Theses and dissertations

1-1-2003

# Analysis of periodically switched nonlinear circuits and nonlinear oversampled sigma-delta modulators

Quan Li
*Ryerson University*

Recommended Citation

Li, Quan, "Analysis of periodically switched nonlinear circuits and nonlinear oversampled sigma-delta modulators" (2003). *Theses and dissertations.* Paper 20.

# Analysis of Periodically Switched Nonlinear Circuits and Nonlinear Oversampled Sigma-Delta Modulators

by

Quan Li

M.Eng, University of Electronic Science and Technology of China,
Chengdu, China, April 1988
B.Eng, University of Electronic Science and Technology of China,
Chengdu, China, July 1985

A thesis
presented to Ryerson University
in partial fulfillment of the
requirement for the degree of
Master of Applied Science
in the Program of
Electrical and Computer Engineering

Toronto, Ontario, Canada, 2003

National Library
of Canada

Bibliothèque nationale
du Canada

Acquisitions and
Bibliographic Services

Acquisisitons et
services bibliographiques

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

# Canadä

# Author's Declaration

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

# Instructions on Borrowers

Ryerson University requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

# Abstract

Analysis of Periodically Switched Nonlinear Circuits and

Nonlinear Oversampled Sigma-Delta Modulators

Quan Li

Master of Applied Science

Department of Electrical and Computer Engineering

Ryerson University

Toronto, Ontario, Canada, 2003

This thesis proposes a new method for time domain response and sensitivity analysis of periodically switched nonlinear circuits and nonlinear oversampled sigma-delta modulators. Using Volterra functional series and interpolating Fourier series, it extends the sampled data simulation and two-step algorithm of linear circuits to periodically switched nonlinear circuits. The method can handle the inconsistent initial conditions encountered at switching instants. It is applied to the analysis of nonlinear oversampled sigma-delta modulators. The efficiency and accuracy of the proposed method are assessed using SPICE, brute-force, and the law of charge conservation on periodically switched nonlinear circuits and nonlinear oversampled sigma-delta modulators. The method does not require the costly Newton-Raphson iterations. It is most efficient for analysis of circuits with mildly nonlinear characteristics and simulation over a long period of time.

# Acknowledgments

I wish to express my deep gratitude and sincere thanks to my supervisor, Professor Fei Yuan, for his constant support and valuable guidance throughout this research. He has shown such great interest and enthusiasm and provided a lot of helpful advice and fruitful discussions in my research work. All these make it possible for my research work to achieve this final accomplishment.

I am grateful to the School of Graduate Studies, Ryerson University for providing me with the Full Scholarship during my two years graduate studies at Ryerson.

Finally, I am pleased to thank my wife and my daughter for their love, patience, and understanding. This work could not have been completed without their support.

# Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1 Motivation

Switched circuits, such as switched capacitor and switched current networks, oversampled sigma-delta modulators, and switching power supplies, are widely encountered in telecommunication, instrumentation, and power electronics systems. Switched circuits distinguish themselves from time-invariant circuits by including switches that are either clocked externally or controlled internally. These circuits can be broadly categorized into two classes as follows:

1. Circuits with internally controlled switches

   Circuits in this category include rectifiers and switching voltage regulators that are widely encountered in power electronics systems. Switches in these circuits take the form of diodes and thyristors. Because the state of switches in these circuits is determined by the network variables associated with the switches, the switching instants are not known *a priori*, and can only be determined during simulation [1, 2].

2. Circuits with externally clocked switches

Typical examples in this category include switched capacitor networks and switched current networks where switches are implemented using either NMOS transistors or CMOS pairs. Unlike circuits with internally controlled switches, the state of the switches in these circuits is solely determined by the state of the external clock and is known *a priori*. These circuits are also known as periodically switched circuits, attributive to the periodical operation nature of the switches in these circuits.

This thesis deals with the circuit of the second category, periodically switched nonlinear circuits only.

Due to the aggressive reduction in both the feature size of MOS devices and the supply voltage of modern CMOS technologies, and the continuous increase in signal frequency, periodically switched circuits exhibit increasing nonlinear characteristics. These nonlinear characteristics include junction capacitances, nonlinear channel current of MOS devices due to velocity saturation and mobility degradation [3], finite slew rate of operational amplifiers, clock feed-through and charge injection [4], to name a few.

The methods both for time domain and frequency domain analysis of periodically switched linear circuits, such as those given in [5, 12], can not be employed to efficiently analyze the periodically switched nonlinear circuits. General-purpose analog simulators, such as PSPICE and Spectre [7], can be used to simulate the switched circuits in time domain. They compute the time-domain response using linear multi-step predictor-corrector (LMS-PC) algorithms [7, 8]. These algorithms are effective and robust in handling both mildly and harsh nonlinearities. However, they suffer from a number of drawbacks when analyzing periodically switched nonlinear circuits, particularly when switches are modeled as an ideal device. These drawbacks include:

1. Periodically switched circuits are dual-time systems that contain rapidly varying

clocks and slowly varying signals. They have to be simulated over a large number of clock cycles with fine steps in order to obtain their time domain characteristics.

2. LMS-PC algorithm fail to locate the exact time at which switches change state. Instead, switching time is approximated by a successive reduction in step size, giving rise to both poor simulation accuracy and excessive computation time [1].

3. Inability to handle inconsistent initial conditions caused by ideal switching [7]. Inconsistent initial conditions that occur at switching instants cause network variables to exhibit discontinuous characteristics that are difficult to be handled by LMS-PC algorithms.

4. Due to the complexity of the full model of switches, general-purpose analog simulators usually model switches as a voltage-controlled resistor with two largely distinct resistances for the ON- and OFF- states. Such a model often leads to stiff systems that require very fine steps to simulate.

5. General-purpose analog simulators rely on Newton-Raphson iterations in every step to find the correct solution. Not only the cost of Newton-Raphson is normally high, the step size used in the predictor of these algorithms has also to be kept small in order to provide a better starting point for Newton-Raphson iterations, subsequently faster converging.

6. The accuracy of general-purpose analog simulators is limited by the local truncation error (LTE) of the numerical methods. Due to stability constraints, the order of numerical integration used by general-purpose analog simulators is low. Most SPICE-like simulators employ only the first and second-order extrapolation algorithms (backward Euler and Trapezoidal). High accuracy can only be achieved at

the expense of CPU time.

The combined effects of periodic switching and nonlinear characteristics give rise to a large number of frequency components in the response of periodically switched nonlinear circuits. As a result, frequency domain analysis of these circuits, such as harmonic balance based methods, is not only complex but also computationally expensive.

In summary, new computational algorithms are required for time domain response and sensitivity analysis of periodically switched nonlinear circuits. The main attributes of new algorithms are accuracy and efficiency.

## 1.2    Objectives

Many circuits encountered in telecommunication systems have fixed DC operating points and the signals to be processed are usually of small amplitude. In this case, these circuits only exhibit mildly nonlinear characteristics [9]. When the characteristics of the nonlinearities are weakly nonlinear, Volterra functional series based methods [20, 21] can be employed to effectively simplify frequency analysis and reduce the cost of computation. Methods based on this approach have been used successfully for nonlinear analysis of power conversion systems [22], radio-frequency circuits [9], and general periodically switched nonlinear circuits [6] in frequency domain.

In this thesis, we extend Volterra functional series based nonlinear analysis approach from frequency domain to time domain, and present a new and efficient method for analysis of the time domain response and sensitivity of periodically switched circuits with mildly nonlinear characteristics. The method extends the sampled data simulation of linear circuits given in [5] to periodically switched nonlinear circuits. In addition, it extends the two-step algorithm for switched linear circuits given in [18] to periodically

4

switched nonlinear circuits. The proposed method handles general periodically switched nonlinear circuits including switched capacitor networks, switched current networks, and oversampled sigma-delta modulators.

## 1.3 Major Contributions

This thesis contains the original contributions for the time domain response and sensitivity analysis of periodically switched nonlinear circuits. The major contributions of this thesis are summarized as follows:

1. The sampled data simulation algorithm for linear circuits is extended to periodically switched nonlinear circuits. Time domain analysis of both response and sensitivity of periodically switched nonlinear circuits is performed efficiently by using the extended sampled data simulation.

2. The two-step algorithm for periodically switched linear circuits is extended to periodically switched nonlinear circuits. The inconsistent initial conditions encountered at switching instants are correctly handled by the extended two-step algorithm.

3. A new time domain sensitivity analysis method is proposed for periodically switched linear circuits. The discontinuity problems at switching instants are correctly handled for sensitivity analysis.

4. A new method for the time domain sensitivity analysis of periodically switched nonlinear circuits is proposed. Time domain sensitivity analysis is performed with response analysis efficiently and the discontinuity problems are correctly handled.

5. A new method for simulation of nonlinear oversampled sigma-delta modulators is proposed to include general nonlinearities encountered in circuit level simulation.

All primitive results of this thesis were published in two conference proceedings [32, 33] and on *IEEE Transactions on Circuits and Systems I - Fundamental Theory and Applications* [31].

## 1.4 Thesis Organization

Chapter 2 reviews the sampled data simulation algorithm for time domain response analysis of linear circuits and the two-step algorithm for handling the inconsistent initial conditions of switched linear circuits. The time domain analysis algorithm for periodically switched linear circuits is obtained based on these two useful algorithms. These algorithms are efficient and accurate. They form the foundation of the algorithms for response and sensitivity analysis of periodically switched nonlinear circuits in the following chapters.

Chapter 3 reviews the Volterra functional series based methods for the time domain analysis method of nonlinear circuits. A nonlinear example circuit is used to illustrate the usefulness of the method.

In chapter 4, a new efficient algorithm for time domain analysis of periodically switched nonlinear circuits is proposed. The algorithm is efficient for analysis of periodically switched nonlinear circuits because the new sampled data simulation method is used instead of the computationally intensive Newton-Raphson iterations in each integration step. At the switching instants, the proposed algorithm handles the inconsistent initial conditions for periodically switched nonlinear circuits correctly and efficiently using the newly proposed extended two-step algorithm.

In chapter 5, time domain sensitivity analysis methods for linear circuits [15] are reviewed. We then propose a new time domain sensitivity analysis method for periodically switched linear circuits. It can handle the discontinuities of sensitivity networks that might occur at switching instants.

In chapter 6, we propose a new method for time domain sensitivity analysis of periodically switched nonlinear circuits. Inside the clock phases, time domain sensitivity was obtained by summing up all the sensitivities of all the Volterra circuits of a nonlinear circuit. At switching instants, the new method can correctly solve the discontinuity problems that might occur at switching instants in sensitivity analysis of periodically switched nonlinear circuits.

Chapter 7 applies the computer algorithms proposed in this thesis to the analysis of nonlinear oversampled sigma-delta modulators. A new method is achieved to include general nonlinearities encountered in circuit level simulation of nonlinear oversampled sigma-delta modulators.

Chapter 8 summarizes the work of this thesis. It also discusses the directions for further research in some related areas.

Finally, methods for computing $\mathbf{M}(T)$, $\mathbf{P}(T)$, $\mathbf{P}_k(T)$ and their Derivatives are detailed in Appendix A.

# Chapter 2

# Time Domain Response of Periodically Switched Linear Circuits

In this chapter, we will focus on time domain response analysis of periodically switched linear circuits. First, in Section 2.1, we review the sampled date simulation algorithm for time domain response analysis of linear circuits proposed in [5]. For periodically switched linear circuits, discontinuity problems might occur at switching instants in their time domain response. It is important that the algorithms for time domain response analysis correctly handles the inconsistent initial conditions at switching instants. The two-step algorithm given in [18] for handling the inconsistent initial conditions is revisited in Section 2.2. A rigorous derivation of the two-step algorithm is also given in this section. Numerical examples are given in Section 2.3 to demonstrate the effectiveness of these algorithms. Finally, the chapter is summarized in Section 2.4.

## 2.1 Sampled Data Simulation of Linear Circuits

Consider a linear circuit with an input $w(t)$. The circuit in time domain is depicted by

$$\mathbf{G}\mathbf{v}(t) + \mathbf{C}\frac{d\mathbf{v}(t)}{dt} = \mathbf{d}w(t), \quad \mathbf{v}(t)|_{t=0^-} = \mathbf{v}(0^-), \tag{2.1}$$

where $\mathbf{v}(t)$ is the network variable vector, $\mathbf{G}$ and $\mathbf{C}$ are the conductance and capacitance matrices, respectively, and $\mathbf{d}$ is a constant vector specifying the nodes to which the input is connected. Using Laplace transform, we get

$$\mathbf{V}(s) = \mathbf{A}^{-1}(s)\mathbf{C}\mathbf{v}(0^-) + \mathbf{A}^{-1}(s)\mathbf{d}W(s), \tag{2.2}$$

where $\mathbf{A}(s) = \mathbf{G} + s\mathbf{C}$ is the system matrix in s-domain and $W(s)$ is the Laplace transform of $w(t)$. Without loss of generality, let the input be $w(t) = e^{j\omega_o t}$. Taking the inverse Laplace transform of (2.2), we obtain the time domain response of the circuit at $t = T$, where $T$ is the time step,

$$\mathbf{v}(T) = \mathbf{M}(T)\mathbf{v}(0^-) + \mathbf{P}(T),$$

where

$$\mathbf{M}(T) = \mathcal{L}^{-1}[\mathbf{A}^{-1}(s)\mathbf{C}]_{t=T},$$
$$\mathbf{P}(T) = \mathcal{L}^{-1}\Big[\mathbf{A}^{-1}(s)\frac{\mathbf{d}}{s - j\omega_o}\Big]_{t=T}, \tag{2.3}$$

and $\mathcal{L}^{-1}[.]$ is the inverse Laplace transform operator. Notice that $\mathbf{M}(T)$, the transition matrix, is independent of the input, and $\mathbf{P}(T)$, the zero-state vector, is input-dependent. Both of them are functions of $T$.

For lumped linear networks, any time point can be chosen as the time origin as long as an appropriate initial condition is taken into account. Resetting the time axis to zero

is equivalent to updating the initial condition vector $v(0^-)$ in equation (2.4) after each time step. Generally, to calculate the response at $t = nT + T$, the time origin should be shifted from $t = 0$ to $t = nT$. Subsequently, the input becomes $w(t) = e^{j\omega_o(nT+t)}$, the initial condition becomes $\mathbf{v}(nT^-)$, and the circuit is depicted by,

$$\mathbf{G}\mathbf{v}(t) + \mathbf{C}\frac{d\mathbf{v}(t)}{dt} = \mathbf{d}e^{j\omega_o(nT+t)}, \quad \mathbf{v}(t)|_{t=0^-} = \mathbf{v}(nT^-).$$

The time domain response can thereby be computed successively in a sampled data manner

$$\mathbf{v}(nT + T) = \mathbf{M}(T)\mathbf{v}(nT) + \mathbf{P}(T)e^{j\omega_o nT}. \tag{2.4}$$

If the time step $T$ is the same for each step, $\mathbf{M}(T)$ and $\mathbf{P}(T)$ are constant and need to be computed only once. They can be calculated using the numerical Laplace inversion technique, an A-stable high-order numerical integration method [16]. The computation required for evaluating (2.4) in each time step is only one matrix-vector multiplication and one vector addition. The response at equally spaced intervals of time can be computed efficiently. The method is known as sampled data simulation. Unlike ordinary integration methods, the time step $T$ of the sampled data simulation is independent of circuit time constant and has no effect on the accuracy of the simulation.

For a linear circuit with multiple exponential inputs $w_k(t) = e^{j\omega_{o,k}t}, k = 1, 2, \cdots, K$, the response can be obtained from superposition

$$\mathbf{v}(nT + T) = \mathbf{M}(T)\mathbf{v}(nT) + \sum_{k=1}^{K} \mathbf{P}_k(T)e^{j\omega_{o,k} nT}, \tag{2.5}$$

10

where $\mathbf{P}_k(T)$ is the zero-state vector for the input $w_k(t)$ and $K$ is the total number of inputs.

The use of complex exponential function as inputs is instrumental in obtaining the simple relation given by (2.5). One special case of interest is when $\omega_o = 0$, corresponding to a step input. Sinusoidal inputs can also be handled easily. For example, the response of the circuit to the input $w(t) = sin(\omega_o t)$ is given by

$$\mathbf{v}(nT + T) = \mathbf{M}(T)\mathbf{v}(nT) + \mathcal{I}m[\mathbf{P}(T)e^{j\omega_o nT}], \tag{2.6}$$

where $\mathcal{I}m[.]$ denotes the imaginary part of a complex argument. Similarly, the response to the input $w(t) = cos(\omega_o t)$ is obtained by replacing $\mathcal{I}m[.]$ with $\mathcal{R}e[.]$ in (2.6), where $\mathcal{R}e[.]$ denotes the real part.

## 2.2 Inconsistent Initial Conditions of Periodically Switched Linear Circuits

Switched circuits distinguish themselves from time-invariant circuits by including switches. These switches appear physically in the form of NMOS transistors, transmission gates, diodes, etc. They can be characterized at various levels of abstraction, depending upon the accuracy requirements. Among which, the full-transistor model, voltage-controlled resistor model, and ideal switch model are the most often used. The full-transistor model captures the rapidly changing transient characteristics of the time-domain behavior and high-frequency behavior of the circuits. Simulation based on the full-transistor model, however, is time consuming. The use of this model can only be warranted if a detailed analysis is required. When the rapidly changing transient characteristics of the time-domain behavior and high-frequency behavior of these circuits are not of a critical

11

concern, switches can be modeled as a voltage-controlled resistor that has two largely distinct resistances with a small resistance in the ON state and a large resistance in the OFF state. The large ratio between the ON and OFF resistance values often results in stiff systems that are difficult to simulate. To avoid this drawback, the ideal switch model that has zero resistance in the ON-state and infinite resistance in the OFF-state can be employed. The difference between these two simplified models only manifests in the transient response of circuits [12]. This transient portion requires many small steps to control the error, and subsequently excessive CPU time to simulate, usually up to 90 % of total simulation time for the switch [10]. If the initial transient portion of the response is not of a critical concern, it is advantageous computationally to model switches as ideal devices. Ideal switching, however, may cause an abrupt change in nodal voltages and gives rise to inconsistent initial conditions. The calculation of inconsistent initial conditions is essentially important for analyzing switched networks with ideal switches because at a switching instant $t = t_0$, a Dirac impulse current or voltage may occur and the network variables become discontinuous. As a result, the initial conditions immediately before switching ($t = t_0^-$) and after switching ($t = t_0^+$) may be different [18].

The two-step method that is based on numerical Laplace inversion was proposed in [18] to correctly handle Dirac impulses and inconsistent initial conditions of switched linear circuits. This is because computation is performed in the frequency domain where Dirac impulses and inconsistent initial conditions can be conveniently represented.

Incorporating the two-step method into the preceding sampled data simulation, the inconsistent initial conditions of periodically switched linear circuits can be handled. Assume a switching occurs at $t = nT$ and the initial condition before switching is given by $\mathbf{v}(nT^-)$. The forward step from $t = nT^-$ to $t = nT^- + T$ calculates $\mathbf{v}(nT^- + T)$

$$\mathbf{v}(nT^- + T) = \mathbf{M}(T)\mathbf{v}(nT^-) + \mathbf{P}(T)e^{j\omega_o nT}. \tag{2.7}$$

The step size $T$ should be chosen such that no switching occurs at $t = nT + T$ and the response is continuous, i.e. $\mathbf{v}(nT + T) = \mathbf{v}(nT^- + T) = \mathbf{v}(nT^+ + T)$. Following the forward step, an identical step size is taken from $t = nT^+ + T$ to $t = nT^+$ to compute the response immediately after switching $\mathbf{v}(nT^+)$. Reset the time origin to $t = nT + T$. The initial condition is given by $\mathbf{v}(nT^+ + T)$ and the input becomes $w(t) = e^{j\omega_o[(nT+T)-t]}$. Laplace transform, however, can not be directly used in the backward step because Laplace integral is from $t = 0^-$ to $t = +\infty$. This can be solved by employing the transform

$$\tau = -t.$$

In $\tau$-domain the origin is $\tau = -(nT + T)$, the initial condition is still $\mathbf{v}(nT^+ + T)$, but the input becomes $w(\tau) = e^{j\omega_o[-(nT+T)+\tau]}$, as shown in Fig.2.1. Rewrite (2.4) for the backward step in $\tau$-domain

$$\mathbf{G}\mathbf{v}(\tau) + \mathbf{C}\frac{d\mathbf{v}(\tau)}{d\tau} = \mathbf{d}w(\tau), \quad \mathbf{v}(\tau)|_{\tau=-(nT^++T)} = \mathbf{v}(nT^+ + T).$$

Using Laplace transform for the backward step, we obtain the response immediately after switching

$$\mathbf{v}(nT^+) = \mathbf{M}_B(T)\mathbf{v}(nT^+ + T) + \mathbf{P}_B(T)e^{-j\omega_o(nT+T)}, \tag{2.8}$$

where

13
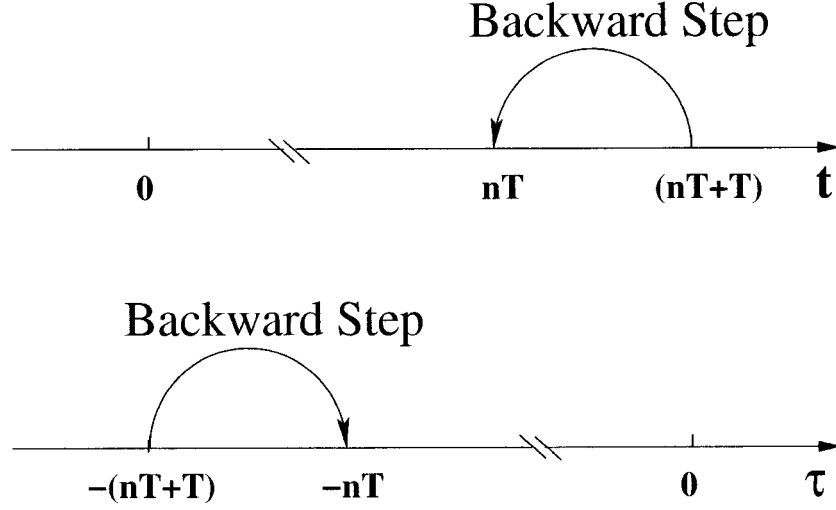
**Figure 2.1:** Transform $\tau = -t$ for backward step

$$\mathbf{M}_B(T) = \mathcal{L}^{-1}[\mathbf{A}^{-1}(s)\mathbf{C}]_{\tau=T}$$

$$= \mathcal{L}^{-1}\Big[\mathbf{A}^{-1}(s)\mathbf{C}\Big]_{t=-T}$$

$$= \mathbf{M}(-T),$$

$$\mathbf{P}_B(T) = \mathcal{L}^{-1}\Big[\mathbf{A}^{-1}(s)\frac{\mathbf{d}}{s - j\omega_o}\Big]_{\tau=T}$$

$$= \mathcal{L}^{-1}\Big[\mathbf{A}^{-1}(s)\frac{\mathbf{d}}{s - j\omega_o}\Big]_{t=-T}$$

$$= \mathbf{P}(-T). \tag{2.9}$$

Similar to $\mathbf{M}(T)$ and $\mathbf{P}(T)$, if the step size $T$ is not changed, $\mathbf{M}_B(T)$ and $\mathbf{P}_B(T)$ are constant and need to be computed only once for a given network topology. Fig.2.2 provides a graphical view of the inconsistent initial conditions and the two-step algorithm.

