AC analysis of Linear Periodic Time-Varying Circuits Using Numerical Integration

Makiko Okumura[1] Jun Shirataki[1] Minoru Nakamura[2]

[1] Department of Electrical Electronic Engineering, Kanagawa Institute of Technology
[2] Fujitsu Limited, Defense Systems Unit, IT System Division

Abstract

This paper describes a frequency analysis method for linear periodic time-varying (LPTV) circuits. In this method, a linear time-varying equation is approximated as a set of differential equivalentequations using a numerical integration. We propose an efficient method for computing the discretized time-varying transfer functions in terms of many frequency points. Furthermore, computational cost and required memories are reduced by transforming the number of variables for modified nodal analysis into the number of capacitors and inductors in the circuit. We estimate effectiveness of the proposed method by applying it to some examples of SCFs.

Key Words: Analog circuit simulation, Linear periodic time-varying circuit, Switching circuit, Periodic time-varying transfer function

1 Introduction

Standard circuit simulators, such as SPICE, cannot compute frequency responses for circuits whose parameters change with time, for example, switching circuits. Some analysis tools and some analysis methods have been proposed for Switched Capacitor Filters (SCFs) [1, 2, 3]. However, these handle only ideal SCFs which include ideal switches, voltage sources, and capacitors. As our proposed method can deal with all linear elements, the effects of on-resistance for switches can be computed easily. An AC analysis method for nonlinear periodic circuits was presented [4], However, there were still problems that there was high computational cost and implementation on standard circuit simulator was difficult. Then, the methods to treat as a macromodel for the reduction of computational cost were proposed [5][6]. As for these methods, there was a problem in accuracy. In this paper, the method of reducing computational cost is proposed with accuracy maintained. The method is easy for implementation into a circuit simulator. In this method, a linear time-varying equation is approximated as a set of differential equivalent equations using a numerical integration. The discretized time-varying transfer functions are computed by applying a complex sinusoidal input into the set of the differential equivalent circuits. We propose an efficient method for computing the transfer functions for many frequency points. Modified Nodal Analysis (MNA) is usually used for formulation in circuit simulators. The variables in MNA are all nodal voltages and branch currents for inductors and voltage sources. In our method, the variables in MNA are transformed to the state vari-

ables using a matrix representing the interconnections of capacitors and inductors of the network. The number of variables is reduced dramatically.

In the second section, we describe the analysis method for linear periodic time-varying (LPTV) circuits using numerical integrations.

In the third section, we present the proposed efficient computational method and variables transformation method.

In the fourth section, we compare the exact solution with our two numerical solutions: One computed by the backward Euler method and the other by the trapezoidal rule for a simple RC circuit.

In the next section, we show the influence of on-resistances for switches in band-pass filter.

Finally, we show the availability of the proposed method by comparing it with conventional method in terms of computational cost for third-order SCF.

2 AC analysis for LPTV circuit using numerical integration

2.1 LPTV transfer function

We describe the LPTV transfer function [4] for circuits whose parameters periodically change with time. An output $\boldsymbol{x}(t)$ of the LPTV circuit with an input $u(t)$ is written as

$$\boldsymbol{x}(t) = \boldsymbol{H}(\omega, t)u(t), \tag{1}$$

where $\boldsymbol{H}(\omega, t)$ is a time-varying transfer function. Then, $\boldsymbol{H}(\omega, t)$ is a T-periodic function;

$$\boldsymbol{H}(\omega, t+T) = \boldsymbol{H}(\omega, t).$$

Fourier series expression is given by

$$\boldsymbol{H}(\omega, t) = \sum_{\ell=-\infty}^{\infty} \boldsymbol{H}_\ell(\omega) e^{j\ell\omega_s t}, \tag{2}$$

where $\boldsymbol{H}_\ell(\omega)$ denotes a ℓth Fourier coefficient and $\omega_s = \frac{2\pi}{T}$. And then,

$$\boldsymbol{H}_\ell(\omega) = \frac{1}{T}\int_0^T \boldsymbol{H}(\omega, t)e^{-j\ell\omega_s t}dt. \tag{3}$$

Now, we consider a complex sinusoidal wave $u(t) = e^{j\omega t}$ as an input in (1). Substituting (2) into (1), we get the next equation,

$$\boldsymbol{x}(t) = \sum_{\ell=-\infty}^{\infty} \boldsymbol{H}_\ell(\omega) e^{j(\omega+\ell\omega_s)t}. \tag{4}$$

