1$. The error for E'_d is 5.3%, the error for Ψ_q is 3.2%, and the δ error is 1.4%. In the three time scale example $C_2=2$, $C_3=4$, thus $h_2 = 2h_1$, and $h_3 = 4h_2 = 8h_1$, and the variables $[E'_d, \delta, \omega]$ are only integrated every $8h_1$, whereas E'_q is integrated every $2h_1$. In this case the error in E'_d is reduced to 1.8%, while the errors in Ψ_q and δ remain fairly constant. Note however, that the computational time is increased slightly from 1.9 CPU to 2.1 CPU, thus trading accuracy for efficiency. Even in the case where $C_2=4$ and $C_3 = 2(h_2 = 4h_1, h_3 = 8h_1)$, the error in E'_d is still reduced to about 3% with all other errors remaining constant. In this case, the computational time is 1.9 CPU, the same as the two time scale case. These results are intuitive; the more frequently a "medium" variable is calculated, the greater the

accuracy, but often at the expense of computational efficiency.

Table 2: C_2 & C_3 vs. Percent Error and CPU Time

| C_2 | C_3 | Ψ_d | Ψ_q | E'_d | E'_q | δ | ω | cpu |
|-------|-------|----------|----------|--------|--------|----------|----------|-----|
| 1 | 1 | | | | | | | 4.0 |
| 2 | 2 | 1.30 | 2.31 | 0.71 | 0.03 | 0.36 | 0.01 | 2.3 |
| 2 | 3 | 1.69 | 3.00 | 1.18 | 0.05 | 0.88 | 0.03 | 2.3 |
| 2 | 4 | 0.75 | 3.39 | 1.86 | 0.08 | 1.66 | 0.04 | 2.1 |
| 2 | 5 | 1.50 | 5.91 | 3.14 | 0.15 | 3.07 | 0.07 | 2.0 |
| 3 | 2 | 1.67 | 3.09 | 1.61 | 0.06 | 0.86 | 0.03 | 2.2 |
| 3 | 3 | 0.98 | 4.67 | 2.95 | 0.11 | 2.35 | 0.06 | 1.9 |
| 3 | 4 | 6.47 | 10.30 | 5.65 | 0.26 | 5.44 | 0.13 | 1.8 |
| 4 | 2 | 0.73 | 3.34 | 2.98 | 0.09 | 1.62 | 0.04 | 1.9 |
| 4 | 3 | 6.50 | 10.20 | 6.33 | 0.26 | 5.42 | 0.13 | 1.8 |
| 5 | 2 | 2.56 | 5.77 | 4.89 | 0.14 | 2.99 | 0.07 | 1.8 |

4. Conclusions

In this paper, the multirate method was discussed in context with both linear and nonlinear systems. The results obtained from the small synchronous machine example for both the two and the three time scale example indicate that the multirate method holds great potential for being an efficient method for power system dynamic simulation. This is especially true in the case where a power system contains a small proportion of "fast" devices, such as DC lines, induction machines, or FACTS devices. The multirate method is extremely well suited for this type of system analysis.

5. Acknowledgments

The support of this project by NSF (grant ECS-9257208) is gratefully acknowledged.

References

- [1] J. Deuse and M. Stubbe, "Dynamic simulation of voltage collapse," to be published in *IEEE Transactions on Power Systems*, paper # 92 SM 396-2 PWRS.
- [2] A. Kurita, et. al. "Multiple time-scale power system dynamic simulation," *IEEE Transactions on Power Systems*, vol. 8, no. 1, pp. 216-223, February 1993.
- [3] Mariesa L. Crow, "Computational Algorithms for the Multirate Simulation of Electric Power System Dynamics". EPRI Report #TR-101795, *EPRI / NSF Workshop on application of advanced Mathematics to power systems*. pp 2-25-2-28, April 1993.
- [4] Mariesa L. Crow and M. Ilic, "The Parallel Implementation of the Waveform Relaxation Method for Transient Stability Simulations". *IEEE Transaction on Power systems*, Vol. PWRS-5, No.3, pp.922-933, August 1990.
- [5] P. W. Sauer, S. Ahmed-Zaid and P.V. Kokotovic, "An Integral Manifold Approach to Reduced Order Dynamic Modeling of Synchronous Machines". *IEEE Transactions on Power Systems*, Vol.3, No.1. pp.17-23, February 1988.