The time domain response of periodically switched linear circuits can be solved, using the algorithms described in this section. Inside each clock phase, the topology of the circuit is not changed and the circuit is a linear time invariant circuit whose response can be computed using the sampled data simulation presented in this chapter. At the

14
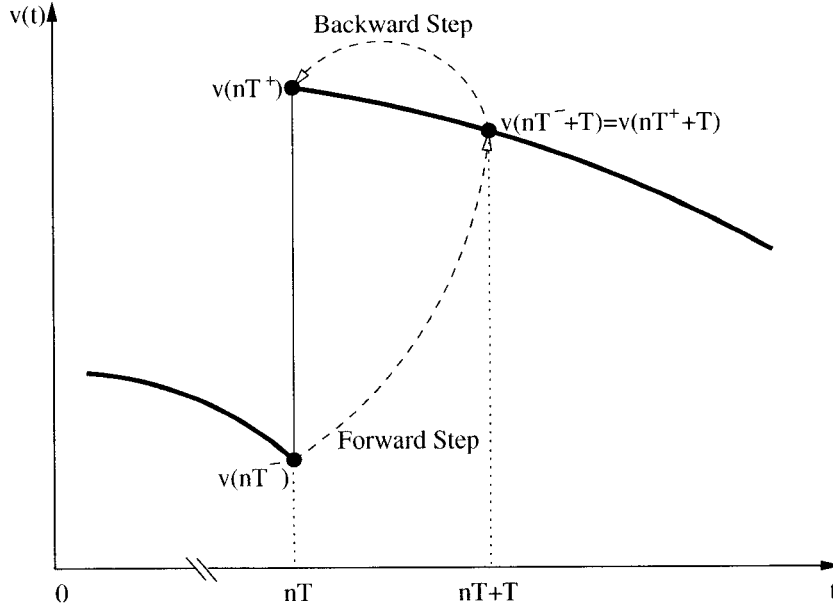
**Figure 2.2:** Inconsistent initial conditions and two-step algorithm

switching instants, inconsistent initial conditions can be handled using the two-step algorithm.

## 2.3 Numerical Examples

To verify the usefulness of the algorithms presented in the preceding sections, a computer program has been written in Matlab. The algorithms have been implemented in the computer program. Numerical examples of periodically switched linear circuits with different forms of input are simulated. Simulation was performed on Sun Ultra-10 workstations with 440MHz CPU and 512MB RAM.

### 2.3.1 Non-inverting Switched-Capacitor Integrator with Unit Step Input

The first example is a non-inverting switched-capacitor integrator shown in Fig.2.3. It contains four externally clocked ideal switches. The clock phases are non-overlapping.

15
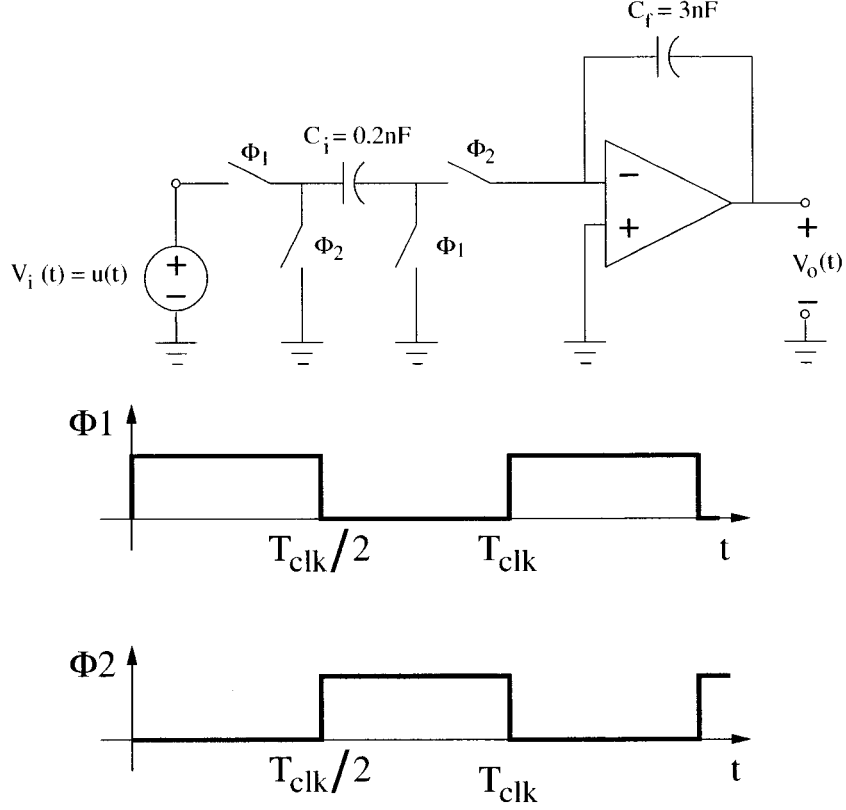
**Figure 2.3:** A non-inverting switched-capacitor integrator

The clock frequency is 1 MHz. Each clock period has two phases of equal width. $C_i = 0.2nF$ and $C_f = 3nF$. Zero initial conditions are assumed for all capacitors. The operational amplifier is assumed to be ideal. The input is a unit step voltage source, $V_i(t) = u(t)$.

The output voltage of the non-inverting switched-capacitor integrator, $V_o(t)$, is computed and plotted in Fig.2.4. As shown in the figure, it is discontinuous at $t = 0.5\mu s, 1.5\mu s, 2.5\mu s, ...$, where inconsistent initial conditions are encountered. It is continuous at $t = 1.0\mu s, 2.0\mu s, 3.0\mu s, ...$, where only consistent initial conditions are encountered. This observation demonstrates that algorithms can correctly handle both the inconsistent and consistent initial conditions for periodically switched linear circuits.

16
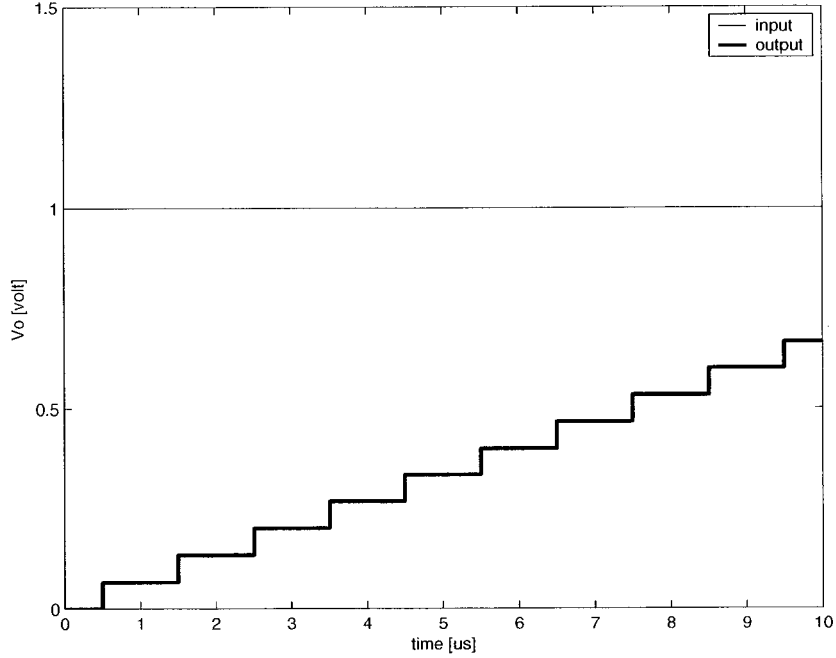
**Figure 2.4:** Response of non-inverting switched-capacitor integrator

## 2.3.2 Inverting Switched-Capacitor Integrator with Sinusoidal Input

The second example is an inverting switched-capacitor integrator, as shown in Fig 2.5. The clock frequency is 100 kHz. Each clock period has two non-overlapping phases with equal width. $C_i = 0.1nF$ and $C_f = 1nF$. Zero initial conditions are assumed for all capacitors. The operational amplifier is assumed to be ideal. The input to the integrator is a voltage source of 1 kHz cosine wave with $0.1V$ amplitude,

$$V_i(t) = 0.1\cos(2000\pi t)V.$$

The time domain response of the switched-capacitor integrator is simulated and its output voltage, $V_o(t)$, is plotted as shown in Fig. 2.6.

17
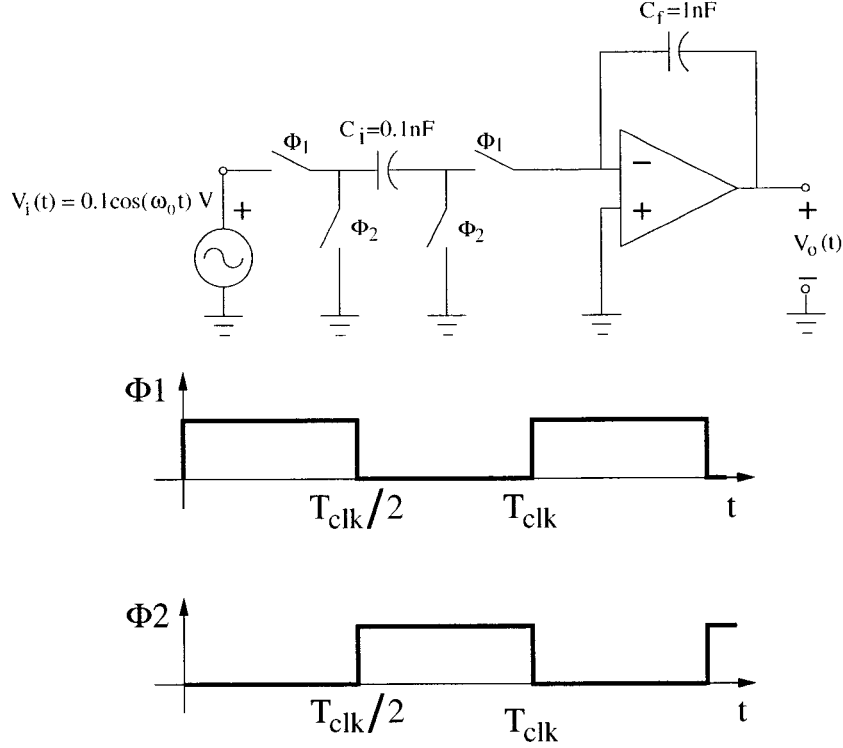
**Figure 2.5:** An inverting switched-capacitor integrator

## 2.4 Summary

Two useful simulation algorithms, sampled data simulation for analyzing the time domain response of linear circuits and two-step algorithm for handling the inconsistent initial conditions of switched circuits, have been reviewed in this chapter. Combining these two algorithms, we have obtained the time domain analysis algorithm for periodically switched linear circuits. These algorithms are efficient, accurate, and stable for the time domain analysis of periodically switched linear circuits. Numerical examples show the usefulness of the algorithms. We will extend these algorithms to response and sensitivity analysis of periodically switched nonlinear circuits in the following chapters.
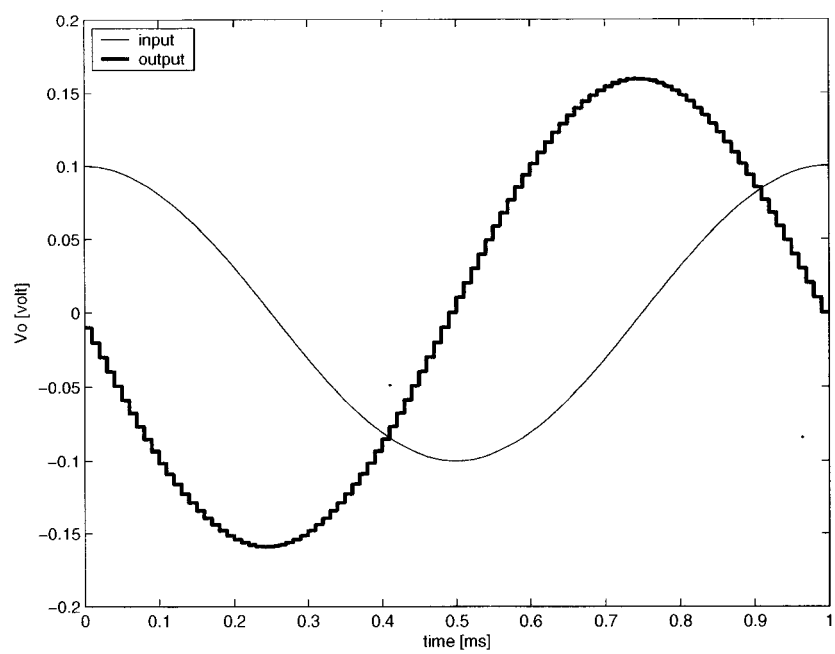
**Figure 2.6:** Response of inverting switched-capacitor integrator

# Chapter 3

# Volterra Series Representation of Nonlinear Circuits

This chapter deals with the time domain analysis method of nonlinear circuits. A brief introduction of the Volterra functional series based methods for analysis of nonlinear circuits is given in Section 3.1. The Volterra circuits of nonlinear circuits are derived in Section 3.2. In Section 3.3, a example circuit is used to illustrate the method. Practical considerations for the use of the method in analysis of nonlinear circuits are given in Section 3.4.

## 3.1  Introduction

Nonlinearities are widely encountered in analysis and design of integrated circuits. Nonlinear characteristics typically encountered in integrated circuits include the exponential I-V characteristics of diodes and transistors, the square-law characteristics of the drain current of MOSFETs, pn-junction capacitances, and other types of nonlinearities which can be modeled using polynomials. These nonlinear characteristics are usually continuously differentiable and can be represented sufficiently by a finite number of their Taylor series expansions at their DC operating point. When the input is small, the

nonlinearities in this case are usually mild [9].

Nonlinear circuits are most often analyzed in time domain using linear multi-step predictor-corrector (LMS-PC) algorithms where Newton-Raphson iterations are required [7, 8]. These methods are robust and effective in handling circuits with both harsh and mild nonlinearities. They are, however, computationally expensive mainly due to the need for Newton-Raphson iterations in every step of integration.

Volterra functional series based methods [20, 21] can be employed to effectively simplify the analysis and reduce the cost of computation. In frequency domain, methods based on this approach have been used successfully for nonlinear analysis of power conversion systems [22], radio-frequency circuits [9], and general periodically switched nonlinear circuits [6]. Volterra functional series based nonlinear analysis approach was also extended successfully from frequency domain to time domain [11]. It was shown that, using Volterra functional series, a nonlinear element can be characterized equivalently as a linear element, along with a set of independent sources in which the nonlinear characteristics are embedded. Consequently, the nonlinear circuit can be represented equivalently by a set of linear circuits called Volterra circuits. The time domain solutions of a nonlinear circuit are obtained by solving the Volterra circuits.

## 3.2   Volterra Circuits

For circuits with mildly nonlinear elements, we characterize the behavior of the nonlinear elements using a finite number of terms of their Taylor series expansion at their DC operating point. Using modified nodal analysis, the nonlinear circuit is depicted by

$$\mathbf{G}\mathbf{v}(t) + \mathbf{C}\frac{d\mathbf{v}(t)}{dt} = \mathbf{d}w(t) + f[\mathbf{v}(t)], \quad \mathbf{v}(t)|_{t=0^-} = \mathbf{v}(0^-), \tag{3.1}$$

where **G** and **C** are the conductance and capacitance matrices respectively. Their entries are made of the coefficients of the linear elements and the first-order term of the Taylor series expansion of the nonlinear characteristics. The higher order terms of the nonlinear characteristics are embedded in the nonlinear function $f[\mathbf{v}(t)]$.

In [20] and [23], it was shown that the response of a nonlinear system $y(t)$ to input $x(t)$ can be represented by the Volterra functional series

$$y(t) = \sum_{m=1}^{\infty} y_m(t),$$

where

$$y_m(t) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h_m(t, \tau_1, \cdots, \tau_m) x(\tau_1) \cdots x(\tau_m) d\tau_1 \cdots d\tau_m,$$

$y_m(t)$ is the $m$th-order term of the Volterra series expansion of $y(t)$ and $h_m(t, \tau_1, \cdots, \tau_m)$ is the $m$th-order Volterra kernel. When the input is changed from $x(t)$ to $\epsilon x(t)$, where $\epsilon$ is a nonzero constant, the response becomes

$$y(t) = \sum_{m=1}^{\infty} y_m(t)\epsilon^m. \tag{3.2}$$

Eq. (3.2) indicates that $y(t)$ is a polynomial in $\epsilon$. Expressing $\mathbf{v}(t)$ using (3.2), substituting the input $\epsilon w(t)$ into (3.1), and equating the terms of the same order in $\epsilon$ lead to the following set of linear differential equations

$$\mathbf{G}\mathbf{v}_1(t) + \mathbf{C}\frac{d\mathbf{v}_1(t)}{dt} = \mathbf{d}_1 f_1(t), \quad \mathbf{v}_1(t)|_{t=0^-} = \mathbf{v}_1(0^-),$$

$$\mathbf{G}\mathbf{v}_2(t) + \mathbf{C}\frac{d\mathbf{v}_2(t)}{dt} = \mathbf{d}_2 f_2[\mathbf{v}_1(t)], \quad \mathbf{v}_2(t)|_{t=0^-} = \mathbf{v}_2(0^-),$$

$$\vdots$$

$$\mathbf{G}\mathbf{v}_m(t) + \mathbf{C}\frac{d\mathbf{v}_m(t)}{dt} = \mathbf{d}_m f_m[\mathbf{v}_1(t), \mathbf{v}_2(t), \cdots, \mathbf{v}_{m-1}(t)], \quad \mathbf{v}_m(t)|_{t=0^-} = \mathbf{v}_m(0^-),$$

$$\vdots \tag{3.3}$$

where $\mathbf{v}_m(t)$ is the $m$-th order term of the Volterra series expansion of $\mathbf{v}(t)$, $\mathbf{d}_m$ is a constant vector specifying the node to which the input $f_m[\mathbf{v}_1(t), \mathbf{v}_2(t), \cdots, \mathbf{v}_{m-1}(t)]$ is connected. The circuit depicted by the $m$-th equation in (3.3) is called Volterra circuit of order $m$.

For the first-order Volterra circuit, its input given by $f_1(t) = w(t)$ and its vector $\mathbf{d}_1 = \mathbf{d}$, which are identical to the input and the vector $\mathbf{d}$ for the original nonlinear circuit, respectively. The input for higher order Volterra circuits $f_m[\mathbf{v}_1(t), \mathbf{v}_2(t), \cdots, \mathbf{v}_{m-1}(t)], m = 2, 3, \cdots$, is a nonlinear function of the solutions of lower order Volterra circuits. Note that for the second and higher order Volterra circuits, the inputs are connected to the same node. So the vectors $\mathbf{d}_m, m = 2, 3, \cdots$, are identical for higher Volterra circuits, i.e.

$$\mathbf{d}_2 = \mathbf{d}_3 = \cdots = \mathbf{d}_m = \cdots.$$

Each Volterra circuit is a linear time invariant circuit and can be solved using the sampled data simulation given in Chapter 2 for both efficiency and accuracy. The time domain solution of the nonlinear circuit is obtained by summing up those of Volterra circuits

$$\mathbf{v}(t) = \sum_{m=1}^{\infty} \mathbf{v}_m(t). \tag{3.4}$$

23

## 3.3 Examples

In this section, we use a simple nonlinear circuit shown in Fig. 3.1 to illustrate the method described in Section 3.2 and to show how the Volterra circuits of a nonlinear circuit are derived. In this example, the nonlinear conductor is characterized by

$$i(t) = g_1 v(t) + g_2 v^2(t) + g_3 v^3(t),$$

where $g_1, g_2$, and $g_3$ are constants. Using KCL we get

$$Gv(t) + [g_1 v(t) + g_2 v^2(t) + g_3 v^3(t)] = w(t). \tag{3.5}$$

Substituting Volterra series expansion

$$v(t) = v_1(t)\epsilon + v_2(t)\epsilon^2 + v_3(t)\epsilon^3$$

and input $\epsilon w(t)$ into (3.5), and equating the terms of the same order in $\epsilon$, we obtain

$$\begin{cases} (G + g_1)v_1(t) = w(t) \\ (G + g_1)v_2(t) = f_2[v_1(t)] \\ (G + g_1)v_3(t) = f_3[v_1(t), v_2(t)], \end{cases}$$

where

$$\begin{cases} f_2[v_1(t)] = -g_2 v_1^2(t) \\ f_3[v_1(t), v_2(t)] = -[2g_2 v_1(t)v_2(t) + g_3 v_1^3(t)]. \end{cases} \tag{3.6}$$

The corresponding Volterra circuits of the nonlinear circuit are shown in Fig. 3.1. It is seen that the input of the first-order Volterra circuit, $w(t)$, is identical to the input of

24

the original nonlinear circuit. The input of the second-order Volterra circuit, $f_2[v_1(t)]$, is a nonlinear function of the solution of the first-order Volterra circuit, $v_1(t)$. The input of the third-order Volterra circuit, $f_3[v_1(t), v_2(t)]$, is a nonlinear function of the solutions of the first-order and the second-order Volterra circuits, $v_1(t)$ and $v_2(t)$.

## 3.4  Practical Considerations

For circuits with mildly nonlinear elements, the Volterra functional series based methods are most effective and computationally efficient. This is because only low order Volterra series expansion is needed. For circuits with harsh nonlinearities, such as comparators and Schmitt triggers, a large number of Volterra circuits are required in order to provide good accuracy. Implementation in this case becomes formidably complex and the cost of computation also becomes excessive. In practice, usually maximum up to the 5th-order Volterra series expansion is used. In the research of this thesis, up to the 3rd-order Volterra series expansion is used.

The order of Volterra series expansion can be determined with reference to that of the Taylor series expansion of nonlinear elements. For a given nonlinear function $f(v)$, its Taylor series expansion at the operating point $v_0$ is

$$f(v) = f(v_0) + f^{'}(v_0)\Delta v + \frac{f^{''}(v_0)}{2}(\Delta v)^2 + \cdots + \frac{f^{(n)}(v_0)}{n!}(\Delta v)^n + \cdots,$$

where

$$\Delta v = v - v_0.$$

The $n$th-order truncation error of its Taylor series expansion bounded by

25

$$\left| \frac{f^{(n+1)}(\xi \Delta v)}{(n+1)!} (\Delta v)^{n+1} \right|,$$

where $0 < \xi < 1$. It can be determined conveniently.

As an example, consider the nonlinear conductor

$$i(t) = g_1 v(t) + g_2 v^2(t) + g_3 v^3(t).$$

The corresponding Volterra circuits are given by

$$\begin{cases} i_1(t) = g_1 v_1(t) \\ i_2(t) = g_1 v_2(t) + g_2 v_1^2(t) \\ i_3(t) = g_1 v_3(t) + 2g_2 v_1(t)v_2(t) + g_3 v_1^3(t) \\ i_4(t) = g_1 v_4(t) + 2g_2 v_1(t)v_3(t) + g_2 v_2^2(t) + 3g_3 v_1^2(t)v_2(t) \\ \vdots \end{cases} \tag{3.7}$$

The difference between employing the 3rd-order and the 4th-order Volterra series expansions is the last equation in Eq. (3.7). If $i_4(t) \ll i_1(t), i_2(t), i_3(t)$, the 3rd-order Volterra series expansion is considered to be sufficient. The last equation in (3.7) provides a practical measure to quantify the error due to the truncation of Volterra series expansion.