From (4), we can interpret that Fourier coefficient $\boldsymbol{H}_\ell(\omega)$ is the transfer function for modulated output that appears at the frequency which moves from input frequency ω to frequency $\omega \pm \ell\omega_s$. These transfer functions $\boldsymbol{H}_\ell(\omega)$ are shown in Fig.1. From (3, $\boldsymbol{H}_\ell(\omega)$ is approximated by

$$\boldsymbol{H}_\ell \approx \frac{j}{2\pi\ell}\sum_{m=1}^{P} \boldsymbol{H}(\omega, \tau_m)e^{j\ell\omega_s h}, \qquad \ell = 1, 2, 3, \cdots$$

$$\boldsymbol{H}_0(\omega) \approx \frac{1}{T}\sum_{m=1}^{P} \boldsymbol{H}(\omega, \tau_m)h \tag{5}$$

where $\boldsymbol{H}(\omega, \tau_m)$ are solutions at sampled time, $\tau_m = mh$ $(m = 1, 2, \cdots, P)$, $T = Ph$, P is a number of sampling points for one period, and h is a time step for the sampling.

2.2 Calculation for transfer functions using numerical integration

We consider how to compute $\boldsymbol{H}(\omega, \tau_m)$ numerically. A differential equation for a LPTV circuit is written as

$$\boldsymbol{a}(t)\boldsymbol{x}(t) + \boldsymbol{b}(t)\dot{\boldsymbol{x}}(t) = \boldsymbol{u}(t), \tag{6}$$

where $\boldsymbol{a}(t)$ and $\boldsymbol{b}(t)$ are parameters of the periodic time-varying circuit and their period is T. Then, applying complex sinusoidal input $\boldsymbol{u}(t) = \boldsymbol{u}e^{j\omega t}$ and substituting $t = nT + \tau_m$ into (6) gives

$$\boldsymbol{a}_m\boldsymbol{x}(nT+\tau_m) + \boldsymbol{b}_m\dot{\boldsymbol{x}}(nT+\tau_m) = \boldsymbol{u}e^{j\omega(nT+\tau_m)} \qquad (m = 1, 2, \cdots, P), \tag{7}$$

where we can replace $\boldsymbol{a}(nT+\tau_m)$ with $\boldsymbol{a}_m$ as $\boldsymbol{a}(t)$ is periodic. Next, we approximate the differential term $\dot{\boldsymbol{x}}(nT+\tau_m)$ in (8) by using a numerical integration method. For example, applying backward Euler discretization to $\dot{\boldsymbol{x}}(nT+\tau_m)$ gives

$$\dot{\boldsymbol{x}}(nT+\tau_m) = \frac{\boldsymbol{x}(nT+\tau_m) - \boldsymbol{x}(nT+\tau_{m-1})}{h}. \tag{8}$$

where $h = \tau_{m+1} - \tau_m$. Substituting $t = nT + \tau_m$ to (1) gives

$$\boldsymbol{x}(nT+\tau_m) = \boldsymbol{H}(\omega, \tau_m)\boldsymbol{u}e^{j\omega(nT+\tau_m)} \qquad (m = 1, 2, \cdots, P). \tag{9}$$

where $\boldsymbol{H}(\omega,\tau_m)=\boldsymbol{H}(\omega,nT+\tau_m)$ as $\boldsymbol{H}(\omega,t)$ is periodic. Substituting (8) and (10) into (8), we get

$$(\boldsymbol{a}_m+\tfrac{\boldsymbol{b}_m}{h})\hat{\boldsymbol{H}}(\omega,\tau_m)\boldsymbol{u}-\tfrac{\boldsymbol{b}_m}{h}\hat{\boldsymbol{H}}(\omega,\tau_{m-1})\boldsymbol{u}e^{-j\omega h}=\boldsymbol{u} \quad (m=1,2,\cdots,P), \tag{10}$$

where $\hat{\boldsymbol{H}}$ denotes an approximation of $\boldsymbol{H}$ for numerical integration. We rewrite the above equation (10) as a matrix,

$$\begin{bmatrix} \boldsymbol{J}_1 & & & \boldsymbol{C}_1 \\ \boldsymbol{C}_2 & \boldsymbol{J}_2 & & \\ & \ddots & \ddots & \\ & & \boldsymbol{C}_P & \boldsymbol{J}_P \end{bmatrix}\begin{bmatrix} \hat{\boldsymbol{X}}_1 \\ \hat{\boldsymbol{X}}_2 \\ \vdots \\ \hat{\boldsymbol{X}}_P \end{bmatrix}=\begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{u} \\ \vdots \\ \boldsymbol{u} \end{bmatrix}, \tag{11}$$

where

$$\hat{\boldsymbol{X}}_m\equiv\hat{\boldsymbol{X}}(\omega,\tau_m)\equiv\hat{\boldsymbol{H}}(\omega,\tau_m)\boldsymbol{u}$$
$$\boldsymbol{J}_m\equiv\boldsymbol{a}_m+\tfrac{\boldsymbol{b}_m}{h}\quad \boldsymbol{C}_m\equiv-\tfrac{\boldsymbol{b}_m}{h}e^{-j\omega h}.$$

Therefore, $\boldsymbol{H}_0(\omega)$ can be calculated by using (5) and (11). In the case of applying the trapezoidal rule to (8), $\boldsymbol{b}_m\dot{\boldsymbol{x}}(nT+\tau_m)$ is approximated by

$$\boldsymbol{b}_m\dot{\boldsymbol{x}}(nT+\tau_m)=\tfrac{2}{h}(\boldsymbol{b}_m\boldsymbol{x}(nT+\tau_m)-\boldsymbol{b}_{m-1}\boldsymbol{x}(nT+\tau_{m-1}))-\boldsymbol{b}_{m-1}\dot{\boldsymbol{x}}(nT+\tau_{m-1}). \tag{12}$$

From (8), $\boldsymbol{b}_{m-1}\dot{\boldsymbol{x}}(nT+\tau_{m-1})$ is given by

$$\boldsymbol{b}_{m-1}\dot{\boldsymbol{x}}(nT+\tau_{m-1})=\boldsymbol{u}_{m-1}-\boldsymbol{a}_{m-1}\boldsymbol{x}(nT+\tau_{m-1}),$$

and we get

$$(\boldsymbol{a}_m+\tfrac{2\boldsymbol{b}_m}{h})\hat{\boldsymbol{H}}(\omega,\tau_m)\boldsymbol{u}+(\boldsymbol{a}_{m-1}-\tfrac{2\boldsymbol{b}_{m-1}}{h})\cdot\hat{\boldsymbol{H}}(\omega,\tau_{m-1})\boldsymbol{u}e^{-j\omega h}=\boldsymbol{u}(1+e^{-j\omega h}). \tag{13}$$

Equation (13) has the form of a difference equation between $\hat{\boldsymbol{H}}(\omega,\tau_m)$ and $\hat{\boldsymbol{H}}(\omega,\tau_{m-1})$. Then, if we write (13) by using matrix, its non-zero blocks are placed the same as in (11). Therefore, the computational cost is the same as in the case of using backward Euler or a trapezoidal algorithm.

3 Efficient calculation methods

3.1 Efficient calculation method for many frequencies

Since equation (11) is computed repeatedly for many frequency points in the above mentioned analysis method, computational cost is a problem. In our method, we transform equation (11), dividing it into two parts; frequency dependent part and frequency independent part. Independent terms of frequency are computed beforehand and are stored. Only the terms which depend on frequency are computed repeatedly. In the following, we present the proposed method.

Firstly, each diagonal block is transformed into a unit matrix by multiplying the next matrix from the left side in (11).

$$\begin{bmatrix} \boldsymbol{J}_1^{-1} & & & & & \\ & \boldsymbol{J}_2^{-1} & & & & \\ & & \boldsymbol{J}_3^{-1} & & & \\ & & & \ddots & & \\ & & & & \boldsymbol{J}_{P-1}^{-1} & \\ & & & & & \boldsymbol{J}_P^{-1} \end{bmatrix} \tag{14}$$