$i(t) = g_1 v(t) + g_2 v^2(t) + g_3 v^3(t)$

Nonlinear circuit

$f_2[v_1(t)] = -g_2 v_1^2(t)$
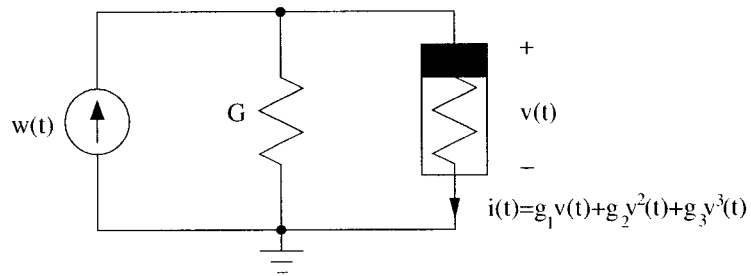
1st-order Volterra circuit
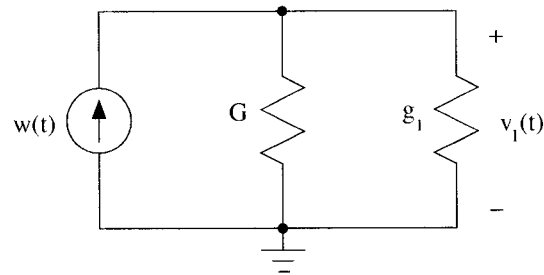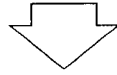
$f_3[v_1(t), v_2(t)] =$
$-[2g_2 v_1(t)v_2(t) + g_3 v_1^3(t)]$
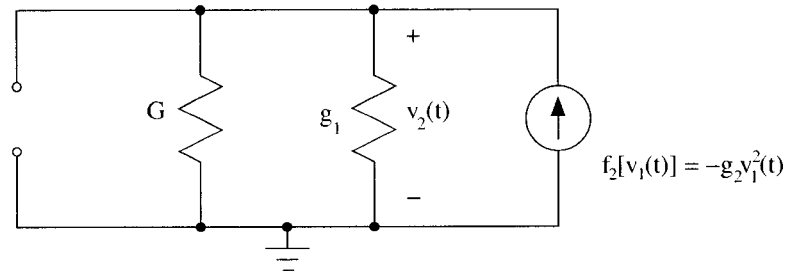
2nd-order Volterra circuit
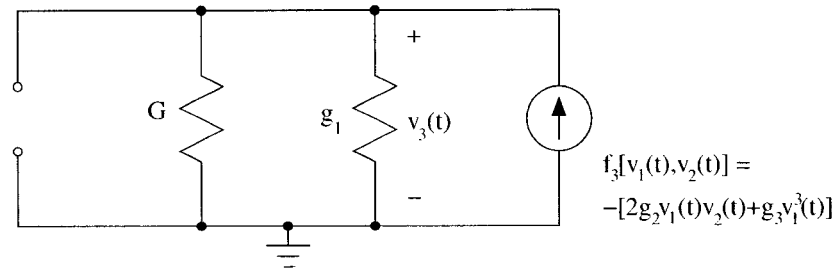
3rd-order Volterra circuit

**Figure 3.1:** Nonlinear circuit and its Volterra circuits

27

# Chapter 4

# Time Domain Response of Periodically Switched Nonlinear Circuits

This chapter proposes a new simulation algorithm for time domain analysis of periodically switched nonlinear circuits. The algorithm is implemented in a computer program and assessed by comparing the results of numerical examples from our program with those from PSPICE. Furthermore, the algorithm has the ability to correctly handle the inconsistent initial conditions at switching instants for periodically switched nonlinear circuits, which is verified by checking the charge conservation of the example circuits immediately before and after switching. The primitive results of this chapter were reported in [32], [33], and[34].

In this chapter, the problem of simulating periodically switched nonlinear circuits in the time domain is divided into two parts. First, the algorithm for analysis of nonlinear circuits inside a clock phase is presented in Section 4.1. Secondly, a new extended two-step algorithm is proposed in Section 4.2 to handle the inconsistent initial conditions at switching instants for periodically switched nonlinear circuits. The usefulness and accuracy of the proposed algorithm are examined using numerical examples in Section

4.3. Finally, the chapter is summarized in Section 4.4.

## 4.1 Time Domain Response of Nonlinear Circuits

As shown in Chapter 3, a nonlinear circuit with input $w(t) = e^{j\omega_o t}$ can be represented using its Volterra circuits as follows:

$$
\begin{aligned}
\mathbf{G}\mathbf{v}_1(t) + \mathbf{C}\frac{d\mathbf{v}_1(t)}{dt} &= \mathbf{d}_1 e^{j\omega_o t} \quad \mathbf{v}_1(t)|_{t=0^-} = \mathbf{v}_1(0^-), \\
\mathbf{G}\mathbf{v}_2(t) + \mathbf{C}\frac{d\mathbf{v}_2(t)}{dt} &= \mathbf{d}_2 f_2[\mathbf{v}_1(t)], \quad \mathbf{v}_2(t)|_{t=0^-} = \mathbf{v}_2(0^-), \\
&\vdots \\
\mathbf{G}\mathbf{v}_m(t) + \mathbf{C}\frac{d\mathbf{v}_m(t)}{dt} &= \mathbf{d}_m f_m[\mathbf{v}_1(t), \mathbf{v}_2(t), \cdots, \mathbf{v}_{m-1}(t)], \quad \mathbf{v}_m(t)|_{t=0^-} = \mathbf{v}_m(0^-), \\
&\vdots
\end{aligned}
\tag{4.1}
$$

Applying the sampled data simulation to each Volterra circuit and summing up the responses of all Volterra circuits, the time domain response of the nonlinear circuits is obtained. This leads to a new sampled data simulation method for nonlinear circuits.

The response of the first-order Volterra circuit in Eq. (4.1) is obtained from Eq. (2.4),

$$
\mathbf{v}_1(nT + T) = \mathbf{M}(T)\mathbf{v}_1(nT) + \mathbf{P}(T)e^{j\omega_o nT}.
\tag{4.2}
$$

To solve the second-order Volterra circuit, we need to know its input function $f_2(t)$. From the solution of the first-order Volterra circuit at discrete time points $\mathbf{v}_1(nT)$, the data of $f_2(t)$ at the discrete time points $f_2(nT) = f_2[\mathbf{v}_1(nT)]$ is obtained. $f_2(t)$ can be approximated subsequently using an interpolating Fourier series [11] that interpolates $f_2(nT)$. Interpolating Fourier series provides superior computational accuracy and numerical stability as compared with other interpolation methods, such as Lagrange or
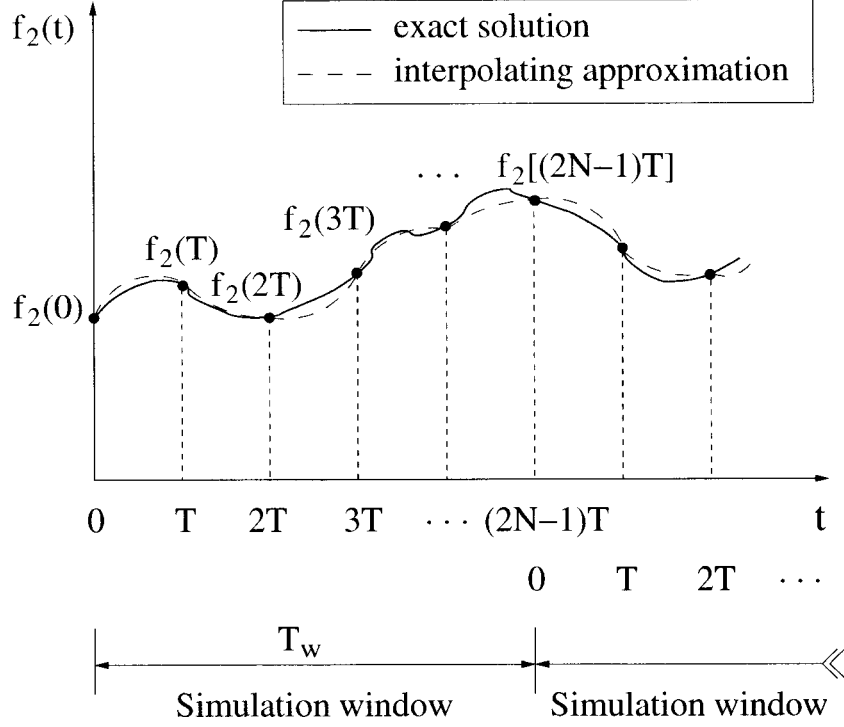
**Figure 4.1:** Simulation window

Newton [25]. To implement Fourier series interpolation, we choose a simulation window $[0, (2N - 1)T]$ with width $T_W$, as shown in Fig. 4.1. It contains $2N$ equally spaced data, $f_2(0), f_2(T), \cdots, f_2[(2N - 1)T]$. The input function $f_2(t)$ for $t \in [0, (2N - 1)T]$ is obtained from

$$f_2(t) \approx \frac{a_{2,0}}{2} + \sum_{k=1}^{N-1} \left[ a_{2,k}\cos(k\omega_s t) + b_{2,k}\sin(k\omega_s t) \right] + \frac{a_{2,N}}{2}\cos(N\omega_s t), \qquad (4.3)$$

where $\omega_s = \dfrac{2\pi}{T_w}$. The coefficients $[\mathbf{a_2}]$ and $[\mathbf{b_2}]$ are determined from

$$[\mathbf{a_2}] = \begin{bmatrix} a_{2,0} \\ a_{2,1} \\ \vdots \\ a_{2,N} \end{bmatrix}$$

30

$$= \mathbf{Q}(N,T) \begin{bmatrix} f_2(0) \\ f_2(T) \\ \vdots \\ f_2[(2N-1)T] \end{bmatrix},$$

$$[\mathbf{b_2}] = \begin{bmatrix} b_{2,1} \\ b_{2,2} \\ \vdots \\ b_{2,N-1} \end{bmatrix}$$

$$= \mathbf{R}(N,T) \begin{bmatrix} f_2(T) \\ f_2(2T) \\ \vdots \\ f_2[(2N-1)T] \end{bmatrix},$$

where

$$\mathbf{Q}(N,T) = \frac{1}{N} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & \cos(\omega_s T) & \cdots & \cos[(2N-1)\omega_s T)] \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cos(N\omega_s T) & \cdots & \cos[N(2N-1)\omega_s T)] \end{bmatrix},$$

$$\mathbf{R}(N,T) = \frac{1}{N} \begin{bmatrix} \sin(\omega_s T) & \sin(2\omega_s T) & \cdots & \sin[(2N-1)\omega_s T)] \\ \sin(2\omega_s T) & \sin(4\omega_s T) & \cdots & \sin[2(2N-1)\omega_s T)] \\ \vdots & \vdots & \ddots & \vdots \\ \sin[(N-1)\omega_s T] & \sin[2(N-1)\omega_s T] & \cdots & \sin[(N-1)(2N-1)\omega_s T] \end{bmatrix}.$$

Neglecting the interpolation error, the behavior of the second-order Volterra circuit in the simulation window is depicted by

$$\mathbf{G}\mathbf{v}_2(t) + \mathbf{C}\frac{d\mathbf{v}_2(t)}{dt} = \left\{ \frac{a_{2,0}}{2} + \frac{a_{2,N}}{2}\cos(N\omega_s t) + \sum_{k=1}^{N-1} \left[ a_{2,k}\cos(k\omega_s t) + b_{2,k}\sin(k\omega_s t) \right] \right\} \mathbf{d}_2.$$

It can be seen that the second-order Volterra circuit is a linear circuit with $2N$ inputs. Its response is obtained using the sampled data simulation and superposition

$$
\mathbf{v}_2(nT + T) = \mathbf{M}(T)\mathbf{v}_2(nT) + \mathcal{R}e\left\{\frac{a_{2,0}}{2}\mathbf{P}_0(T) + \frac{a_{2,N}}{2}\mathbf{P}_N(T)e^{jN\omega_s nT}\right.
$$

$$
\left. + \sum_{k=1}^{N-1} a_{2,k}\mathbf{P}_k(T)e^{jk\omega_s nT}\right\} + \mathcal{I}m\left\{\sum_{k=1}^{N-1} b_{2,k}\mathbf{P}_k(T)e^{jk\omega_s nT}\right\}, \quad (4.4)
$$

where

$$
\mathbf{P}_k(T) = \mathcal{L}^{-1}\left[\mathbf{A}^{-1}(s)\frac{\mathbf{d}_2}{s - jk\omega_s}\right]_{t=T},
$$

$$
k = 0, 1, \cdots, N,
$$

is the zero-state vector of the circuit to the input $w_k(t) = e^{jk\omega_s t}$. If the width of the window $T_W$ and the number of samples per window $2N$ are unchanged, $\mathbf{P}_k(T), k = 0, 1, \cdots, N$, will be the same for all Volterra circuits. For each order of Volterra circuit, only one new set of coefficients $[\mathbf{a}_m]$ and $[\mathbf{b}_m]$ need to be computed. They can be obtained efficiently using

$$
[\mathbf{a}_m] = \begin{bmatrix} a_{m,0} \\ a_{m,1} \\ \vdots \\ a_{m,N} \end{bmatrix}
$$

$$
= \mathbf{Q}(N, T)\begin{bmatrix} f_m(0) \\ f_m(T) \\ \vdots \\ f_m[(2N-1)T] \end{bmatrix}, \quad (4.5)
$$

$$
[\mathbf{b}_m] = \begin{bmatrix} b_{m,1} \\ b_{m,2} \\ \vdots \\ b_{m,N-1} \end{bmatrix}
$$

32

$$= \mathbf{R}(N,T) \begin{bmatrix} f_m(T) \\ f_m(2T) \\ \vdots \\ f_m[(2N-1)T] \end{bmatrix}. \tag{4.6}$$

Similarly, the behavior of the $m$th-order Volterra circuit in the simulation window is depicted by

$$\mathbf{G}\mathbf{v}_m(t) + \mathbf{C}\frac{d\mathbf{v}_m(t)}{dt} = \left\{ \frac{a_{m,0}}{2} + \frac{a_{m,N}}{2}\cos(N\omega_s t) \right.$$
$$\left. + \sum_{k=1}^{N-1}\Big[a_{m,k}\cos(k\omega_s t) + b_{m,k}\sin(k\omega_s t)\Big] \right\}\mathbf{d}_2$$
$$m = 2, 3, \cdots.$$

The response of the $m$th-order Volterra circuits in the simulation window can be computed in a similar manner

$$\mathbf{v}_m(nT+T) = \mathbf{M}(T)\mathbf{v}_m(nT) + \mathcal{R}e\left\{ \frac{a_{m,0}}{2}\mathbf{P}_0(T) + \frac{a_{m,N}}{2}\mathbf{P}_N(T)e^{jN\omega_s nT} \right.$$
$$\left. + \sum_{k=1}^{N-1} a_{m,k}\mathbf{P}_k(T)e^{jk\omega_s nT} \right\} + \mathcal{I}m\left\{ \sum_{k=1}^{N-1} b_{m,k}\mathbf{P}_k(T)e^{jk\omega_s nT} \right\}$$
$$m = 2, 3, \cdots. \tag{4.7}$$

The complete response of the nonlinear circuit in the simulation window is obtained by summing up the response of all Volterra circuits

$$\mathbf{v}(nT) = \sum_{m=1}^{\infty} \mathbf{v}_m(nT).$$

For each simulation window, because $\mathbf{M}(T), \mathbf{P}_k(T), k = 0, 1, \cdots, N, \mathbf{Q}(N,T)$, and $\mathbf{R}(N,T)$ are constant, the only computation required is to obtain the coefficients of the interpolating Fourier series $[\mathbf{a}_m]$ and $[\mathbf{b}_m]$. The response of the nonlinear circuit at equally

33

spaced time points can therefore be obtained efficiently. Simulation proceeds window by window until a given stop time is reached.

## 4.2 Inconsistent Initial Conditions of Periodically Switched Nonlinear Circuits

Similar to switched linear circuits, inconsistent initial conditions may occur at switching instants in switched nonlinear circuits. In Section 2.2, we have shown that the two-step algorithm yields the consistent initial conditions for periodically switched linear circuits. In this section, we propose a new extended two-step algorithm to handle the inconsistent initial conditions of switched nonlinear circuits. Because the Volterra circuits of a switched nonlinear circuit are a set of switched linear circuits, the two-step algorithm can be applied to each Volterra circuit to yield the consistent initial conditions of these circuits. At the switching instants, the initial condition of the switched nonlinear circuit is obtained by summing up that of all Volterra circuits. We refer it as the new extended two-step algorithm for periodically switched nonlinear circuits.

Suppose a switching occurs at $t = nT$ and the initial conditions of Volterra circuits before the switching instant are given by $\mathbf{v}_m(nT^-), m = 1, 2, \cdots$. The forward step from $t = nT^-$ to $t = nT^- + T$ yields $\mathbf{v}_m(nT^- + T), m = 1, 2, \cdots$,

$$
\begin{aligned}
\mathbf{v}_1(nT^- + T) &= \mathbf{M}(T)\mathbf{v}_1(nT^-) + \mathbf{P}(T)e^{j\omega_o nT}, \\
\mathbf{v}_m(nT^- + T) &= \mathbf{M}(T)\mathbf{v}_m(nT^-) + \mathcal{R}e\left\{ \frac{a_{m,0}}{2}\mathbf{P}_0(T) + \frac{a_{m,N}}{2}\mathbf{P}_N(T)e^{jN\omega_s nT} \right. \\
&\qquad \left. + \sum_{k=1}^{N-1} a_{m,k}\mathbf{P}_k(T)e^{jk\omega_s nT} \right\} + \mathcal{I}m\left\{ \sum_{k=1}^{N-1} b_{m,k}\mathbf{P}_k(T)e^{jk\omega_s nT} \right\}, \\
&\qquad m = 2, 3, \cdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (4.8)
\end{aligned}
$$

With $\mathbf{v}_m(nT^+ + T) = \mathbf{v}_m(nT^- + T)$ in mind, the backward step from $t = nT^+ + T$ to $t = nT^+$ yields $\mathbf{v}_m(nT^+), m = 1, 2, \cdots$

$$\mathbf{v}_1(nT^+) = \mathbf{M}_B(T)\mathbf{v}_1(nT^+ + T) + \mathbf{P}_B(T)e^{-j\omega_o(nT+T)}, \tag{4.9}$$

$$\mathbf{v}_m(nT^+) = \mathbf{M}_B(T)\mathbf{v}_m(nT^+ + T) + \mathcal{R}e\left\{\frac{a_{m,0}}{2}\mathbf{P}_{B0}(T) + \frac{a_{m,N}}{2}\mathbf{P}_{BN}(T)e^{-jN\omega_s(nT+T)}\right.$$

$$\left. + \sum_{k=1}^{N-1} a_{m,k}\mathbf{P}_{Bk}(T)e^{-jk\omega_s(nT+T)}\right\} + \mathcal{I}m\left\{\sum_{k=1}^{N-1} b_{m,k}\mathbf{P}_{Bk}(T)e^{-jk\omega_s(nT+T)}\right\},$$

$$m = 2, 3, \cdots \tag{4.10}$$

where

$$\mathbf{P}_{Bk}(T) = \mathcal{L}^{-1}\left[\mathbf{A}^{-1}(s)\frac{\mathbf{d}_2}{s - jk\omega_s}\right]_{t=-T}$$

$$= \mathbf{P}_k(-T),$$

$$k = 0, 1, \cdots, N.$$

The initial conditions used in the backward step are $\mathbf{v}_m(nT^+ + T) = \mathbf{v}_m(nT^- + T), m = 1, 2, \cdots$, provided that the time step $T$ is properly chosen such that no discontinuity at $t = nT + T$. Finally we obtain the initial conditions immediately after switching,

$$\mathbf{v}(nT^+) = \sum_{m=1}^{\infty} \mathbf{v}_m(nT^+). \tag{4.11}$$

The width of the clock phase should be taken into account when choosing the width of the simulation window, $T_w$, for the periodically switched nonlinear circuits. Each clock phase should be divided evenly into several sub-phases and the simulation window is set for each sub-phase. This arrangement ensures that a switchings occurs only at the

35

edges of, but not inside the simulation windows. The extended two-step algorithm is implemented only at the beginning of the first simulation window inside a clock phase, as the switching instants are known *a priori*.

## 4.3 Numerical Examples

The proposed algorithms for computing the response of periodically switched nonlinear circuits have been implemented in a computer program RSPSN (Response and Sensitivity of Periodically Switched Nonlinear circuits) using MATLAB. In this section, the response of example periodically switched nonlinear circuits is analyzed using RSPSN. The time domain response from RSPSN is compared with that from PSPICE. Simulation was performed on Sun Ultra-10 workstations with 440MHz CPU and 512MB RAM.