Then, we get

$$\begin{bmatrix} \boldsymbol{I} & & & & \boldsymbol{D}_1 \\ \boldsymbol{D}_2 & \boldsymbol{I} & & & \\ & \ddots & \ddots & & \\ & & \boldsymbol{D}_{P-1} & \boldsymbol{I} & \\ & & & \boldsymbol{D}_P & \boldsymbol{I} \end{bmatrix}\begin{bmatrix} \hat{\boldsymbol{X}}_1 \\ \hat{\boldsymbol{X}}_2 \\ \vdots \\ \hat{\boldsymbol{X}}_{P-1} \\ \hat{\boldsymbol{X}}_P \end{bmatrix}=\begin{bmatrix} \boldsymbol{U}_1 \\ \boldsymbol{U}_2 \\ \vdots \\ \boldsymbol{U}_{P-1} \\ \boldsymbol{U}_P \end{bmatrix}, \tag{15}$$

where $\boldsymbol{I}$ is the unit matrix,

$$\boldsymbol{D}_m=\boldsymbol{J}_m^{-1}\boldsymbol{C}_m\equiv-(\boldsymbol{a}_m+\tfrac{\boldsymbol{b}_m}{h})^{-1}\tfrac{\boldsymbol{b}_m}{h}e^{-j\omega h}\equiv-\bar{\boldsymbol{D}}_m e^{-j\omega h},$$
$$\boldsymbol{U}_m=\boldsymbol{J}_m^{-1}\boldsymbol{u}\quad (m=1,2,\cdots,P).$$

Secondly, eliminate the Pth block in the first row in (15). Adding D_1 times Pth row to the first row, we have

$$\begin{bmatrix} \boldsymbol{I} & & & -\boldsymbol{D}_1\boldsymbol{D}_P \\ \boldsymbol{D}_2 & \boldsymbol{I} & & \\ & \ddots & \ddots & \\ & & \boldsymbol{D}_{P-1} & \boldsymbol{I} \\ & & & \boldsymbol{D}_P & \boldsymbol{I} \end{bmatrix}\begin{bmatrix} \hat{\boldsymbol{X}}_1 \\ \hat{\boldsymbol{X}}_2 \\ \vdots \\ \hat{\boldsymbol{X}}_{P-1} \\ \hat{\boldsymbol{X}}_P \end{bmatrix}=\begin{bmatrix} \boldsymbol{U}_1' \\ \boldsymbol{U}_2 \\ \vdots \\ \boldsymbol{U}_{P-1} \\ \boldsymbol{U}_P \end{bmatrix}.$$

where $\boldsymbol{U}_1'=\boldsymbol{U}_1+e^{j\omega h}\bar{\boldsymbol{D}}_1\boldsymbol{U}_P$.

Repeating the above process for other rows, we get

$$\begin{bmatrix} \boldsymbol{I}-\boldsymbol{Q} & & & \\ \boldsymbol{D}_2 & \boldsymbol{I} & & \\ & \ddots & \ddots & \\ & & \boldsymbol{D}_{P-1} & \boldsymbol{I} \\ & & & \boldsymbol{D}_P & \boldsymbol{I} \end{bmatrix}\begin{bmatrix} \hat{\boldsymbol{X}}_1 \\ \hat{\boldsymbol{X}}_2 \\ \vdots \\ \hat{\boldsymbol{X}}_{P-1} \\ \hat{\boldsymbol{X}}_P \end{bmatrix}=\begin{bmatrix} \boldsymbol{Z} \\ \boldsymbol{U}_2 \\ \vdots \\ \boldsymbol{U}_{P-1} \\ \boldsymbol{U}_P \end{bmatrix}, \tag{16}$$

where

$$\begin{aligned}\boldsymbol{Q} &= e^{-j\omega T}\bar{\boldsymbol{D}}_1\bar{\boldsymbol{D}}_P\bar{\boldsymbol{D}}_{P-1}\cdots\bar{\boldsymbol{D}}_3\bar{\boldsymbol{D}}_2,\\ \boldsymbol{Z} &= \boldsymbol{U}_1 - \boldsymbol{D}_1\boldsymbol{U}_P + \boldsymbol{D}_1\boldsymbol{D}_P\boldsymbol{U}_{P-1} - \boldsymbol{D}_1\boldsymbol{D}_P\boldsymbol{D}_{P-1}\boldsymbol{U}_{P-2}\\ &\quad +\cdots+\boldsymbol{D}_1\boldsymbol{D}_P\boldsymbol{D}_{P-1}\boldsymbol{D}_{P-2}\cdots\boldsymbol{D}_3\boldsymbol{U}_2\\ &= \boldsymbol{U}_1 + e^{-j\omega h}\bar{\boldsymbol{D}}_1\boldsymbol{U}_P + e^{-j\omega(2h)}\bar{\boldsymbol{D}}_1\bar{\boldsymbol{D}}_P\boldsymbol{U}_{P-1}\\ &\quad +\cdots+e^{-j\omega(P-1)h}\bar{\boldsymbol{D}}_1\bar{\boldsymbol{D}}_P\bar{\boldsymbol{D}}_{P-1}\bar{\boldsymbol{D}}_{P-2}\cdots\bar{\boldsymbol{D}}_3\boldsymbol{U}_2.\end{aligned}$$