### 4.3.1 Switched Circuit with Nonlinear Conductors
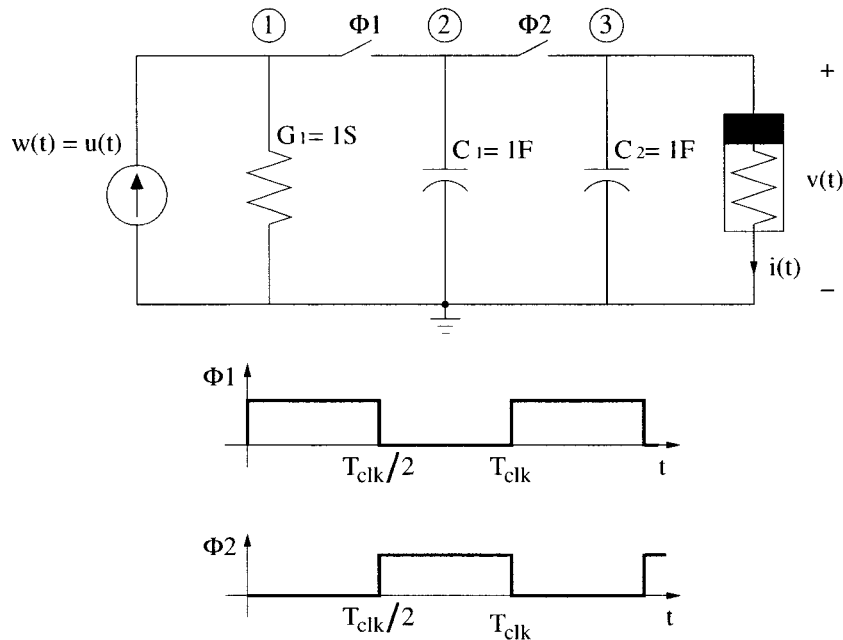


**Figure 4.2:** Switched circuit with nonlinear conductor
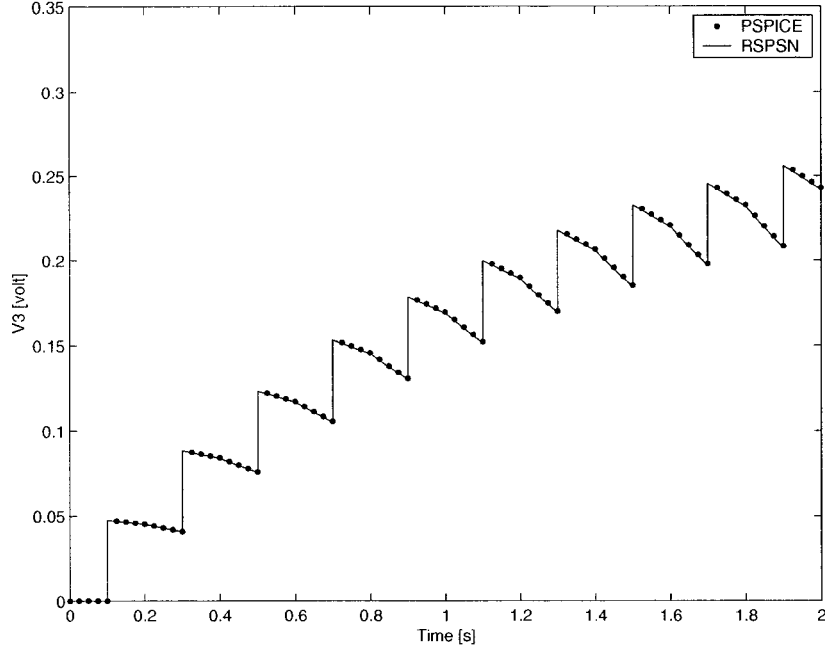
**Figure 4.3:** Response of switched nonlinear circuit

The first example is a switched nonlinear circuit shown in Fig.4.2 that contains two externally clocked ideal switches. The clock phases are non-overlapping. The voltages of the two capacitors are well defined inside either clock phases but may discontinue at switching instants. The clock frequency of the circuit is 5 Hz. Each clock period has two phases of equal width. $G_1 = 1S$, $C_1 = 1F$, and $C_2 = 1F$. Zero initial conditions are assumed for all capacitors. The nonlinear conductor is modeled as

$$i(t) = g_1 v(t) + g_2 v^2(t) + g_3 v^3(t),$$

where $g_1, g_2, g_3$ are constants. In our simulation, they are chosen as

$$\{g_1, g_2, g_3\} = \{1.0, 0.5, 0.25\}.$$

The input is a unit step current source, $w(t) = u(t)$. Up to the third-order Volterra

37
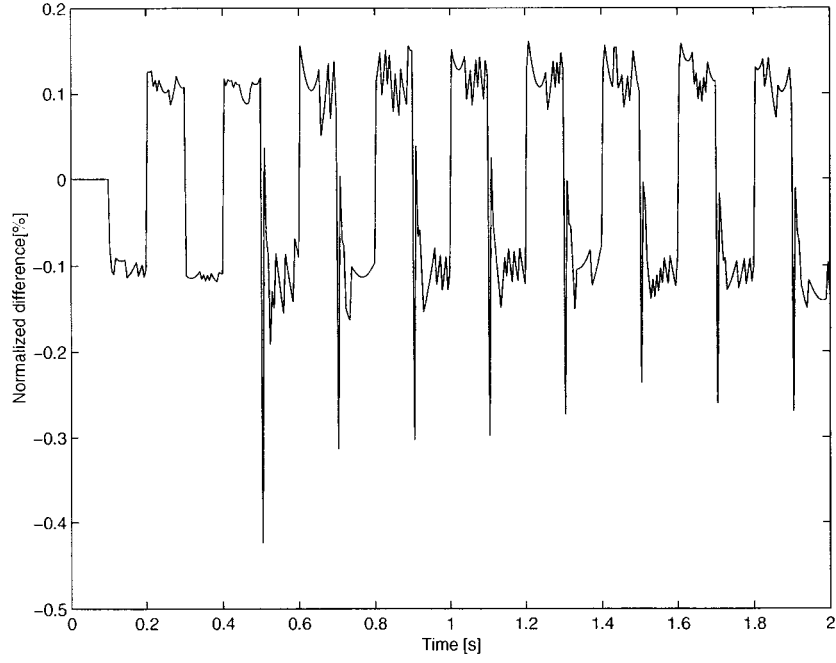
**Figure 4.4:** Normalized difference of response between RSPSN and PSPICE

series expansion is used. To simplify simulation, the width of the simulation window is set to be the same as that of the clock phase, i.e., $0.1s$. The number of samples per window is 20.

The voltages at all nodes are solved using RSPSN and the results for node 3 are plotted in Fig.4.3. As shown in the figure, the voltage at node 3 is discontinuous at $t = 0.1s, 0.3s, 0.5s, ...$, where inconsistent initial conditions are encountered. It is continuous at $t = 0.2s, 0.4s, 0.6s, ...$, where only consistent initial conditions are encountered. This observation demonstrates that the proposed method can correctly handle both the inconsistent and consistent initial conditions at the switching instants.

The results of PSPICE analysis is also plotted in the figure for comparison. They are virtually indistinguishable from those from RSPSN. The normalized difference between the results from RSPSN and those from PSPICE are measured. Two different situations are considered. Inside each clock phase where no discontinuity occurs in the response,

38
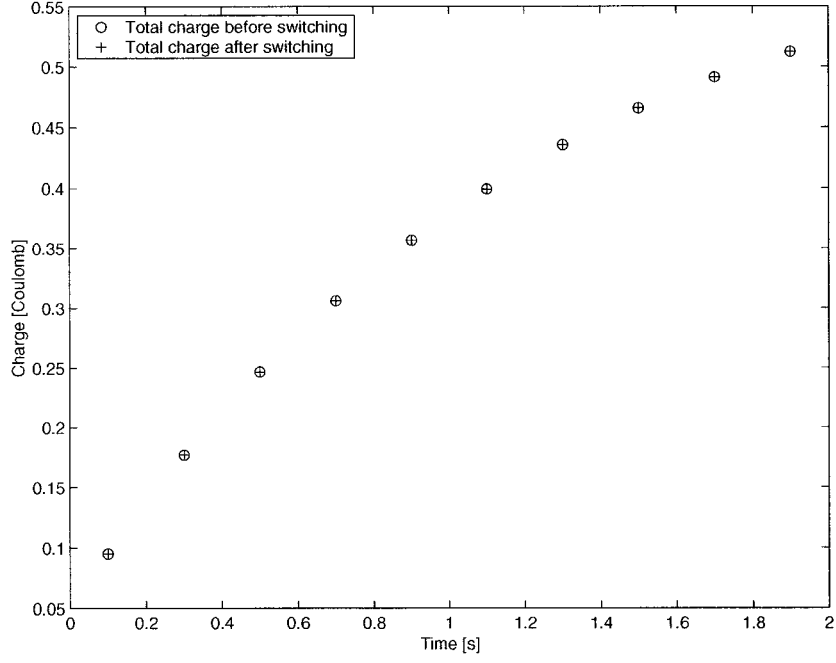
**Figure 4.5:** Charge conservation before and after switching instants

the normalized difference for time points except switching instants is defined as

$$\text{Normalized difference} = \frac{v(nT) - \hat{v}(nT)}{v(nT)} \times 100\%,$$

where $v(nT)$ and $\hat{v}(nT)$ are the responses from RSPSN and PSPICE, respectively. It can be obtained directly from the responses, and is shown in Fig. 4.4. It is seen that the maximum normalized difference is below 0.5%.

At the switching instants, a direct comparison of the results from RSPSN and PSPICE is difficult. The reason is as follows: If a switching action causes a discontinuity at $t = t_0$, although the inconsistent initial conditions immediately before and after switching $\mathbf{v}(t_0^-)$ and $\mathbf{v}(t_0^+)$ can be obtained from RSPSN, they can not be obtained from PSPICE. This is because, for traditional integration methods such as those used in PSPICE, the continuity of network variables is required. In order to handle a
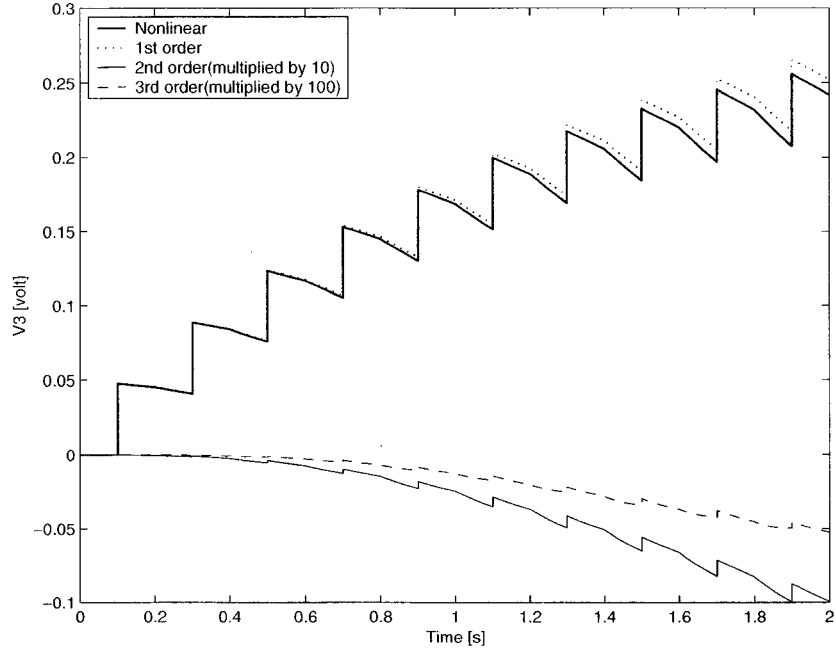
**Figure 4.6:** Responses of switched nonlinear circuit and its Volterra circuits

rapid change in response, very small time steps must be taken to ensure the accuracy of integration. No time point can be clearly defined as a "switching instant".

At the switching instants, the initial conditions obtained from RSPSN are verified by checking the charge conservation before and after switching. The total charge on capacitors $C_1$ and $C_2$ immediately before and after switching is computed and plotted in Fig. 4.5. It is seen that the total charge before switching is equal to the total charge after switching, revealing that RSPSN handles the inconsistent initial conditions at the switching instants correctly.

The responses of the first, second, and third-order Volterra circuits are shown in Fig. 4.6, together with the complete response of the nonlinear circuit. It is evident that the sum of the three Volterra circuits converges to the response of the periodically switched nonlinear circuit.

## 4.3.2 Switched-Capacitor Integrator with Nonlinear Opamp
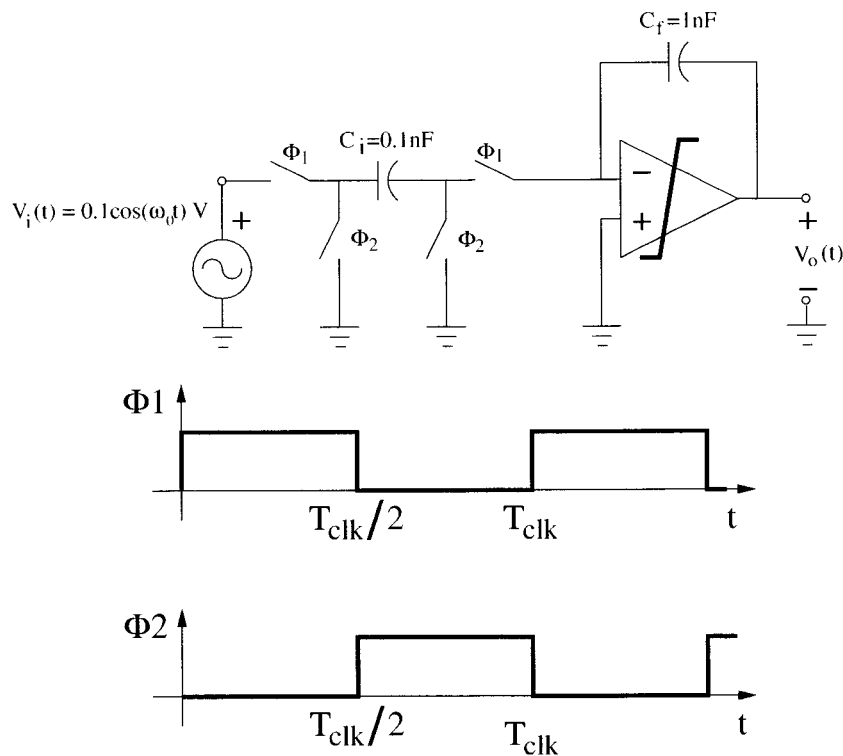


**Figure 4.7:** Nonlinear switched-capacitor integrator

The second example is a switched-capacitor integrator with a nonlinear opamp, as shown in Fig 6.6. There are four externally clocked ideal switches. The clock frequency is 100 kHz. Each clock period has two non-overlapping phases with equal width. The input to the integrator is a voltage source of 1 kHz cosine wave with $0.1V$ amplitude, $V_i = 0.1\cos(2000\pi t)V$. $C_i$ and $C_f$ are $0.1nF$ and $1nF$, respectively. Zero initial conditions are assumed for all capacitors. The gain of the opamp is modeled as

$$Gain = a_0 + a_1(v_+ - v_-) + a_2(v_+ - v_-)^2,$$

where $a_0, a_1, a_2$ are constants. In our implementation, they were chosen as

41
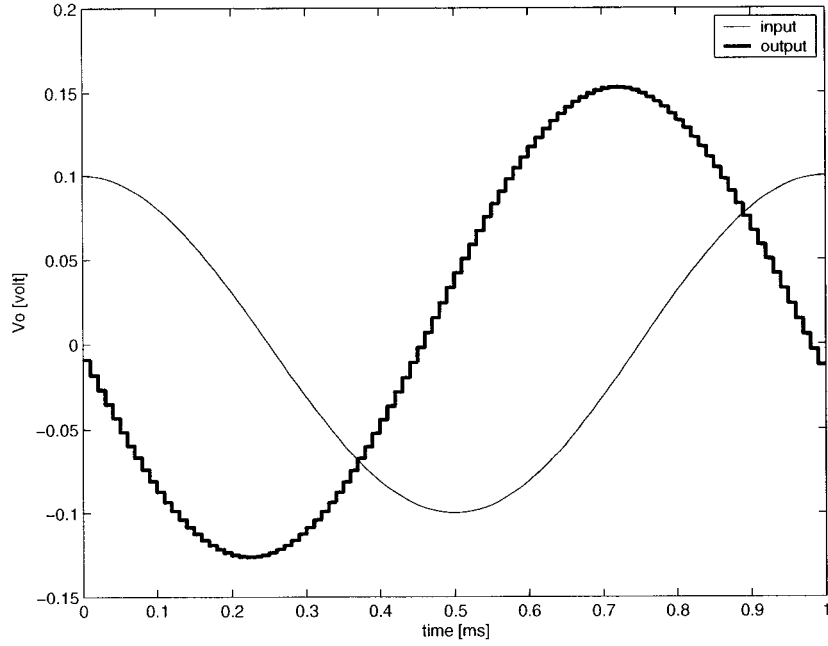
**Figure 4.8:** Response of nonlinear switched-capacitor integrator

$$\{a_0, a_1, a_2\} = \{10, -5, -2.5\}.$$

The time domain output of the switched-capacitor integrator, $V_o$, is calculated using RSPSN and shown in Fig. 4.8. The output responses of the first, second, and third-order Volterra circuits of the integrator are also shown in Fig. 4.9.

## 4.4 Summary

In this chapter, we have proposed a new efficient algorithm for time domain analysis of periodically switched nonlinear circuits. In the algorithm, a nonlinear circuit is represented by its Volterra circuits and its time domain response is calculated window by window. The algorithm is efficient for the analysis of periodically switched nonlinear circuits because the sampled data simulation method is used instead of computationally intensive Newton-Raphson algorithm in each integration step. At switching instants, the
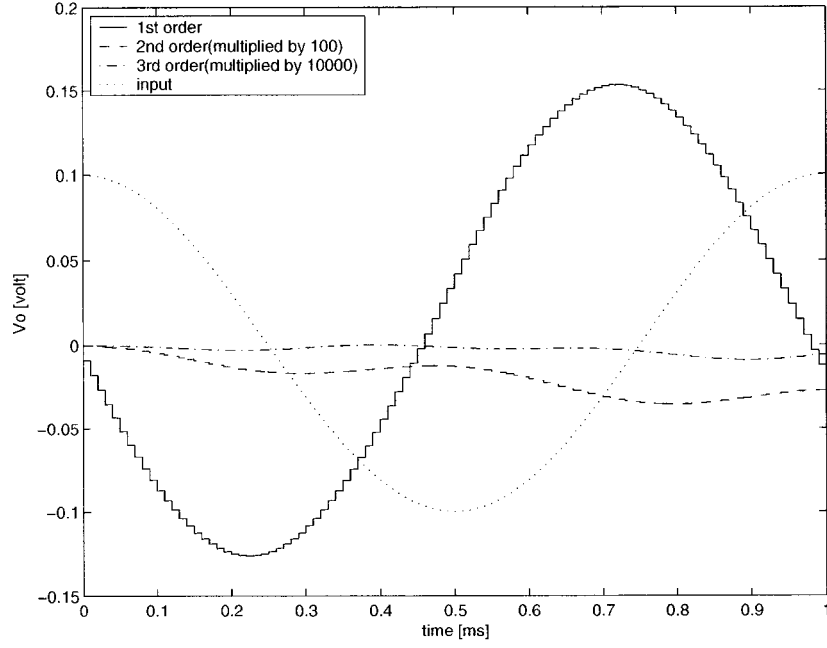
**Figure 4.9:** Responses of Volterra circuits of nonlinear switched-capacitor integrator

proposed algorithm handles the inconsistent initial conditions correctly and efficiently using the extended two-step algorithm also developed in this chapter. The accuracy and efficiency of the new algorithm have been assessed using example circuits. Inconsistent initial conditions at the switching instants are correctly handled by RSPSN and verified by the charge conservation of all capacitors before and after switching.

# Chapter 5

# Time Domain Sensitivity of Periodically Switched Linear Circuits

Sensitivity analysis is required for the evaluation and optimization of circuit design. This chapter presents a new method for time domain sensitivity analysis of periodically switched linear circuits. After a brief introduction of different sensitivity analysis approaches in Section 5.1, the sampled data simulation based time domain sensitivity analysis method for linear circuit given in [15] is reviewed in Section 5.2. In Section 5.3, we propose a new time domain sensitivity analysis method for periodically switched linear circuits. It can handle the discontinuities of sensitivity networks that might occur at switching instants. Numerical examples are given in Section 5.4. The chapter is summarized in Section 5.5. The primitive results of this chapter were presented in [32] and [33].

## 5.1   Introduction

Sensitivity is a mathematical measure of circuit performance. Sensitivity analysis provides an insight into the behavior of circuits. There are many reasons for sensitivity

analysis in the circuits design. For example, it is used to evaluate the effect of parameter variation on the response of circuits. It also plays an important role in the tolerance analysis and design optimization of circuits.

In [16], various definitions of sensitivity for linear circuits were given both in frequency and time domains. Different approaches were available for sensitivity analysis of linear circuits. In the sensitivity network approach, the sensitivities of all variables with respect to one circuit parameter are computed. In the adjoint network approach, the sensitivities of one variable with respect to all parameters are calculated.

Some computer methods for frequency domain sensitivity analysis of periodically switched linear circuits were developed in the last decade. In [12], an accurate frequency domain method that is based on the sensitivity network approach to compute the sensitivity of periodically switched linear circuits was proposed. Using the adjoint network approach and Tellegen's theorem, Yuan and Opal presented another efficient and accurate computational method for sensitivity analysis of periodically switched linear circuits in frequency domain [14].

In the time domain, based on the sampled date simulation and sensitivity network approach, an efficient computational method was proposed in [15] for the sensitivity analysis of linear circuits. Time domain sensitivity analysis method for general periodically switched linear circuits are needed and it is the subject of this chapter. Further more, the time domain sensitivity method for periodically switched linear circuits developed in this chapter will be extended for periodically switched nonlinear circuits in the following chapter.

## 5.2 Time Domain Sensitivity of Linear Circuits

The time domain sensitivity of the response $\mathbf{v}(t)$ with respect to a circuit parameter $x$ is defined as

$$\mathbf{z}(t) = \frac{\partial \mathbf{v}(t)}{\partial x}.$$

The time domain sensitivity $\mathbf{z}(t)$ is calculated by solving the sensitivity network, which can be obtained by assuming the independence of the input $w(t)$ on the circuit parameter $x$ and differentiating Eq. (2.4) with respect to $x$

$$\mathbf{G}\mathbf{z}(t) + \mathbf{C}\frac{d\mathbf{z}(t)}{dt} = -\left[\frac{\partial \mathbf{G}}{\partial x}\mathbf{v}(t) + \frac{\partial \mathbf{C}}{\partial x}\frac{d\mathbf{v}(t)}{dt}\right], \quad \mathbf{z}(t)|_{t=0^-} = \mathbf{z}(0^-). \tag{5.1}$$

Equation (5.1) shows that the sensitivity network of a linear time invariant circuit is still a linear time invariant system. The sensitivity network has the same topology and components values as that of its original circuit. But its inputs are different. They are functions of the response of its original circuit. The time domain sensitivities of all variables with respect to a parameter $x$ are obtained after solving the sensitivity network depicted in Eq. (5.1).

It can be proven easily that the solution of the sensitivity network Eq. (5.1) can also be obtained by directly differentiating the time domain response of its original circuit, as shown in Eq. (2.4), with respect to $x$

$$\begin{aligned}
\mathbf{z}(nT + T) &= \frac{\partial \mathbf{M}(T)}{\partial x}\mathbf{v}(nT) + \mathbf{M}(T)\frac{\partial \mathbf{v}(nT)}{\partial x} + \frac{\partial \mathbf{P}(T)}{\partial x}e^{j\omega_o nT} \\
&= \mathbf{M}_s(T)\mathbf{v}(nT) + \mathbf{M}(T)\mathbf{z}(nT) + \mathbf{P}_s(T)e^{j\omega_o nT},
\end{aligned} \tag{5.2}$$

where

$$
\begin{aligned}
\mathbf{M}_s(T) &= \frac{\partial}{\partial x}\mathbf{M}(T) \\
&= \frac{\partial}{\partial x}\mathcal{L}^{-1}[\mathbf{A}^{-1}(s)\mathbf{C}]_{t=T} \\
&= \mathcal{L}^{-1}\left[\frac{\partial \mathbf{A}^{-1}(s)}{\partial x}\mathbf{C} + \mathbf{A}^{-1}(s)\frac{\partial \mathbf{C}}{\partial x}\right]_{t=T}, \\
\mathbf{P}_s(T) &= \frac{\partial}{\partial x}\mathbf{P}(T) \\
&= \frac{\partial}{\partial x}\mathcal{L}^{-1}\left[\mathbf{A}^{-1}(s)\frac{\mathbf{d}}{s - j\omega_o}\right]_{t=T} \\
&= \mathcal{L}^{-1}\left[\frac{\partial \mathbf{A}^{-1}(s)}{\partial x}\frac{\mathbf{d}}{s - j\omega_o}\right]_{t=T}.
\end{aligned}
$$

We moved the partial differentiation operator $\dfrac{\partial}{\partial x}$ inside the brackets because $x$ is independent of both frequency $s$ and the time step $T$. Note that $\dfrac{\partial \mathbf{A}^{-1}(s)}{\partial x}$ can be computed efficiently using

$$
\frac{\partial \mathbf{A}^{-1}(s)}{\partial x} = -\mathbf{A}^{-1}(s)\frac{\partial \mathbf{A}(s)}{\partial x}\mathbf{A}^{-1}(s),
$$

where

$$
\frac{\partial \mathbf{A}(s)}{\partial x} = \frac{\partial \mathbf{G}}{\partial x} + s\frac{\partial \mathbf{C}}{\partial x}.
$$

Using the sampled data simulation given in Chapter 2, we can calculate the time domain sensitivity of linear circuits from Eq. (5.2).

47

## 5.3    Time Domain Sensitivity of Periodically Switched Linear Circuits

In this section, a new method for time domain sensitivity analysis of periodically switched linear circuits is presented. The two-step algorithm is applied to solve the discontinuity problems that might occur at the switching instants in the sensitivity network of periodically switched linear circuits.

The sensitivity networks of periodically switched linear circuits are also periodically switched linear systems. Both sampled data simulation and two-step algorithm can be applied to the sensitivity networks. The time domain sensitivity of periodically switched linear circuits is obtained by solving the sensitivity networks.