In (16), LU decomposition is required only fir the first row. Other rows can be calculated by the next substitution.

$$\hat{\boldsymbol{X}}_m = \boldsymbol{U}_m - \boldsymbol{D}_m\hat{\boldsymbol{X}}_{m-1}\ (m=2,3,\cdots,P),$$

For the calculation of $\boldsymbol{Q}$ matrix and $\boldsymbol{Z}$ vector in (17), independent terms of frequency, that is terms except $e^{-j\omega h}, e^{-j\omega 2h},\cdots,e^{-j\omega T}$, are calculated beforehand and are stored. Then, computational cost for repeated calculation with many frequencies is reduced.

3.2 Contraction of circuit size

Modified nodal analysis is often used in simulation. Variables of MNA are node voltages and branch currents for inductors and voltage sources. They increase as the circuit size becomes larger. In this method, the amount of memory required and computational cost increase in proportion to product of circuit size and the number of time steps in the numerical integration for one period. So, for reducing computational cost and required memories, we propose a method to decrease variables in the circuit. The number of variables is reduced to the total number of capacitors and inductors in the circuit. In our proposed method, define the next equation,

$$\begin{cases}\hat{\boldsymbol{Y}}_m = \boldsymbol{A}^t\hat{\boldsymbol{X}}_m\\ \boldsymbol{b}_m = \boldsymbol{A}\hat{\boldsymbol{b}}_m\boldsymbol{A}^t,\end{cases}\qquad(17)$$

where $\hat{\boldsymbol{X}}_m$ is a vector of variables in MNA and $\hat{\boldsymbol{X}}_m \in \boldsymbol{C}^N$. $\hat{\boldsymbol{Y}}_m$ is a vector of reduced variables to voltages across capacitors and branch currents of inductors and $\hat{\boldsymbol{Y}}_m \in \boldsymbol{C}^{N_s}$. and $\boldsymbol{A} \in \boldsymbol{R}^{N\times N_s}$ is a matrix representing connecting information of capacitors and inductors, whose element is zero or ±1, and $\boldsymbol{b}_m \in \boldsymbol{R}^{N\times N}$ is the same matrix as $\boldsymbol{b}_m$ in (13) and $\hat{\boldsymbol{b}}_m \in \boldsymbol{R}^{N_s\times N_s}$ is a diagonal matrix whose elements are values of capacitors and inductors. In most case, $N > N_s$. For example, we consider a circuit shown in Fig.2.

The number of variables in MNA, N, is 5, and the number of variables, N_s, is 1. As the capacitor is connected from node 2 to node 3, the transposition matrix of $\boldsymbol{A}$ is

$$\boldsymbol{A}^t = \begin{bmatrix}0 & 1 & -1 & 0 & 0\end{bmatrix}.$$

Rewriting (15),

$$\hat{\boldsymbol{X}}_m = -\boldsymbol{J}^{-1}\frac{\boldsymbol{b}_m}{h}e^{-j\omega h}\hat{\boldsymbol{X}}_m + \boldsymbol{J}_m^{-1}\boldsymbol{u}\qquad(18)$$
$$(m=1,2,\cdots,P).$$

Multiplying $\boldsymbol{A}^t$ from the left side of (19) and substituting (17),

$$\hat{\boldsymbol{Y}}_m = -e^{-j\omega h}\hat{\boldsymbol{D}}_m\hat{\boldsymbol{Y}}_{m-1} + \hat{\boldsymbol{U}}_m$$
$$(m=1,2,\cdots,P),$$

where

$$\begin{cases}\hat{\boldsymbol{D}}_m = \boldsymbol{A}^t\boldsymbol{J}_m^{-1}\boldsymbol{A}\frac{\hat{\boldsymbol{b}}_m}{h}\\ \hat{\boldsymbol{U}}_m = \boldsymbol{A}^t\boldsymbol{J}_m^{-1}\boldsymbol{u}\end{cases}\quad(\hat{\boldsymbol{D}}_m \in \boldsymbol{R}^{N_s\times N_s}, \hat{\boldsymbol{U}}_m \in \boldsymbol{R}^{N_s}).$$

The matrix of (15) is reduced from $(N\times P)\times(N\times P)$ to $(N_s\times P)\times(N_s\times P)$. Therefore, calculation cost and the amount of memory required are reduced.