Similar to the inconsistent initial conditions encountered in solving the periodically switched linear circuits, the discontinuity problems may also be encountered at switching instants in solving the sensitivity networks of periodically switched linear circuits. The two-step algorithm used to handle the inconsistent initial conditions of response also can be applied the sensitivity networks of periodically switched linear circuits to handle the inconsistent initial conditions. Assume a switching occurs at $t = nT$ and the initial condition before switching is given by $\mathbf{z}(nT^-)$. $\mathbf{z}(nT^+ + T)$ is computed using Eq. (5.2) in the forward step. In the backward step, the sensitivity immediately after switching, denoted by $\mathbf{z}(nT^+)$, can be solved by differentiating Eq. (2.8) with respect to $x$

$$
\begin{aligned}
\mathbf{z}(nT^+) &= \frac{\partial \mathbf{M}_B(T)}{\partial x}\mathbf{v}(nT^+ + T) + \mathbf{M}_B(T)\frac{\partial \mathbf{v}(nT^+ + T)}{\partial x} + \frac{\partial \mathbf{P}_B(T)}{\partial x}e^{-j\omega_o(nT+T)} \\
&= \mathbf{M}_{Bs}(T)\mathbf{v}(nT^+ + T) + \mathbf{M}_B(T)\mathbf{z}(nT^+ + T) + \mathbf{P}_{Bs}(T)e^{-j\omega_o(nT+T)}, \quad (5.3)
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{M}_{Bs}(T) &= \frac{\partial}{\partial x}\mathbf{M}_B(T) \\
&= \frac{\partial}{\partial x}\mathcal{L}^{-1}[\mathbf{A}^{-1}(s)\mathbf{C}]_{t=-T} \\
&= \mathcal{L}^{-1}\left[\frac{\partial \mathbf{A}^{-1}(s)}{\partial x}\mathbf{C} + \mathbf{A}^{-1}(s)\frac{\partial \mathbf{C}}{\partial x}\right]_{t=-T} \\
&= \mathbf{M}_s(-T), \\
\mathbf{P}_{Bs}(T) &= \frac{\partial}{\partial x}\mathbf{P}_B(T) \\
&= \frac{\partial}{\partial x}\mathcal{L}^{-1}\left[\mathbf{A}^{-1}(s)\frac{\mathbf{d}}{s-j\omega_o}\right]_{t=-T} \\
&= \mathcal{L}^{-1}\left[\frac{\partial \mathbf{A}^{-1}(s)}{\partial x}\frac{\mathbf{d}}{s-j\omega_o}\right]_{t=-T} \\
&= \mathbf{P}_s(-T).
\end{aligned}
\tag{5.4}
$$

Similar to $\mathbf{M}(T)$, $\mathbf{P}(T)$, $\mathbf{M}_B(T)$, and $\mathbf{P}_B(T)$ for response analysis, $\mathbf{M}_s(T)$, $\mathbf{P}_s(T)$, $\mathbf{M}_{Bs}(T)$, and $\mathbf{P}_{Bs}(T)$ for sensitivity analysis are constant if the time step $T$ is not changed. They need to be computed only once, and can be computed along with $\mathbf{M}(T)$, $\mathbf{P}(T)$, $\mathbf{M}_B(T)$, $\mathbf{P}_B(T)$ in a pre-processing step prior to the start of simulation. Once these matrices and vectors are known, the time domain sensitivity at time points of an equal interval can be obtained efficiently from two matrix-vector multiplications and one vector addition.

The time domain sensitivity of periodically switched linear circuits is obtained by solving the sensitivity networks inside the clock phases. The discontinuities of sensitivity that might occur at the switching instants are computed using the two-step algorithms for the sensitivity networks described in this section.

49

# 5.4  Numerical Examples

In this section, we use the inverting switched-capacitor integrator as an example to assess the time domain sensitivity analysis algorithms for periodically switched linear circuits presented in the previous sections. It is worthful to note that the algorithms of sensitivity analysis presented in this chapter and the algorithms of response analysis used in Chapter 2 for periodically switched linear are all implemented in the same computer program RSPSN for the time domain analysis of periodically switched nonlinear circuits. Since the time domain sensitivity simulation and response simulation are merged in the same program, the sensitivity simulation can be done without too much extra computational effort along with response simulation. The results from RSPSN is validated by comparing with those from the brute-force (BF) method.

## 5.4.1  Sensitivity of Inverting Switched-Capacitor Integrator

The example circuit for time domain sensitivity analysis is the same inverting switched-capacitor integrator used for time domain response analysis in Chapter 2, as shown in Fig. 5.1. Ideal operational amplifier is assumed in the integrator. The time domain sensitivities of the output voltage $V_o$ with respect to capacitors $C_i$ and $C_f$ are calculated with the same settings as those used in response simulation. The results are plotted in Fig. 5.2 and Fig. 5.3. In these figures, the definition of semi-normalized sensitivity is used [16], which defines the sensitivity of function $F$ with respect to a parameter $x$ as

$$S_x^F = x \frac{\partial F}{\partial x}.$$

The sensitivities of the output voltage $V_o$ with respect to capacitors $C_i$ and $C_f$ are also computed using the brute-force method, which estimates the sensitivity of the response

50
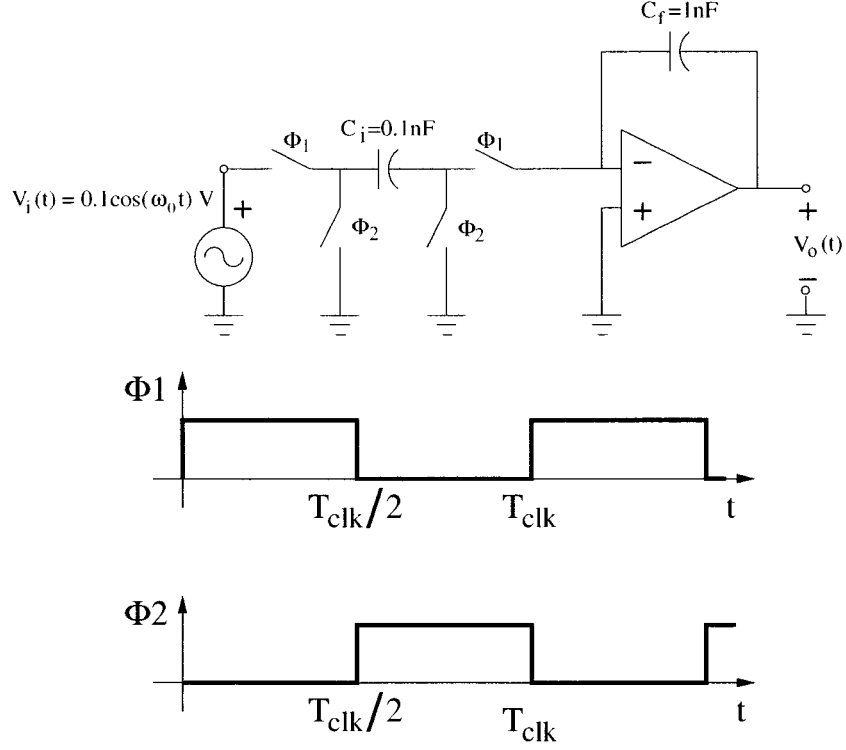
**Figure 5.1:** Switched-capacitor integrator with ideal opamp

$v(t)$ with respect to a circuit parameter $x$ using

$$\frac{\partial v(t)}{\partial x} \approx \frac{\Delta v(t)}{\Delta x}.$$

In our implementation of the brute-force method, $\frac{\Delta x}{x} = 1\%$ is used to calculate $\Delta v(t)$. The results from BF are plotted in Fig. 5.4 and Fig. 5.5 together with those from RSPSN for comparison. It is seen that the time domain sensitivities obtained from RSPSN and BF match well.

## 5.5   Summary

In this chapter, we have presented the new method for time domain sensitivity analysis of periodically switched linear circuits. The method is based on the combination of sam-

51
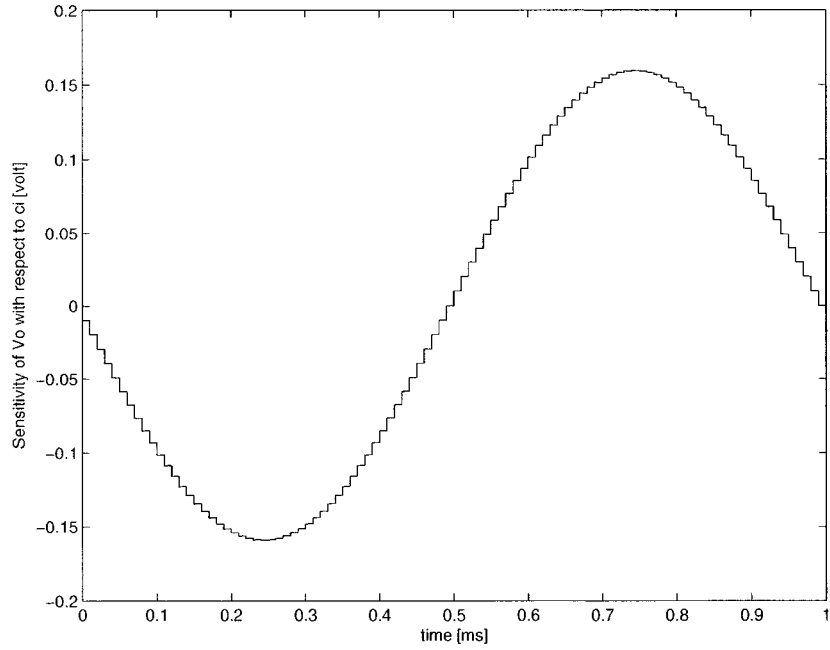
**Figure 5.2:** Sensitivity of $V_o$ with respect to $C_i$

pled date simulation and two-step algorithm of Chapter 2. Inside the clock phases, the time domain sensitivity is calculated by solving the sensitivity networks. The method has the ability to handle the discontinuities of sensitivity that might occur at the switching instants. The time domain sensitivities of all variables with respect to one circuit parameter are computed efficiently for periodically switched linear circuits. The method was assessed using numerical examples.
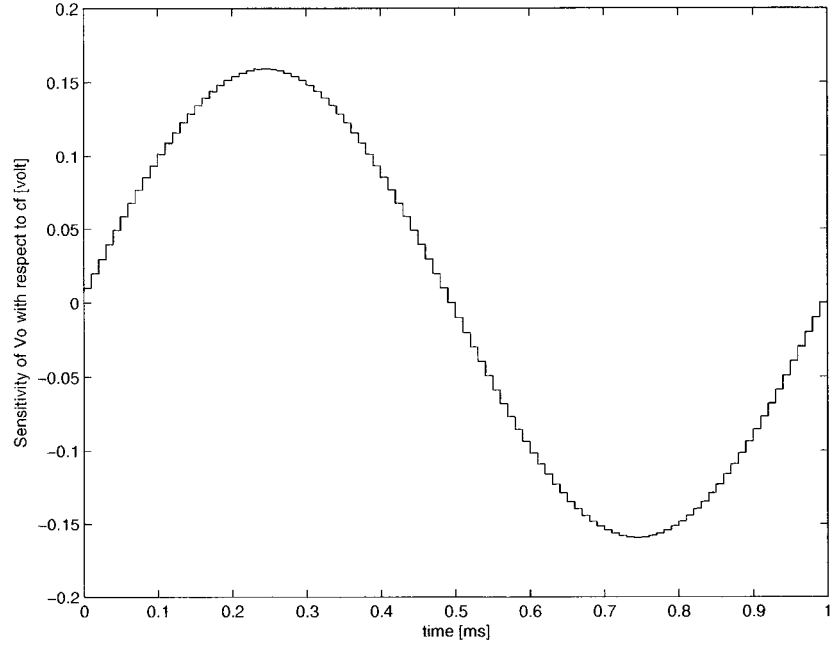
**Figure 5.3:** Sensitivity of $V_o$ with respect to $C_f$



**Figure 5.4:** Sensitivity of $V_o$ with respect to $C_i$ from RSPSN and BF

53
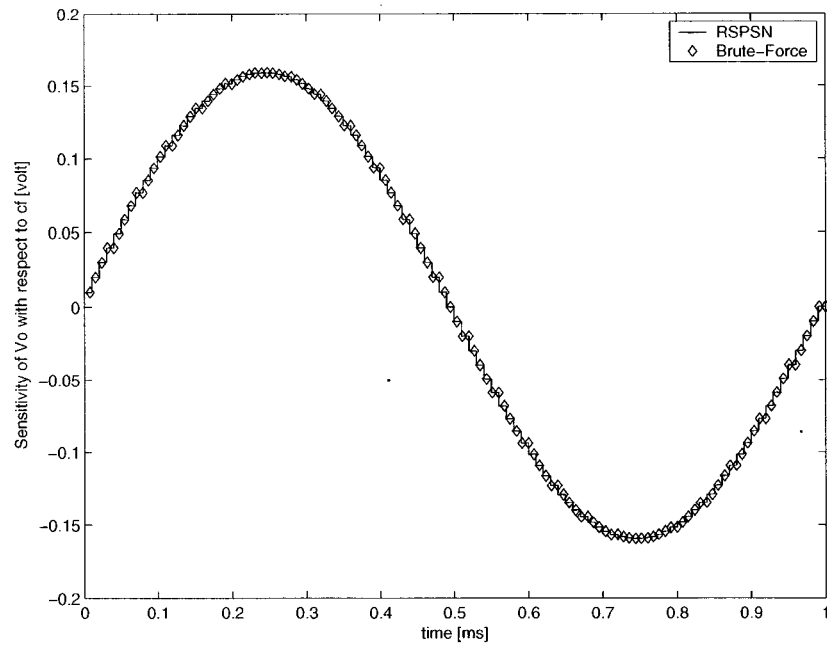
**Figure 5.5:** Sensitivity of $V_o$ with respect to $C_f$ from RSPSN and BF

# Chapter 6

# Time Domain Sensitivity of Periodically Switched Nonlinear Circuits

The sensitivity analysis of nonlinear circuits is mostly performed in time domain, because direct frequency domain sensitivity analysis for nonlinear circuits is not straightforward. In this chapter, we propose a new method for time domain sensitivity analysis of periodically switched nonlinear circuits. The algorithm is implemented in a computer program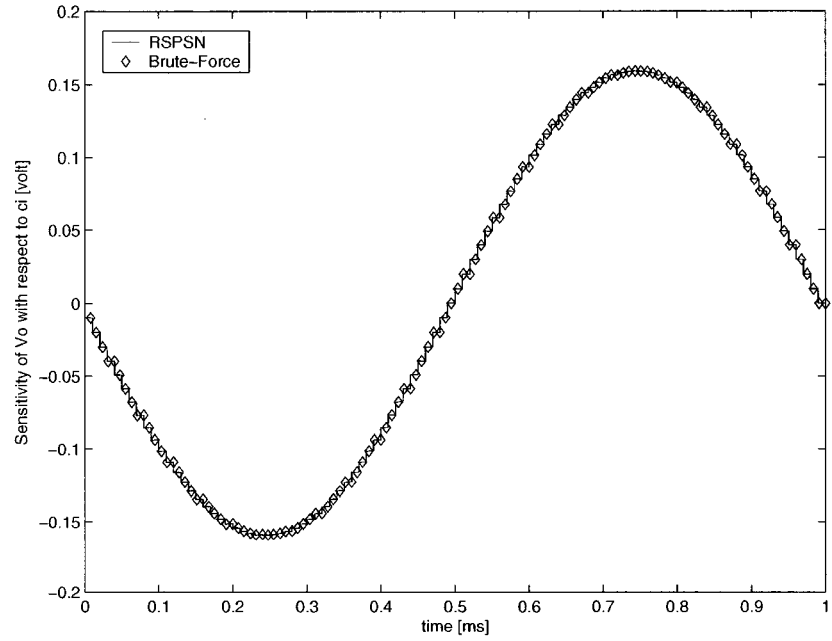 in Matlab and assessed using numerical examples. The primitive results of this chapter were presented in [32] and [33].

First, a brief introduction of sensitivity analysis methods for nonlinear circuits is given in Section 6.1. A Volterra circuits based time domain sensitivity analysis method for nonlinear circuits [13] is reviewed in Section 6.2. In Section 6.3, we propose a new time domain sensitivity method for periodically switched nonlinear circuits. It can handle the discontinuity problems encountered in sensitivity analysis of these circuits. Numerical examples are given in Section 6.4. Finally, the chapter is summarized in Section 6.5.

# 6.1   Introduction

Sensitivity analysis of electronic circuits can be performed in either time or frequency domains. For linear time invariant circuits and even periodically switched linear circuits, both time domain and and frequency domain sensitivity analysis methods are available, as described in Chapter 5. But it seems that the time domain sensitivity analysis becomes the only possibility for nonlinear circuits and periodically switched nonlinear circuits. This is because the frequency domain sensitivity analysis is not straightforward and complicated in the nonlinear cases.

In addition to the response analysis of nonlinear circuits, efficient and accurate simulation methods for sensitivity analysis of nonlinear circuits is needed. Based on Volterra functional series and sampled data simulation, Yuan and Opal proposed an efficient time domain sensitivity analysis method for mildly nonlinear circuits in [13]. But, to the best knowledge of the author, time domain sensitivity analysis of periodically switched nonlinear circuits is not available. This becomes the subject of this chapter.

# 6.2   Time Domain Sensitivity of Nonlinear Circuits

Time domain sensitivity of the response with respect to a circuit parameter $x$ is defined as,

$$\mathbf{z}(t) = \frac{\partial \mathbf{v}(t)}{\partial x}.$$

For a nonlinear circuit, its time domain sensitivity, at discrete time $t = nT, n = 1, 2, \cdots$, can be represented as

$$\mathbf{z}(nT) = \frac{\partial \mathbf{v}(nT)}{\partial x}.$$

The response of a nonlinear circuit at $t = nT$ is obtained from its Volterra circuits, as described in Chapter 4,

$$\mathbf{v}(nT) = \sum_{m=1}^{\infty} \mathbf{v}_m(nT).$$

Define the time domain sensitivity of the $m$-th order Volterra circuit at discrete time $t = nT$ as,

$$\mathbf{z}_m(nT) = \frac{\partial \mathbf{v}_m(nT)}{\partial x}.$$

For a nonlinear circuit, the time domain sensitivity of the response with respect to a circuit parameter $x$, at discrete time $t = nT$, is obtained

$$
\begin{aligned}
\mathbf{z}(nT) &= \frac{\partial \mathbf{v}(nT)}{\partial x} \\
&= \sum_{m=1}^{\infty} \frac{\partial \mathbf{v}_m(nT)}{\partial x} \\
&= \sum_{m=1}^{\infty} \mathbf{z}_m(nT)
\end{aligned}
\tag{6.1}
$$

Eq. (6.1) shows that the time domain sensitivity of a nonlinear circuit is obtained by summing up all the sensitivities of its Volterra circuits.

The sensitivity of the first-order Volterra circuit is obtained directly from Eq. (5.2),

$$\mathbf{z}_1(nT + T) = \mathbf{M}_s(T)\mathbf{v}_1(nT) + \mathbf{M}(T)\mathbf{z}_1(nT) + \mathbf{P}_s(T)e^{j\omega_o nT}. \tag{6.2}$$

To calculate the sensitivity of the $m$-th order Volterra circuit, we differentiate Eq (4.7) with respect to $x$ and note that $\frac{\partial}{\partial x}(e^{jk\omega_s nT}) = 0$,

$$
\begin{aligned}
\mathbf{z}_m(nT+T) =\ & \mathbf{M}_s(T)\mathbf{v}_m(nT) + \mathbf{M}(T)\mathbf{z}_m(nT) \\
& +\frac{1}{2}\mathcal{R}e\left\{\left[a_{m,0}\mathbf{P}_{0s}(T) + \frac{\partial a_{m,0}}{\partial x}\mathbf{P}_0(T)\right]\right. \\
& \left. +\left[a_{m,N}\mathbf{P}_{Ns}(T) + \frac{\partial a_{m,N}}{\partial x}\mathbf{P}_N(T)\right]e^{jN\omega_s nT}\right\} \\
& +\mathcal{R}e\left\{\sum_{k=1}^{N-1}\left[a_{m,k}\mathbf{P}_{ks}(T) + \frac{\partial a_{m,k}}{\partial x}\mathbf{P}_k(T)\right]e^{jk\omega_s nT}\right\} \\
& +\mathcal{I}m\left\{\sum_{k=1}^{N-1}\left[b_{m,k}\mathbf{P}_{ks}(T) + \frac{\partial b_{m,k}}{\partial x}\mathbf{P}_k(T)\right]e^{jk\omega_s nT}\right\}, \\
& m = 2,3,\cdots \quad\quad (6.3)
\end{aligned}
$$

where $\mathbf{M}_s(T)$, $\mathbf{P}_s(T)$, and $\mathbf{P}_k(T)$ were defined previously and

$$
\begin{aligned}
\mathbf{P}_{ks}(T) &= \frac{\partial \mathbf{P}_k(T)}{\partial x} \\
&= \frac{\partial}{\partial x}\mathcal{L}^{-1}\left[\mathbf{A}^{-1}(s)\frac{\mathbf{d}_2}{s-jk\omega_s}\right]_{t=T} \\
&= \mathcal{L}^{-1}\left[\frac{\partial \mathbf{A}^{-1}(s)}{\partial x}\frac{\mathbf{d}_2}{s-jk\omega_s}\right]_{t=T}, \\
& k = 0,1,\cdots,N.
\end{aligned}
$$

To calculate the partial derivatives of $\frac{\partial}{\partial x}[\mathbf{a}_m]$ and $\frac{\partial}{\partial x}[\mathbf{b}_m]$ in Eq. (6.3), we differentiate Eq. (4.5) and Eq. (4.6) with respect to $x$, respectively,

$$
\frac{\partial}{\partial x}[\mathbf{a}_m] = \frac{\partial}{\partial x}
\begin{bmatrix}
a_{m,0} \\
a_{m,1} \\
\vdots \\
a_{m,N}
\end{bmatrix}
$$

$$= \mathbf{Q}(N,T)\frac{\partial}{\partial x}\begin{bmatrix} f_m(0) \\ f_m(T) \\ \vdots \\ f_m[(2N-1)T] \end{bmatrix},$$

$$\frac{\partial}{\partial x}[\mathbf{b}_m] = \frac{\partial}{\partial x}\begin{bmatrix} b_{m,1} \\ b_{m,2} \\ \vdots \\ b_{m,N-1} \end{bmatrix}$$

$$= \mathbf{R}(N,T)\frac{\partial}{\partial x}\begin{bmatrix} f_m(T) \\ f_m(2T) \\ \vdots \\ f_m[(2N-1)T] \end{bmatrix}.$$

Since $f_m(nT)$ is a function of $\mathbf{v}_i(nT)$, $i = 1, 2, \cdots, m-1$, the partial derivative of $f_m(nT)$ with respect to $x$, $\frac{\partial}{\partial x}f_m(nT)$, is a function of $\mathbf{v}_i(nT)$ and $\mathbf{z}_i(nT)$, $i = 1, 2, \cdots, m-1$. It can be computed once the responses $\mathbf{v}_i(nT)$ and sensitivities $\mathbf{z}_i(nT)$ of the lower-order Volterra circuits are available. Take the circuit shown in Fig. 3.1 as an example, from Eq. (3.6), the partial derivatives are obtained

$$\begin{aligned}
\frac{\partial}{\partial x}f_2(nT) &= \left\{\frac{\partial}{\partial x}f_2[v_1(t)]\right\}_{t=nT} \\
&= -\left\{2g_2 v_1(t)z_1(t)\right\}_{t=nT} \\
&= -2g_2 v_1(nT)z_1(nT).
\end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial x}f_3(nT) &= \left\{\frac{\partial}{\partial x}f_3[v_1(t), v_2(t)]\right\}_{t=nT} \\
&= -\left\{2g_2[z_1(t)v_2(t) + v_1(t)z_2(t)] + 3g_3 v_1^2(t)z_1(t)\right\}_{t=nT} \\
&= -2g_2[z_1(nT)v_2(nT) + v_1(nT)z_2(nT)] - 3g_3 v_1^2(nT)z_1(nT).
\end{aligned}$$

It is seen that the the partial derivatives $\frac{\partial}{\partial x}f_2(nT)$ and $\frac{\partial}{\partial x}f_3(nT)$ are functions of $[v_1(nT), z_1(nT)]$ and $[v_1(nT), z_1(nT), v_2(nT), z_2(nT)]$, respectively. They are readily calculated from the responses and sensitivities of the low-order Volterra circuits.

## 6.3 Time Domain Sensitivity of Periodically Switched Nonlinear Circuits

By means of the extended two-step algorithm presented in Chapter 2, we propose a new method for time domain sensitivity analysis of periodically switched nonlinear circuits in this section. The discontinuity problem that might occur at switching instants in sensitivity analysis of periodically switched nonlinear circuits is solved correctly. Inside the clock phases, a periodically switched nonlinear circuit is treated as a nonlinear time-invariant circuit. Its time domain sensitivity is calculated using the method given in Section 6.2. At the switching instants, the two-step algorithm is applied to obtain the consistent initial conditions for sensitivity analysis of periodically switched nonlinear circuits, as detailed below.

Assume a switching occurs at $t = nT$. In the forward step, we calculate the sensitivity of the Volterra circuits at $t = nT^+ + T, z_m(nT^+ + T), m = 1, 2, \cdots$, using Eq (6.2) and Eq (6.3). In the backward step, we calculate the sensitivity of the Volterra circuits at $t = nT^+$ by differentiating Eq (4.9) and Eq (4.10) with respect to $x$,

$$\mathbf{z}_1(nT^+) = \mathbf{M}_{Bs}(T)\mathbf{v}_1(nT^+ + T) + \mathbf{M}_B(T)\mathbf{z}_1(nT^+ + T) + \mathbf{P}_{Bs}(T)e^{-j\omega_o(nT+T)},$$

$$\mathbf{z}_m(nT^+) = \mathbf{M}_{Bs}(T)\mathbf{v}_m(nT^+ + T) + \mathbf{M}_B(T)\mathbf{z}_m(nT^+ + T)$$
$$+ \frac{1}{2}\mathcal{R}e\left\{\left[a_{m,0}\mathbf{P}_{B0s}(T) + \frac{\partial a_{m,0}}{\partial x}\mathbf{P}_{B0}(T)\right]\right.$$

$$+\left[a_{m,N}\mathbf{P}_{BNs}(T) + \frac{\partial a_{m,N}}{\partial x}\mathbf{P}_{BN}(T)\right]e^{-jN\omega_s(nT+T)}\Bigg\}$$

$$+\mathcal{R}e\left\{\sum_{k=1}^{N-1}\left[a_{m,k}\mathbf{P}_{Bks}(T) + \frac{\partial a_{m,k}}{\partial x}\mathbf{P}_{Bk}(T)\right]e^{-jk\omega_s(nT+T)}\right\}$$

$$+\mathcal{I}m\left\{\sum_{k=1}^{N-1}\left[b_{m,k}\mathbf{P}_{Bks}(T) + \frac{\partial b_{m,k}}{\partial x}\mathbf{P}_{Bk}(T)\right]e^{-jk\omega_s(nT+T)}\right\},$$

$$m = 2,3,\cdots$$

where $\mathbf{M}_{Bs}(T)$, $\mathbf{P}_{Bs}(T)$, and $\mathbf{P}_{Bk}(T)$ were defined previously and

$$
\begin{aligned}
\mathbf{P}_{Bks}(T) &= \frac{\partial \mathbf{P}_{Bk}(T)}{\partial x} \\
&= \frac{\partial}{\partial x}\mathcal{L}^{-1}\left[\mathbf{A}^{-1}(s)\frac{\mathbf{d}_2}{s - jk\omega_s}\right]_{t=-T} \\
&= \mathcal{L}^{-1}\left[\frac{\partial \mathbf{A}^{-1}(s)}{\partial x}\frac{\mathbf{d}_2}{s - jk\omega_s}\right]_{t=-T} \\
&= \mathbf{P}_{ks}(-T),
\end{aligned}
$$

$$k = 0,1,\cdots,N.$$

The sensitivity immediately after switching is obtained by summing up that of all Volterra circuits

$$\mathbf{z}(nT^+) = \sum_{m=1}^{\infty}\mathbf{z}_m(nT^+).$$

## 6.4 Numerical Examples

The proposed method for computing the time domain sensitivity of periodically switched nonlinear circuits is implemented in the same computer program RSPSN in MATLAB. In this section, the time domain sensitivity of example circuits is analyzed using RSPSN.
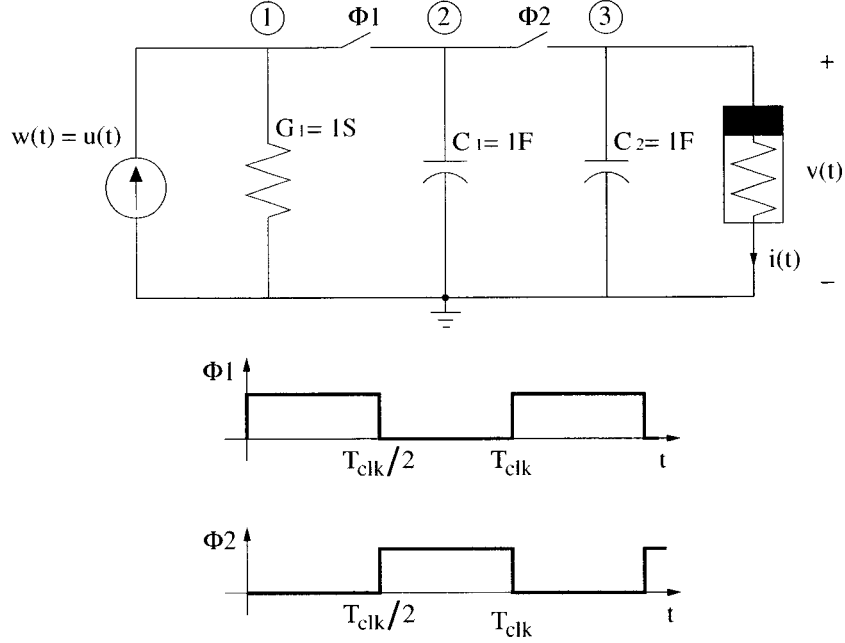
**Figure 6.1:** Switched circuit with nonlinear conductor

The results from RSPSN is validated by comparing with those from the brute-force method.

## 6.4.1 Switched Circuit with Nonlinear Conductors

For the first example, we calculate the time domain sensitivity for the same periodically switched nonlinear circuit used in Chapter 4. The example circuit is redrawn in Fig. 6.1. Same settings are assumed as those used in response simulation before. The sensitivities of all nodes with respect to conductor $G_1$ and capacitor $C_1$ are calculated using RSPSN. The time domain sensitivities of the output voltage at node 3, $z_3(t)$, with respect to $G_1$ and $C_1$ are investigated. They were also calculated using the brute-force method. The time domain sensitivities from both RSPSN and BF are shown in Fig. 6.2 and Fig. 6.3 together for comparison. It is seen that the time domain sensitivities calculated from RSPSN and BF match well. To further assess the accuracy in sensitivity analysis,
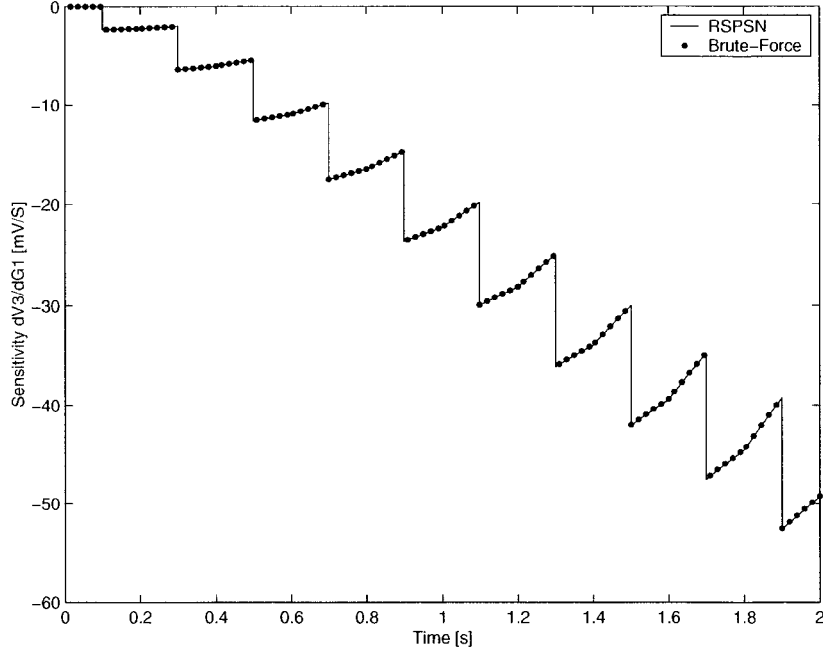
**Figure 6.2:** Sensitivity of $v_3(t)$ with respect to $G_1$

the normalized difference between the results from RSPSN and those from BF for 100 clock cycles are measured. They are plotted in Fig. 6.4 for $G_1$ and Fig. 6.5 for $C_1$, respectively. As shown in the figures, the maximum normalized differences are less than 0.3% for both cases. It is evident that RSPSN yields correct time domain sensitivity analysis results for periodically switched nonlinear circuits.

## 6.4.2 Switched-Capacitor Integrator with Nonlinear Opamp

The second example for the time domain sensitivity analysis of periodically switched nonlinear circuits is a switched-capacitor integrator with nonlinear operational amplifier, as shown in Fig. 6.6. It is the same example circuit used for time domain response analysis of periodically switched nonlinear circuits in Chapter 4. Under the same conditions, the time domain sensitivities of the output voltage $V_o$ with respect to capacitors $C_i$ and $C_f$ are simulated. The results are plotted in Fig. 6.7 and Fig. 6.8. The defini-
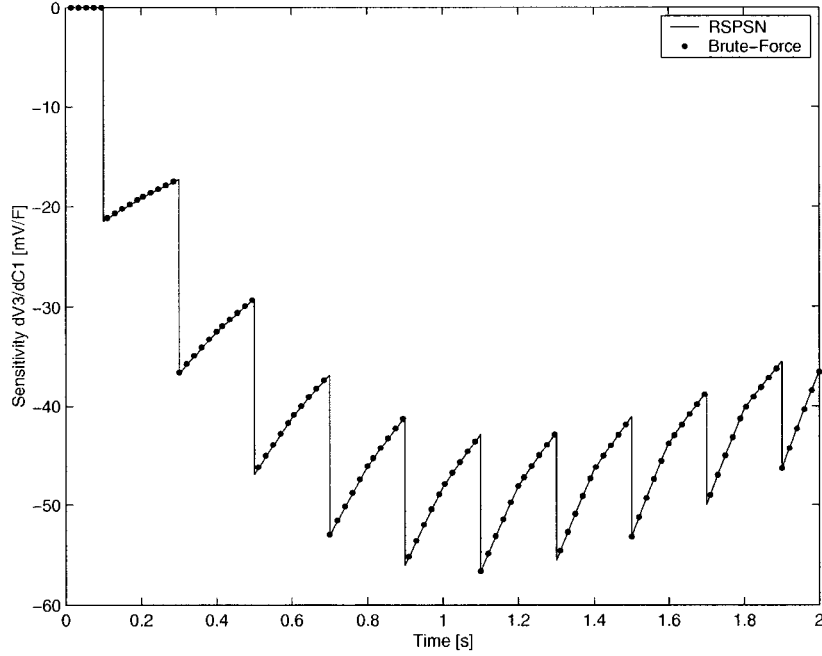
**Figure 6.3:** Sensitivity of $v_3(t)$ with respect to $C_1$

tion of semi-normalized sensitivity is used in these plots. The sensitivities of the output voltage $V_o$ with respect to capacitors $C_i$ and $C_f$ are also computed using the brute-force method. $\dfrac{\Delta x}{x} = 1\%$ is used to calculate $\Delta v(t)$ for the implementation of the brute-force method. The results from BF are plotted in Fig. 6.9 and Fig. 6.10 together with those from RSPSN for comparison. An enlarged plot of Fig. 6.10 is redrawn in Fig. 6.11. Good agreements are shown for the time domain sensitivities obtained from RSPSN and BF in these figures.

## 6.5 Summary

In this chapter, a new method has been proposed for time domain sensitivity analysis of periodically switched nonlinear circuits. Inside the clock phases, time domain sensitivity is obtained by summing up the sensitivities of Volterra circuits of a nonlinear circuit.

64
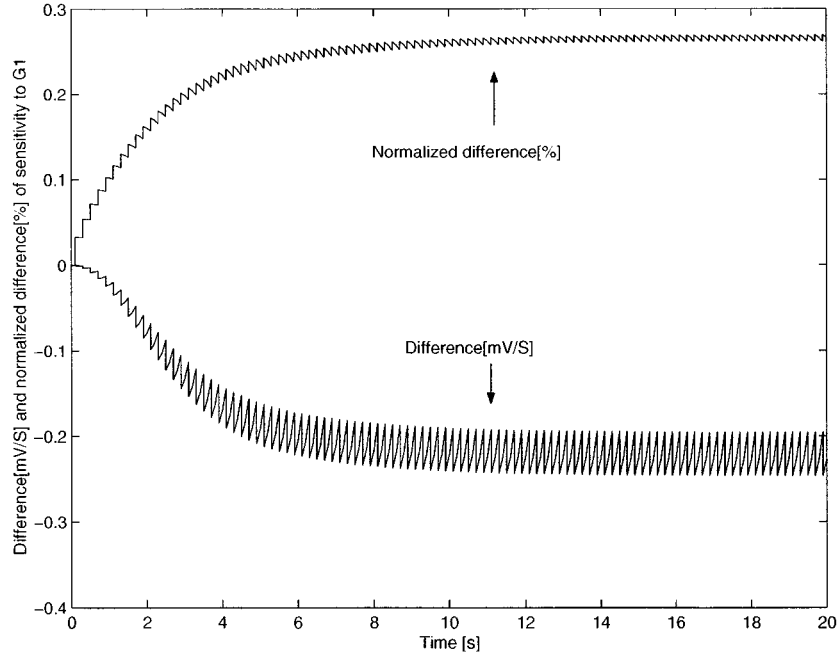
**Figure 6.4:** Difference between RSPSN and BF for sensitivity of $v_3(t)$ with respect to $G_1$

Using the extended two-step algorithm presented in Chapter 4, the method can correctly solve the discontinuity problems that might occur at switching instants for periodically switched nonlinear circuits. Numerical results were given to assess the proposed method.
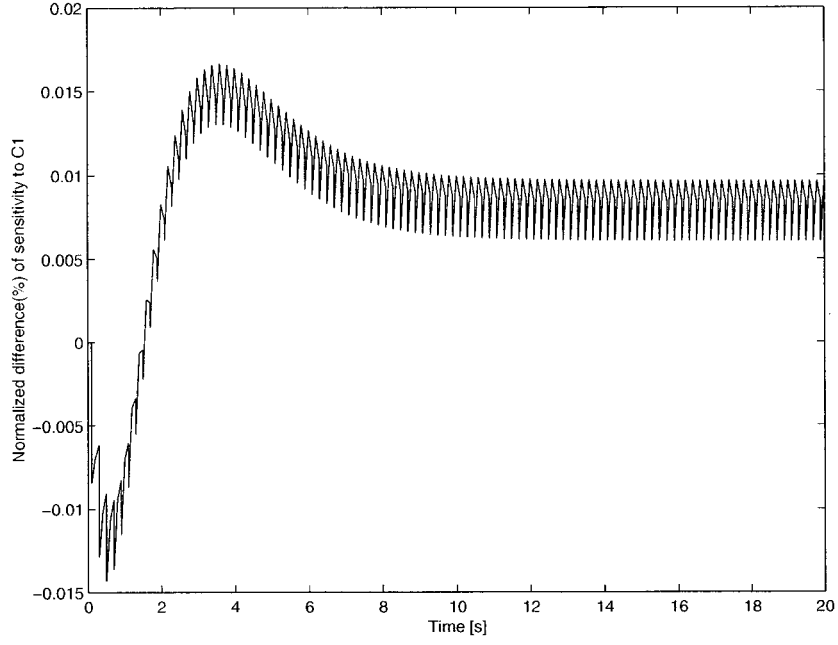
**Figure 6.5:** Difference between RSPSN and BF for sensitivity of $v_3(t)$ with respect to $C_1$
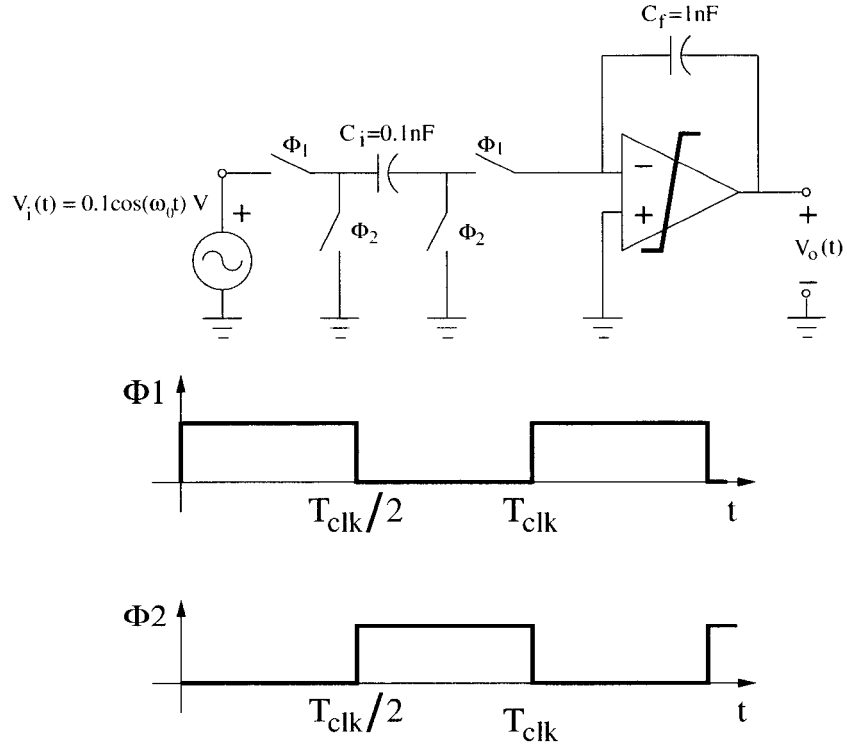


**Figure 6.6:** Switched-capacitor integrator with nonlinear opamp

**Figure 6.7:** Sensitivity of $V_o$ with respect to $C_i$



**Figure 6.8:** Sensitivity of $V_o$ with respect to $C_f$

67

**Figure 6.9:** Sensitivity of $V_o$ with respect to $C_i$ from RSPSN and BF
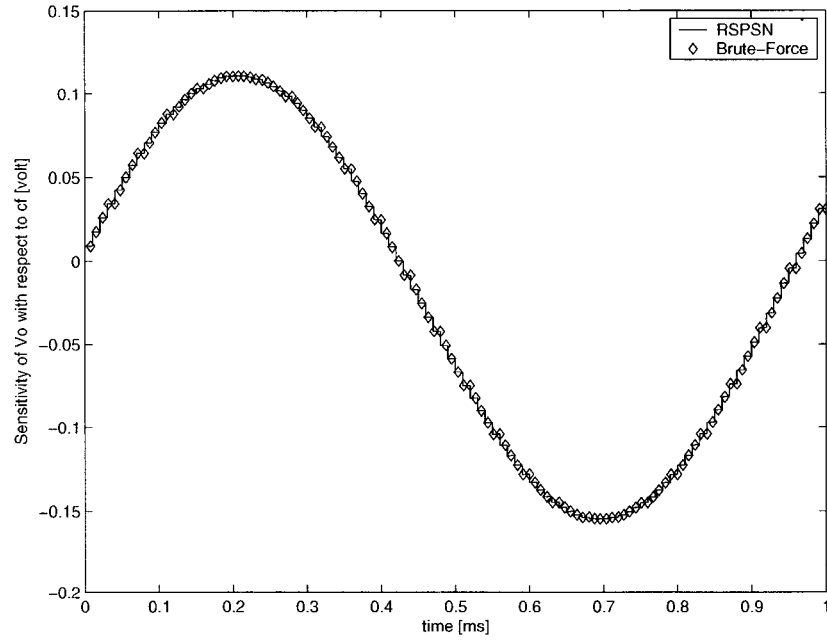


**Figure 6.10:** Sensitivity of $V_o$ with respect to $C_f$ from RSPSN and BF

68
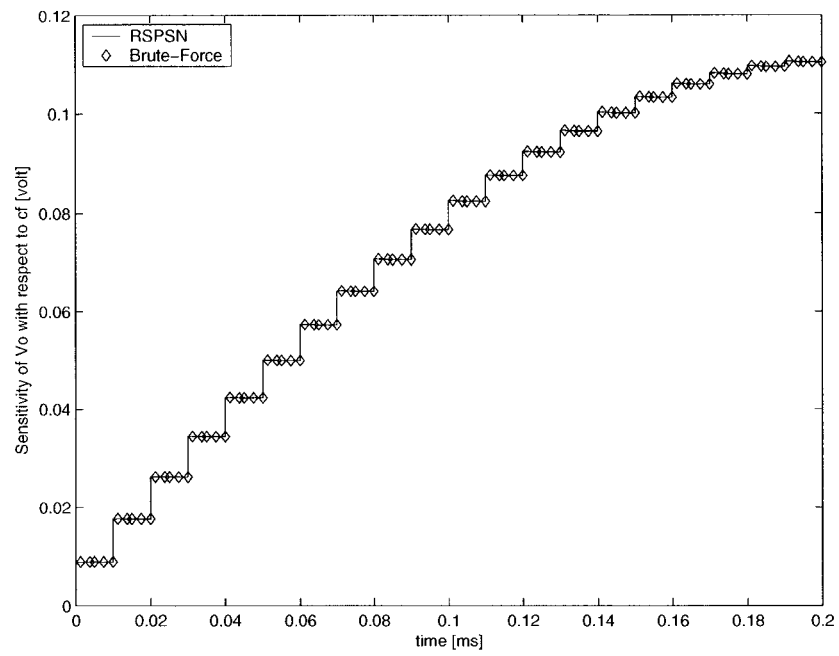
**Figure 6.11:** Sensitivity of $V_o$ with respect to $C_f$ from RSPSN and BF (enlarged)