4 Simulation for example networks

4.1 Comparison between simulation and exact solution

Consider an example circuit shown in Fig.3. $R_1//R_2$ is 10[Ω] when switch S is on, and R_1 is 10[kΩ] when switch S is off. Capacitance C is 1[μF] and period T is 1[μsec].

Calculation for exact solution We calculate analytically an exact solution of $\boldsymbol{H}_0(\omega)$ for the circuit shown in Fig.3. The result is given by

$$\boldsymbol{H}_0(\omega) = \frac{1}{T}\left\{\frac{A(\omega)Z_1(\omega)+\frac{\alpha T}{2}}{j\omega+\alpha} + \frac{B(\omega)Z_2(\omega)+\frac{\beta T}{2}}{j\omega+\beta}\right\},\qquad(19)$$

where

$$\begin{aligned}&\alpha \equiv \tfrac{1}{R_1C}, \qquad \beta \equiv \tfrac{1}{R_2C},\\ &A(\omega) = 1-e^{-(j\omega+\alpha)\frac{T}{2}}, \quad B(\omega) = 1-e^{-(j\omega+\beta)\frac{T}{2}},\\ &Z_1(\omega) \equiv \frac{1}{1-e^{-\left\{j\omega+\frac{1}{2}(\alpha+\beta)\right\}T}}\left[\frac{\alpha\cdot a e^{-(j\omega+\beta)\frac{T}{2}}}{j\omega+\alpha}+\frac{\beta\cdot b}{j\omega+\beta}\right]-\frac{\alpha}{j\omega+\alpha},\\ &Z_2(\omega) \equiv \frac{1}{1-e^{-\left\{j\omega+\frac{1}{2}(\alpha+\beta)\right\}T}}\left[\frac{\beta\cdot b e^{-(j\omega+\alpha)\frac{T}{2}}}{j\omega+\beta}+\frac{\alpha\cdot a}{j\omega+\alpha}\right]-\frac{\beta}{j\omega+\beta}.\end{aligned}$$

Comparison between numerical solution and exact solution We compare the exact solution with two solutions: one computed by using backward Euler and the other by the trapezoidal rule. And we define the error function ε in

$$\varepsilon = \left| \frac{M-K}{K} \right| \times 100 \quad [\%],$$

where K is an exact solution and M is a numerical solution. The results for the amplitude characteristic of $H_0(\omega)$ are shown in Fig.5, where $P = 50$ and $h = 20$[nsec] for simulation.

The figure shows the error function of $H_0(\omega)$ depends on the error of numerical integrations.

4.2 Examination of influence of on-resistance

We examine influence of on-resistance, resistance on a switch when switch is on, using the band-pass SCF as shown in Fig.6. The results obtained with the simulators that we developed and meaured values are shown in Fig.7.

4.3 Estimation of computational cost

We estimate computational cost of third-order SCF shown in Fig.8. And the simulation result is shown in Fig.9.

Table. 1 shows the comparison between computational cost of the equation (15) using LU decomposition and that of equation (16) using our proposed method. p and q indicate the number of time steps and the number of frequency points, respectively. It is found that

表 1 Estimation of computational cost using proposed method of 3.1

Method	Computational cost	3rd's SCF $N = 61$, $p = 50$, $q = 100$
Directly	$q \times (N \times p)^3$	2.8×10^{12}
Proposed method 3.1	$(p+q) \times N^3$	3.4×10^6

our proposed method is useful from the view point of reducing computational cost. Table. 2 shows computational cost when variables are reduced by using (17) in the circuit of Fig.8. It decreased to one-third.

表 2 Estimation of methods 3.1 and 3.2

Methods	Computational cost	3rd's SCF $N = 61$, $N_s = 12$, $p = 50$, $q = 100$
3.1 only	$(p+q) \times N^3$	3.4×10^6
3.1+3.2	$p \times N^3 + q \times N_s^3$	1.2×10^6

5 Conclusion

This paper described a frequency analysis method for LPTV circuits. A time varying derivative equation is approximated as a set of differential equivalent equations by using numerical integration in this method. We proposed an efficient method for computing the discretized time varying transfer functions in terms of a lot of frequency points. Furthermore, computational cost and required memories are reduced by transforming the number of variables for modified nodal analysis into the number of capacitors and inductors in the circuit. We showed the availability of our method by comparing the proposed method with conventional method of computational cost for third-order SCF.