# Chapter 7

# Analysis of Nonlinear Oversampled Sigma-Delta Modulators

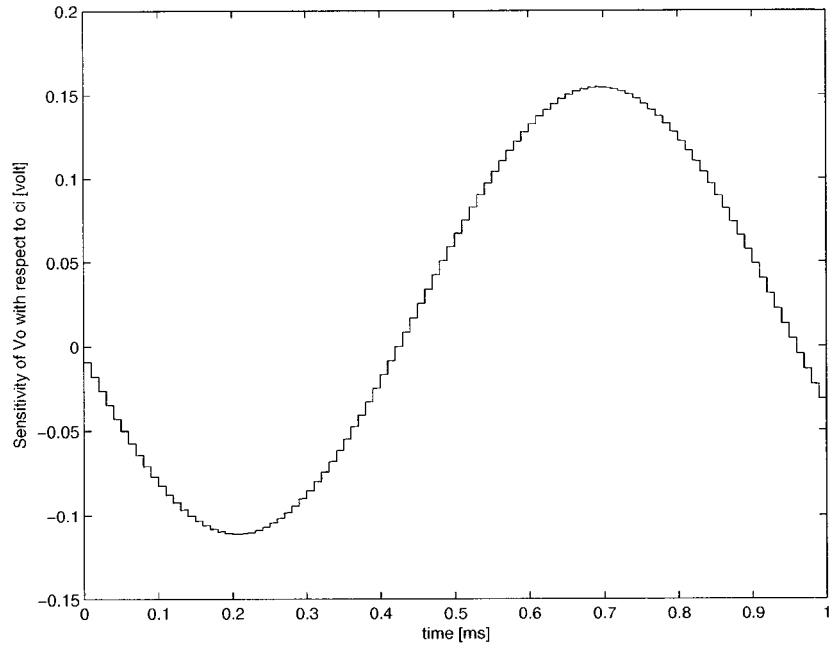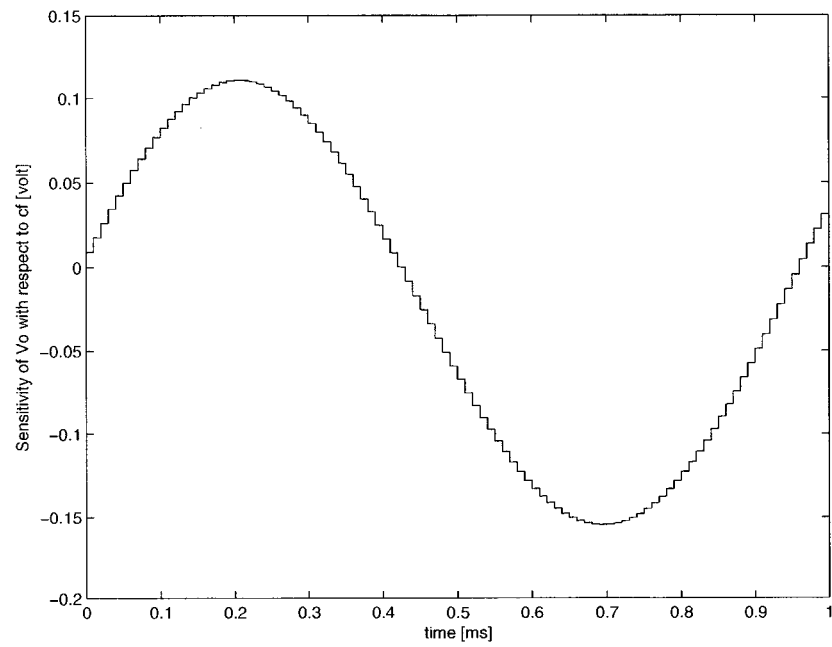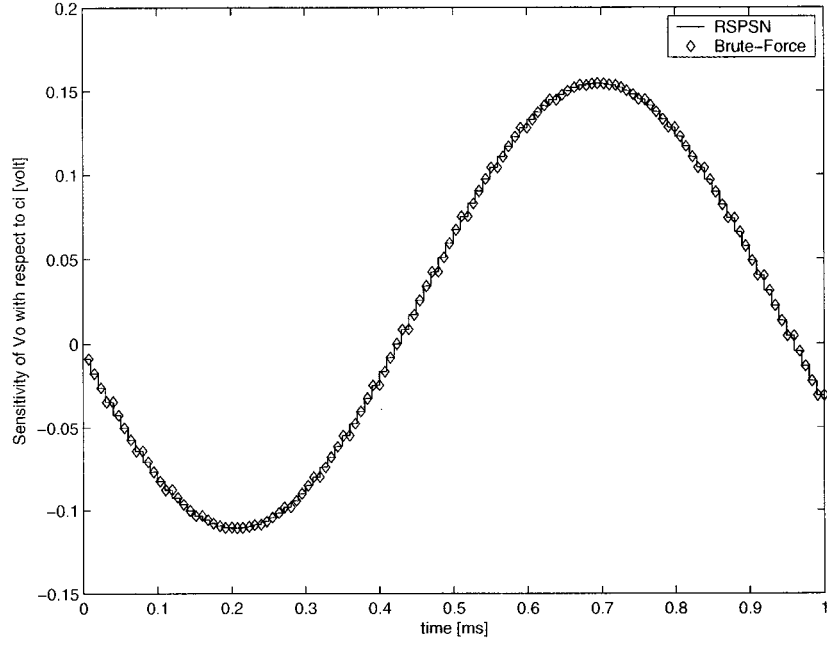One of the important applications of the algorithms developed in preceding chapters is the analysis of nonlinear oversampled sigma-delta modulators. In this chapter, we present a new simulation method for nonlinear oversampled sigma-delta modulators. The time domain response of sigma-delta modulator is analyzed using the proposed method and the output spectrum of the sigma-delta modulators is obtained by taking FFT on the time domain response.

In Section 7.1, the theory of oversampled sigma-delta modulation is briefly reviewed. Challenges for analysis of nonlinear oversampled sigma-delta modulators are described in Section 7.2. The new method for simulation of nonlinear oversampled sigma-delta modulators is presented in Section 7.3. Numerical examples are given in Section 7.4. The chapter is summarized in Section 7.5.

## 7.1   Introduction

Oversampled sigma-delta modulation technique has become popular for achieving high resolution in analog-to-digital and digital-to-analog data conversion. The most impor-

tant advantage of this technique is that high resolution data conversion is achieved using only low resolution, usually a 1-bit A/D converter and the precision required for the analog processing circuits is much lower than the resolution of the overall data converter [26]. Although the concepts of sigma-delta modulation have existed for many years, this technique has become more attractive only after the recent development of the modern VLSI technology, which has made it possible to process the digital signals in adequately high speed.

There are two important techniques employed in oversampled sigma-delta modulation. One is oversampling and the other is noise shaping. Most conventional A/D converters, such as the successive approximation, sub-ranging, and flash converters, sample the signals at, or slightly above the Nyquist rate. Consequently, they are often referred to as Nyquist rate converters. For this kind of converters, the sampled signal is quantized at the full precision or resolution of the converter. So the resolution of these converters is limited by the technology in which the circuits are fabricated. It is extremely difficult to attain high resolution for Nyquist rate converters. Oversampling technique [27] improves the resolution by sampling the signal with a bandwidth of $f_B$ at a frequency $f_s$ that is significantly higher than the Nyquist frequency $f_N = 2f_B$. In this case, the same amount of quantization noise power is spread over a bandwidth of $f_s$, which is much larger than the signal bandwidth $f_B$. The amount of noise power fallen in the signal band is only a small fraction of the total amount of noise power, which is almost entirely inside the signal band for Nyquist rate conversion. The noise power outside the signal band can be effectively attenuated using a low-pass filter. One limit of oversampling technique is the maximum sampling frequency that can be realized in a particular fabrication technique.

To further improve the resolution, another technique known as noise shaping of sigma-delta modulation has been used [26]. In this technique, different transfer characteristics are applied to the input signal and the quantization noise, such that the signal is left undisturbed but the noise power is shaped out of the signal band to allow a high resolution of output. The process of noise shaping can be viewed as pushing the quantization noise power form the signal band to other frequencies. In summary, to achieve high resolution, oversampling reduces the quantization noise power in the signal band by spreading the total amount of the quantization noise power over a bandwidth that is much larger than the signal band and noise shaping further attenuates the noise inside the signal band and amplifies it outside the signal band.

Oversampled sigma-delta modulation technique trades off speed for resolution and shifts the complexity from analog circuitry to digital circuitry. The price paid for low precision of analog circuitry and high resolution of data conversion is the increased complexity of digital circuitry and low-to-medium signal bandwidth. The benefit brought by low precision requirement of analog circuitry is that the circuit can be fabricated in a digital compatible technology.

Oversampled sigma-delta modulators can be implemented in either discrete-time or continuous-time circuits. Discrete time implementations, especially switched-capacitor sigma-delta modulators, gained most of the attention in the past. Switched-current and continuous-time sigma-delta modulators have experienced rapid development in recent years.

## 7.2  Analysis Challenges

The difficulties associated with analysis of nonlinear oversampled sigma-delta modulators are as follows:

1. Time domain analysis is the only possibility, because a harsh nonlinear component - quantizer - is in the feedback loop of sigma-delta modulators.

2. The circuit is a dual time system with a high frequency sampling clock and a low frequency input signal. The results of a few signal cycles are obtained from the simulation of thousands of clock cycles.

3. Because oversampled sigma-delta modulators are high resolution systems, high simulation accuracy is required.

4. General circuit nonlinearities need to be handled.

Although SPICE type transistor level simulation tools, which are widely used for circuit analysis, can be used in investigation of various circuit imperfections and non-linearities, they reach their limits for simulation of oversampled sigma-delta modulators [28]. Since both high accuracy and large amount of data are required, this kind of simulation often results in extremely long CPU time, which might be counted in days. Efficient simulation methods are needed for analysis of oversampled sigma-delta modulators, especially with the ability to investigate the circuit imperfections and nonlinearities.

Oversampled sigma-delta modulators are simulated in different model abstraction levels. Tradeoffs between simulation time and accuracy are often made in different phases during the design cycle. Block level simulation is fast and only used in conceptual design phase [29]. Because it is highly abstract and no circuit implementation details are

available, any circuit imperfection and nonlinearity can not be investigated in this level. In circuit design phase, electrical circuit level simulation is used to consider the value of circuit parameters, such as the value of resistors and capacitors and the gain of an opamp. Because of their nonlinear nature, sigma-delta modulators are only simulated in time domain. The sampled data simulation proposed in [5] was applied for fast and accurate simulation of oversampled sigma-delta modulators. But it is based on the simulation of linear circuits and only a restricted set of nonlinear elements, whose characteristics changes only at switching instants, can be included. Many other circuit nonlinearities can not be investigated.

## 7.3   Time Domain Simulation Method

Time domain simulation of nonlinear oversampled sigma-delta modulators demands for a method that addresses the issues of efficiency and accuracy simultaneously and possesses the ability of handling general nonlinearities encountered in circuit analysis. The algorithms presented in preceding chapters are efficient and accurate. They are applicable to the simulation of nonlinear oversampled sigma-delta modulators.

To illustrate the simulation method, we take the first order switched-capacitor sigma-delta modulator in Fig 7.1 for example. In this circuit, all the switches and the comparator are controlled by a external clock with two phases. The circuit can be partitioned into two parts. One part is a nonlinear switched-capacitor integrator, which is a typical periodically switched nonlinear circuit. The other part is a clocked 1-bit quantizer or comparator. Its output changes only at the clock edges based on its input at the same time. It is modeled as a clocked nonlinear voltage controlled voltage source with two distinct output levels.
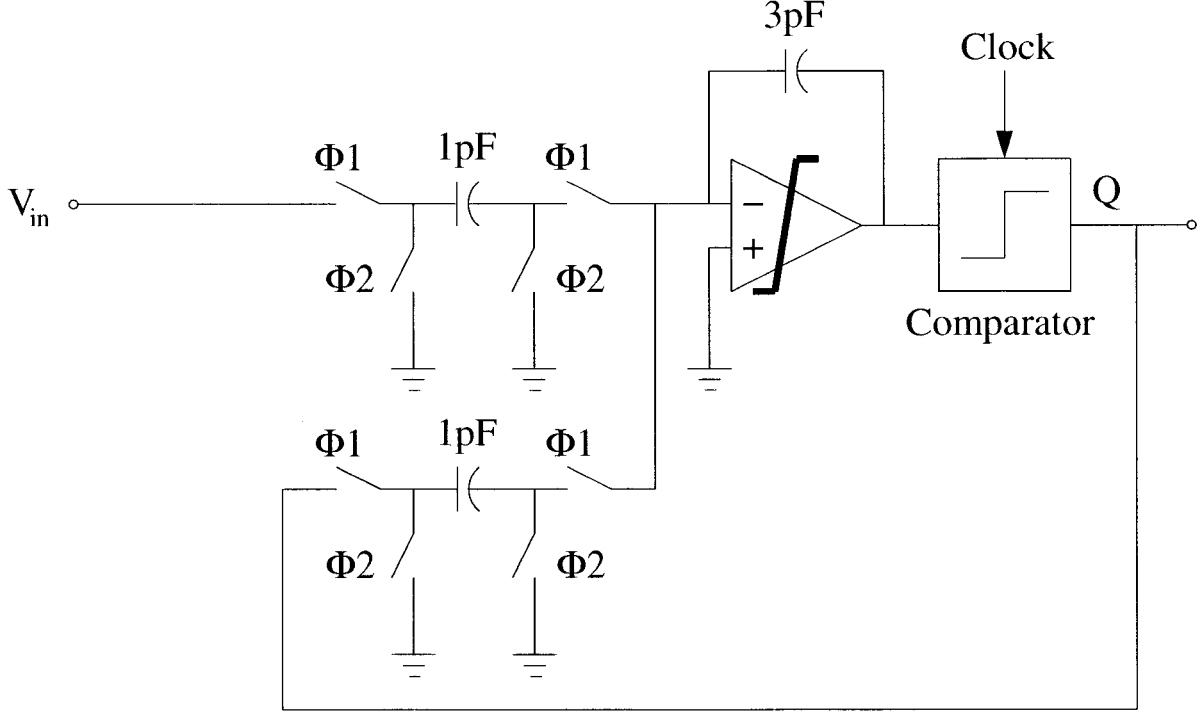
74

**Figure 7.1:** First order switched-capacitor sigma-delta modulator with nonlinear opamp

Time domain simulation of nonlinear oversampled sigma-delta modulators proceeds as follows. Inside a clock cycle, the integrator circuit is simulated using the proposed algorithms for periodically switched nonlinear circuits. At the end edge of the clock, the output of the integrator is sampled to update the state of the comparator, which decides the feedback voltage level to the input of the integrator for the next clock cycle. The feedback voltage level is not changed over a clock cycle even though the input of the comparator might change. The output of the integrator is needed only at the clock edges when the comparator samples its input. Between the clock edges, the output of the integrator changes but does not affect the operation of the sigma-delta modulator. During simulation, two types of input for the integrator are considered. The primary input is the signal to be processed and it is normally a sinusoidal function. The secondary input is the feedback voltage form the comparator. Since it is a constant over a clock

cycle, it can be treated as a step input for the clock duration. After the simulation of one clock cycle, same process is repeated for the next clock cycle.

The method can be used to investigate the effect of any circuit imperfection and nonlinearity on the performance of nonlinear oversampled sigma-delta modulators, such as nonlinear characteristics of circuit components, nonlinear gain of opamps, and the integrator leakage effect [30].

## 7.4   Numerical Examples

The computer program RSPSN is used for time domain simulation of nonlinear over-sampled sigma-delta modulators. In this section, we give the simulation results for a few sigma-delta modulators to show the usefulness of the proposed method. The switched-capacitor sigma-delta modulator are simulated with the assumption that all switches are ideal and each clock period has two non-overlapping phases with equal width. Simulation was performed in Matlab on Sun Ultra-10 workstations with 440MHz CPU and 512MB RAM and the execution CPU time was measured under this conditions.

### 7.4.1   Switched-Capacitor Nonlinear Sigma-Delta Modulators
**First Order Nonlinear Sigma-Delta Modulators**

First, the first order switched-capacitor sigma-delta modulator shown in Fig 7.1 is simulated. All switches and the comparator is clocked at 1.024 MHz. The input signal is a 1 kHz sinusoid of amplitude of 2 V. The values of capacitors are labeled in the figure and zero initial conditions are assumed for them. The gain of the opamp is modeled as

$$Gain = a_0 + a_1(v_+ - v_-) + a_2(v_+ - v_-)^2,$$

where $a_0, a_1, a_2$ are constants. In our implementation, they were chosen as
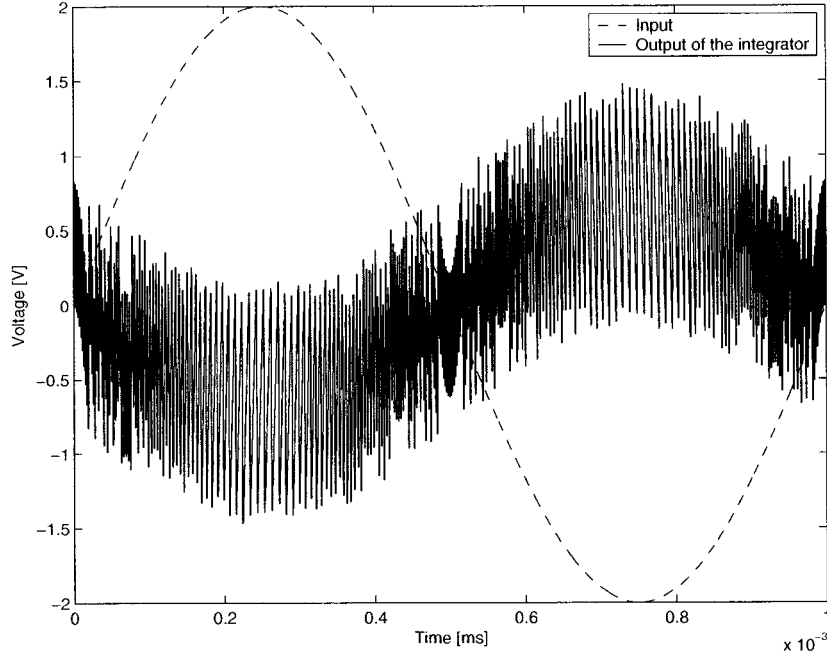
**Figure 7.2:** Time domain waveform at output of the integrator

$$\{a_0, a_1, a_2\} = \{10000, -5000, -1000\}.$$

The time domain output of the integrator is shown in Fig 7.2 for one signal cycle. The spectrum at the output of the modulator is shown in Fig 7.3, which shows many tones exist in the output spectrum of the first order sigma-delta modulator. These results are obtained by simulating the circuit for 75776 clock cycles, discarding the first 10240 data points to remove any circuit transient, and performing a FFT on the remaining 65536 data points. It takes RSPSN 19.44 minutes of CPU time to simulate 75776 clock cycles.
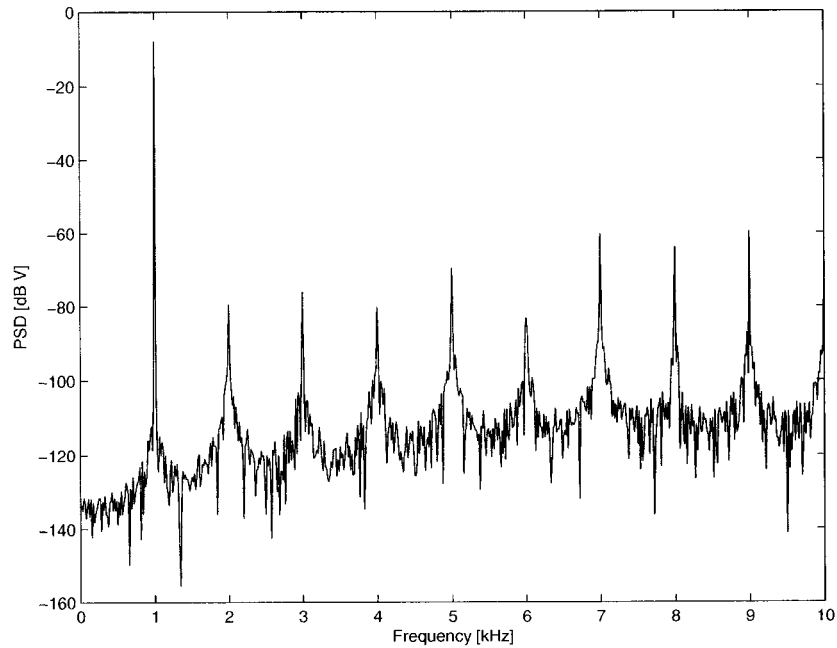
**Figure 7.3:** Output spectrum of the first order switched-capacitor sigma-delta modulator

## Second Order Nonlinear Sigma-Delta Modulators

A second order switched-capacitor sigma-delta modulator shown in Fig 7.4 is simulated. The clock frequency is 1.024 MHz and the input signal is a 1 kHz sinusoid of amplitude of 1 V. The two opamps are modeled in the same way as that of the last example. Time domain response at the output of the second integrator is shown in Fig 7.5. The spectrum at the output of the modulator is shown in Fig 7.6, which shows the noise shaping characteristics of the second order sigma-delta modulator. The results is obtained by the same means used in the previous example. For the second order switched-capacitor sigma-delta modulator, it takes RSPSN 34.70 minutes of CPU time to simulate 75776 clock cycles.
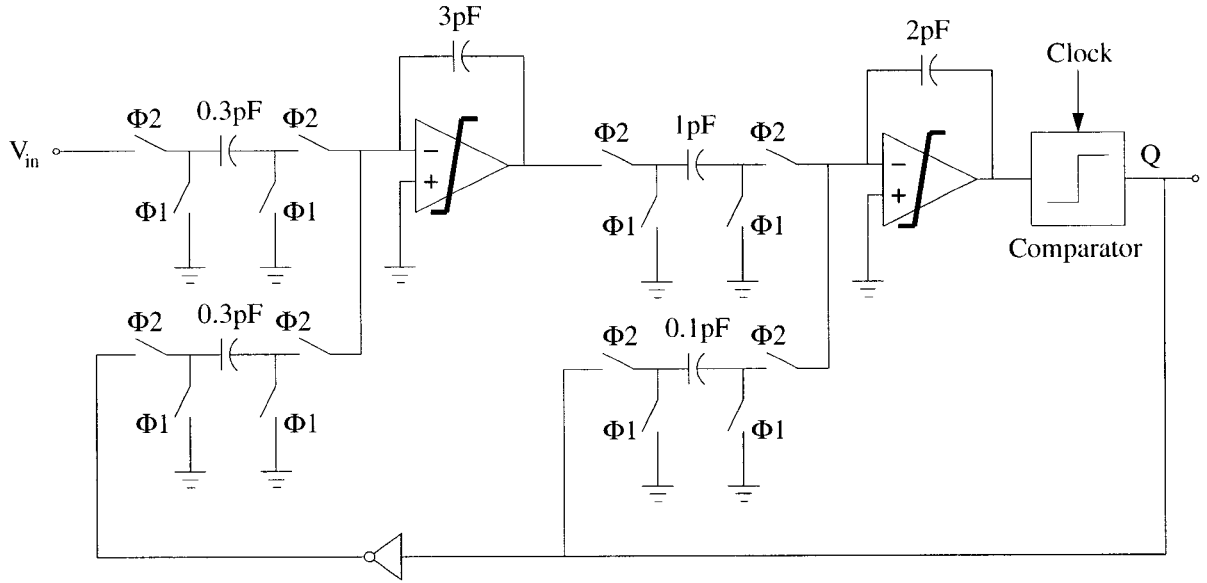
**Figure 7.4:** Second order switched-capacitor sigma-delta modulator with nonlinear opamp

## 7.4.2 Continuous-Time Oversampled Sigma-Delta Modulators

Although it is mainly targeted on and used in the analysis of periodically switched non-linear circuits, the proposed method is applicable to the simulation of continuous-time oversampled sigma-delta modulators. The continuous-time second order sigma-delta modulator shown in Fig 7.7 is originally used in [5]. In this circuit, the opamp is ideal, the comparator is clocked at 1.024 MHz, and the input signal is a 1 kHz sinusoid of amplitude of 2 V. The values of resistors and capacitors are labeled in the figure. This example is simulated using RSPSN. Time domain response at output of the second integrator is shown in Fig 7.8. The spectrum at the output of the modulator is shown in Fig 7.9, which shows the noise shaping characteristics of the second order sigma-delta modulator. For the continuous-time second order sigma-delta modulator, it takes RSPSN 23.2 seconds of CPU time to simulate 75776 clock cycles. Same results from RSPSN are obtained as those published in [5]. This evidences the usefulness of the proposed method in simulation of both switched and continuous-time oversampled sigma-delta modulators.
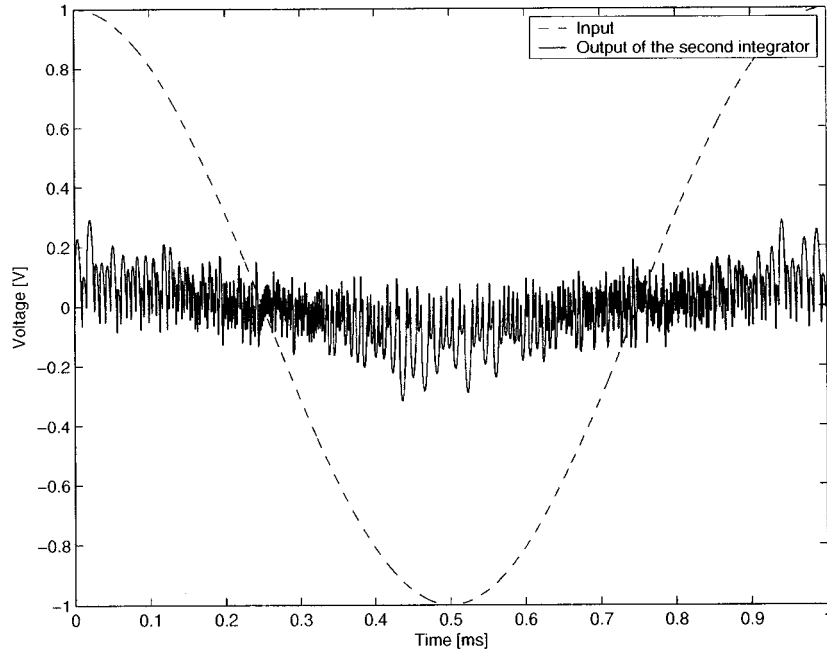
**Figure 7.5:** Time domain waveform at output of the second integrator

## 7.5 Summary

In this chapter, a new simulation method for nonlinear oversampled sigma-delta modulators has been proposed. Several oversampled sigma-delta modulators were simulated. Efficiency and accuracy of the method have been evidenced by measuring the CPU time and comparing some of the simulation results with the published results given in [5], under the same conditions.

**Figure 7.6:** Output spectrum of the second order switched-capacitor sigma-delta modulator



**Figure 7.7:** Continuous-time second order sigma-delta modulator ($V_{ref} = 2.5V$)

81

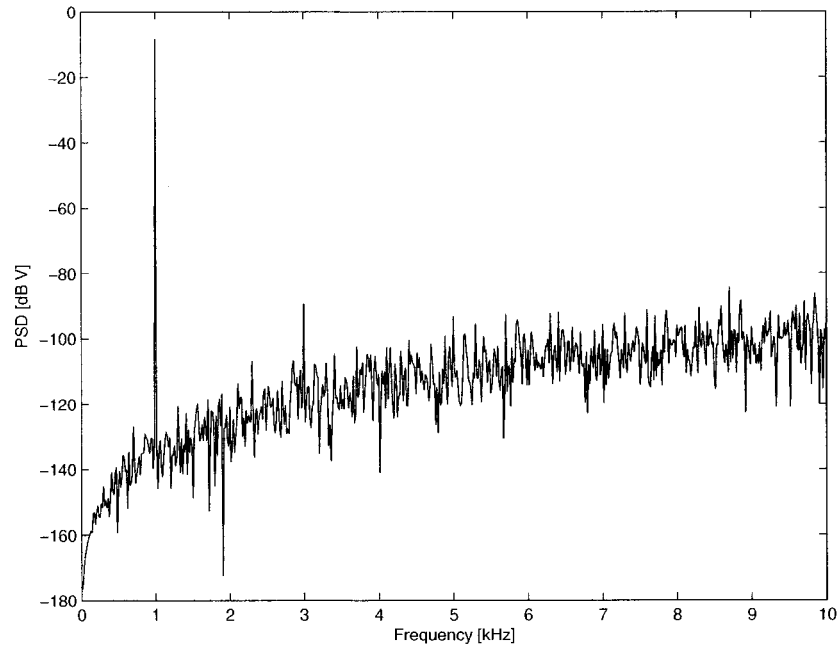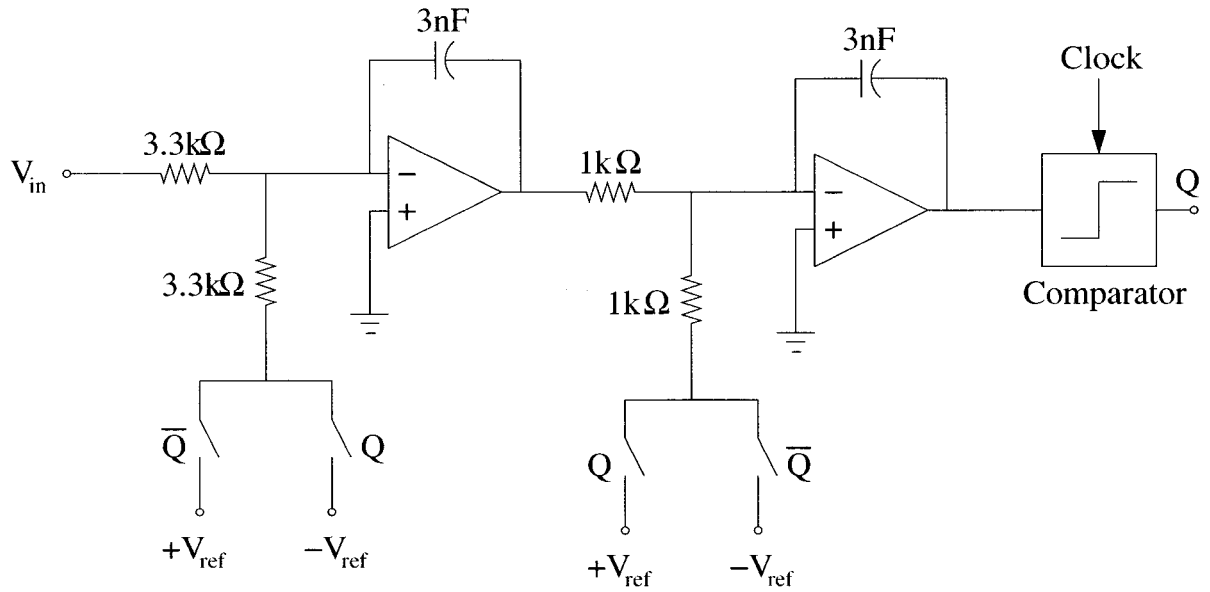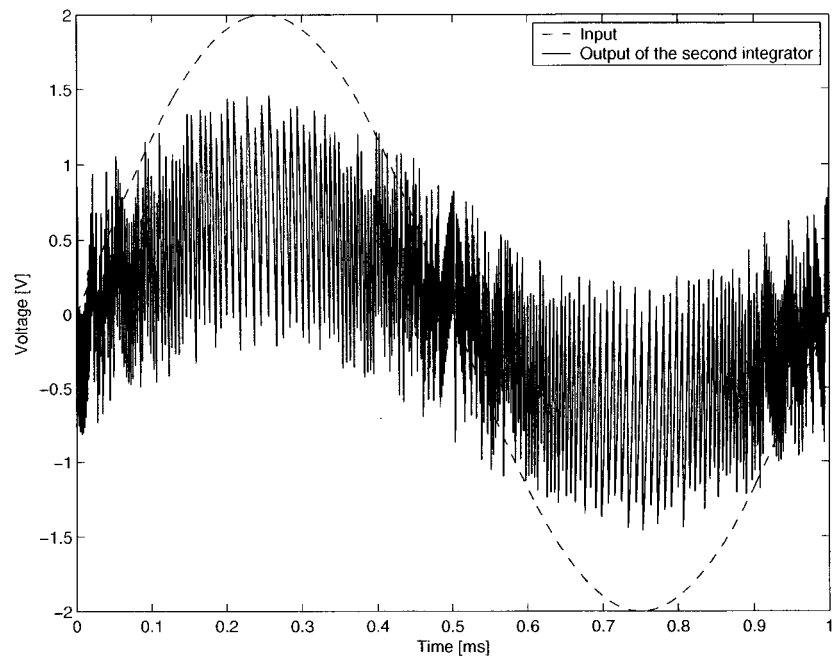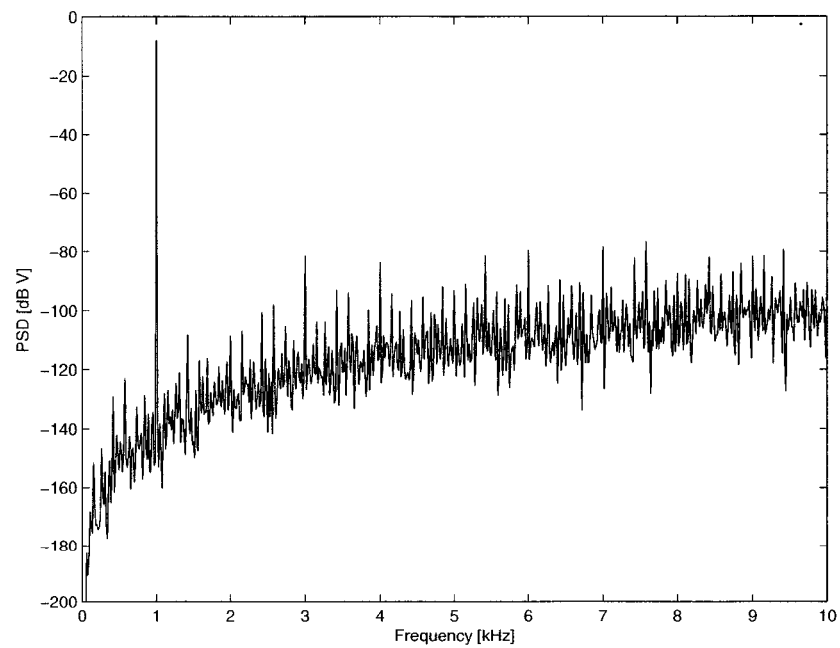**Figure 7.8:** Time domain waveform at output of the second integrator



**Figure 7.9:** Output spectrum of the continuous-time second order sigma-delta modulator

82

# Chapter 8

# Conclusions

## 8.1   Summary of the thesis

A new and efficient time-domain response and sensitivity analysis method for period-
ically switched nonlinear circuits has been presented. By depicting the behavior of
these circuits using Volterra functional series and approximating the input of high-
order Volterra circuits using interpolating Fourier series, a new sampled data simulation
method for analysis of response and sensitivity of periodically switched nonlinear circuits
has been developed. To handle the inconsistent initial conditions encountered in anal-
ysis of these circuits, a new two-step algorithm has been proposed. The response and
sensitivity of periodically switched nonlinear circuits at equally spaced time points are
obtained efficiently and accurately. The superior computational efficiency of the method
is achieved by pre-computing a set of constant matrices, vectors, and parameters prior
to the start of simulation. Although the pre-processing step is costly, this initial cost
can quickly be amortized if the response and sensitivity over a long period of time are
needed. LMS-PC methods, however, not only can not handle inconsistent initial con-
ditions but also require computationally expensive Newton-Raphson iterations in each
step of integration, resulting in poor simulation accuracy and excessive CPU time.

The method is most effective for periodically switched nonlinear circuits with mild nonlinearities whose behavior can be characterized sufficiently using the low-order Volterra series expansion of the network variables. The accuracy of the method depends on the order of the Volterra series used in depicting the network variables of the nonlinear circuits and the order of interpolating Fourier series.

One of the important applications of the method is simulation of nonlinear oversampled sigma-delta modulators. Using this method, effects of general circuit nonlinearities on the performance of oversampled sigma-delta modulators can be investigated in circuit level.

## 8.2  Future Work

Many interesting subjects can be explored in this area. The work can be extended in two major directions. One is the study of computational algorithms for time domain noise analysis of periodically switched nonlinear circuits. Since noise normally is very small compared with the deterministic signals, efforts are needed to further improve the accuracy of the computational algorithms for noise analysis. Another practical direction is the investigation of various nonlinear effects on the performance of oversampled sigma-delta modulators, such as nonlinear characteristic of circuit components, nonlinear gain of the opamps, and the integrator leakage effect. During the design of oversampled sigma-delta modulators, the stability and undesirable tones of the modulator are two major concerns. They can not be studied only using the methods based on linear models. Using the method proposed in this thesis and modeling the nonlinearity in circuit level, many nonlinear effects can be investigated and useful information about these issues can be gained for the design of oversampled sigma-delta modulators.

# Appendix A

# Computation of $\mathbf{M}(T)$, $\mathbf{P}(T)$, $\mathbf{P}_k(T)$, and their Derivatives

Numerical Laplace inversion was used to compute $\mathbf{M}(T)$, $\mathbf{P}(T)$, $\mathbf{P}_k(T)$, $k = 0, 1, ..., N$, and their derivatives with respect to circuit elements. Numerical Laplace inversion is an A-stable high-order numerical integration technique [16]. It computes time-domain response $\mathbf{v}(t)$ from its Laplace transform $\mathbf{V}(s)$

$$\mathbf{v}(t) = -\frac{1}{t} \sum_{i=1}^{J} K_i \mathbf{V}\left(\frac{z_i}{t}\right),$$

where $z_i$ and $K_i$ are poles and residues of the Padé polynomial that approximates $e^z$ in the vicinity of the origin, and $J$ is the order of numerical Laplace inversion. $z_i$ and $K_i$ are calculated with high precision [16]. Padé polynomials and its applications in numerical Laplace inversion have been covered extensively in [18, 19, 16]. Interesting readers are referred to the cited references for details. Because $\mathcal{L}[\delta(t)] = 1$ and $\mathcal{L}[u(t)] = \frac{1}{s}$, where $\delta(t)$ and $u(t)$ are Dirac impulse function and unit step function respectively, the difficulties associated with discontinuities encountered at switching instants, which are difficult to be handled by conventional numerical integration methods, can be handled with ease in Laplace domain. In our implementation, the order of integration is 19. In

85

addition, the multi-step numerical Laplace inversion algorithm given in [11] is extended here to compute the derivatives of $\mathbf{M}(T)$, $\mathbf{P}(T)$, $\mathbf{P}_k(T)$, $k = 0, 1, \cdots, N$ with respect to circuit elements. To minimize the error due to large displacement from the origin, as shown in [16], we divide the step $T$ into $N$ subintervals of equal width $\Delta t = T/N$. It was shown in [11] that

$$\mathbf{N}[(n + 1)\Delta t] = \mathbf{Q}^n(\Delta t)\mathbf{N}(\Delta t).$$

$$\mathbf{P}[(n + 1)\Delta t] = \mathbf{P}(\Delta t)e^{j\omega_o n\Delta t} + \mathbf{Q}(\Delta t)\mathbf{P}(n\Delta t), \qquad (\text{A.1})$$

where

$$\mathbf{N}(\Delta t) = -\frac{1}{\Delta t}\sum_{i=1}^{J} K_i \mathbf{A}_i^{-1}\left(\frac{z_i}{\Delta t}\right).$$

$$\mathbf{P}(\Delta t) = -\frac{1}{\Delta t}\sum_{i=1}^{J} K_i \mathbf{A}_i^{-1}\left(\frac{z_i}{\Delta t}\right)\left[\frac{\mathbf{g}}{\frac{z_i}{\Delta t} - j\omega_o}\right],$$

$\mathbf{N}(n\Delta t)$ relates to $\mathbf{M}(n\Delta t)$ by $\mathbf{M}(n\Delta t) = \mathbf{N}(n\Delta t)\mathbf{C}$, and $\mathbf{Q}(\Delta t) = \mathbf{N}_k(\Delta t)\mathbf{C}$ is constant if $\Delta t$ is fixed. The derivatives with respect to $x$ are obtained by differentiating (A.1) $x$ directly

$$\frac{\partial \mathbf{N}[(n + 1)\Delta t]}{\partial x} = n\mathbf{Q}^{n-1}(\Delta t)\frac{\partial \mathbf{Q}(\Delta t)}{\partial x}\mathbf{N}(\Delta t) + \mathbf{Q}^n(\Delta t)\frac{\partial \mathbf{N}(\Delta t)}{\partial x},$$

$$\frac{\partial \mathbf{P}[(n + 1)\Delta t]}{\partial x} = \frac{\partial \mathbf{P}(\Delta t)}{\partial x}e^{j\omega_o n\Delta t} + \frac{\partial \mathbf{Q}(\Delta t)}{\partial x}\mathbf{P}(n\Delta t) + \mathbf{Q}(\Delta t)\frac{\partial \mathbf{P}(n\Delta t)}{\partial x}, \qquad (\text{A.2})$$

It is seen that both the matrices and their derivatives are obtained in a recursive manner efficiently.

# Bibliography

[1] D. Bedrosian and J. Vlach, "Time-domain analysis of networks with internally controlled switches," *IEEE Trans. on Circuits and Systems I*, Vol.39, No. 3, pp. 199-212, March 1992.

[2] D. Bedrosian and J. Vlach, "Analysis of switched networks," *International Journal of Circuit Theory and Applications*, vol. 20, No. 3, pp. 309-325, May-June 1992.

[3] D. Foty, *MOSFET modeling with SPICE - principles and practice*, New Jersey : Prentical-Hall, 1997.

[4] J. Sheieh, M. Patel, and B. Sheu, "Measurements and Analysis of Charge Injection in MOS Analog Switches," *IEEE Journal of Solid-State Circuits*, vol. 22, pp 277-281, April 1987.

[5] A. Opal, "Sampled data simulation of linear and nonlinear circuits," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, Vol.15, No. 3, pp. 295-307, March 1996.

[6] F. Yuan and A. Opal, "Distortion analysis of periodically switched nonlinear circuits using time-varying Volterra series," *IEEE Transactions on Circuits and Systems I - Fundamental Theory and Applications*, Vol 48, No. 6, pp. 726-738, June 2001.

[7] *Spectre User's Guide*, Cadence Design Systems, Inc., 1998.

[8] R. Kielkowski, *Inside Spice*. New York : McGraw-Hill, 1998.

[9] P. Wampacq and W. Sansen, *Distortion Analysis of Analog Integrated Circuits*. Boston, MA : Kluwer, 1998.

[10] J. Valas and J. Vlach, "SWANN - A programme for analysis of switched analogue nonlinear networks," *Int'l J. Circuit Theory and Application*, Vol. 23, pp. 369-379, 1995.

[11] F. Yuan and A. Opal, "An efficient transient analysis algorithm for mildly nonlinear circuits," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 21, No. 6, pp. 662-673, June 2002.

[12] A. Opal and J. Vlach, "Analysis and sensitivity of periodically switched linear networks," *IEEE Trans. on Circuits and Syst.*, Vol. 36, No. 4, pp. 522-532, April 1989.

[13] F. Yuan and A. Opal, "A unified sampled-data simulation method for nonlinear circuits response, sensitivity, and stochastic behavior," *Proceedings of the 44th IEEE 2001 Midwest Symposium on Circuits and Systems*, Vol. 1, pp. 28-32, Aug. 2001

[14] F. Yuan and A. Opal, "Noise and sensitivity analysis of periodically switched linear circuits in frequency domain," *IEEE Transactions on Circuits and Systems I - Fundamental Theory and Applications*, Vol 47, No. 7, pp. 986-998, July 2000.

[15] B. Raahemi and A. Opal, "Time domain sensitivity of linear circuits using sampled data simulation," *IEEE Transactions on Circuits and Systems I - Fundamental Theory and Applications*, Vol 47, No. 6, pp. 948-957, June 2000.

[16] J. Vlach and K. Singhal, *Computer methods for circuit analysis and design*, 2nd edition, New York : Van Nostrand Reinhold, 1994.

[17] J. Vlach, J. Wojciechowski, and A. Opal, "Analysis of nonlinear networks with inconsistent initial conditions," *IEEE Trans. on Circuits and Systems I*, Vol. 42, No. 4, pp. 195-200, April 1995.

[18] A. Opal and J. Vlach, "Consistent initial conditions of linear switched networks," *IEEE Trans. on Circuits and Systems*, Vol. 37, No. 3, pp. 364-372, March 1990.

[19] G. Baker, Jr., *Essentials of Padé approximation*, New York : Academic Press, 1975.

[20] M. Schetzen, *The Volterra and Wiener theory of nonlinear systems*, New York : John Wiley and Sons, 1981.

[21] M. Schetzen, "Multilinear theory of nonlinear networks," *Journal of the Franklin Institute*, Vol. 320, pp. 221-247, November 1985.

[22] R. Tymerski, "Volterra series modeling of power conversion systems," *IEEE Transactions on Power Electronics*, Vol. 6, No. 4, pp. 712-718, October 1991.

[23] W. Yu, S. Sen, and B. Leung, "Distortion analysis of MOS track-and-hold sampling mixers using time-varying Volterra series," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 46, No. 2, pp. 101-113, February 1999.

[24] A. Opal and J. Vlach, "Consistent initial conditions of nonlinear networks with switches," *IEEE Trans. on Circuits and Systems*, Vol. 38, No. 7, pp. 698-710, July 1991.

[25] K. Singhal and J. Vlach, "Interpolation using the fast Fourier transform," *Proceeding of IEEE*, pp. 1558, December 1972.

[26] J. C. Candy, and G. C. Temes, "Oversampling methods for A/D and D/A conversion," in *Oversampling Delta-Sigma Data Converters: Theory, Design and Simulation*, pp. 1-25, IEEE Press, New York, 1992.

[27] P. M. Aziz, H. V. Sorensen, and J. Van der Spiegel, "An overview of sigma-delta converters," *IEEE Signal Processing Magazine*, Vol. 13, No. 1, pp. 61-84, January 1996.

[28] G. G. E. Gielen and J. E. Franca, "CAD tools for data converter design: An overview," *IEEE Trans. on Circuits and Systems II*, Vol. 43, No. 2, pp. 77-89, February 1996.

[29] L. A. Williams III and B. A. Wooley, "MIDAS - a functional simulator for mixed digital and analog sampled data systems," *IEEE Proceedings of International Symposium on Circuits and Systems (ISCAS)*, pp. 2148-2151, 1992.

[30] R. J. Baker, *CMOS Mixed-Signal Circuit Design*, IEEE Press, 2002.

[31] Q. Li and F. Yuan, "Time domain response and sensitivity of periodically switched nonlinear circuits," *IEEE Transactions on Circuits and Systems I - Fundamental Theory and Applications*, Accepted for publication in April 2003.

[32] Q. Li and F. Yuan, "Time domain response and sensitivity of periodically switched nonlinear circuits," *IEEE Proceedings of International Symposium on Circuits and Systems (ISCAS)*, Vol. 4, pp. 696-699, Bangkok, Thailand, May 2003.

[33] Q. Li and F. Yuan, "Sampled-data simulation of periodically switched nonlinear circuits," *2003 Canadian Conference on Electrical and Computer Engineering (CCECE2003)*, Vol. 1, pp. 219-222, Montréal, Canada, May 2003.

[34] Q. Li and F. Yuan, "Efficient sampled-data simulation of switched nonlinear circuits and nonlinear sigma-delta modulators," presented in *18th Symposium on Microelectronics Research and Development in Canada*, Ottawa, Ontario, Canada, June 2002.