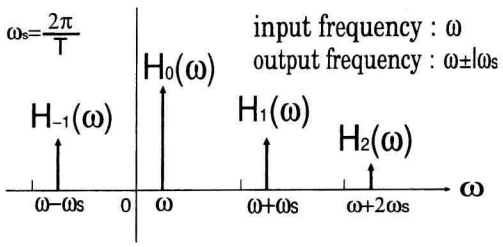


図 1 Spectrum of $\boldsymbol{H}_\ell(\omega)$

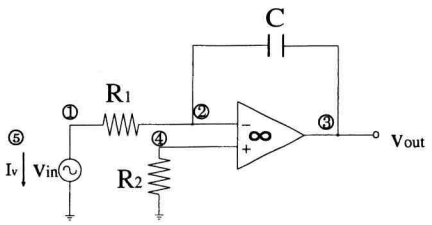


図 2 An example circuit

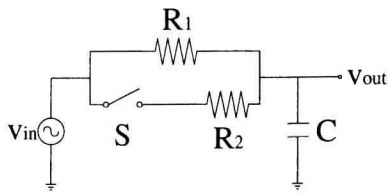


図 3 RC time-varing Circuit concluding a switch

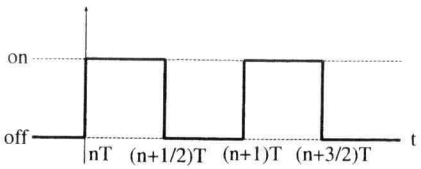


図 4 Waveform of a clock for switch S

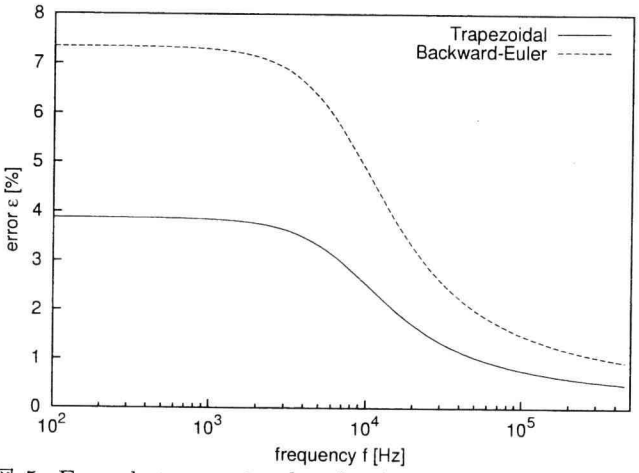


図 5 Error between simulated solution and exact solution

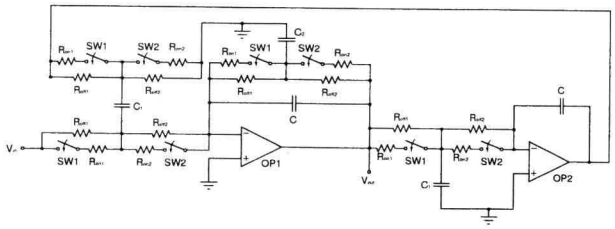


図 6 Band-pass switched-capacitor filter

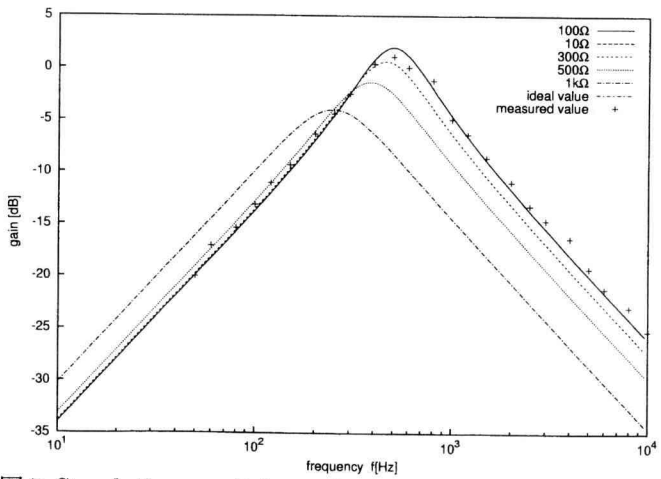
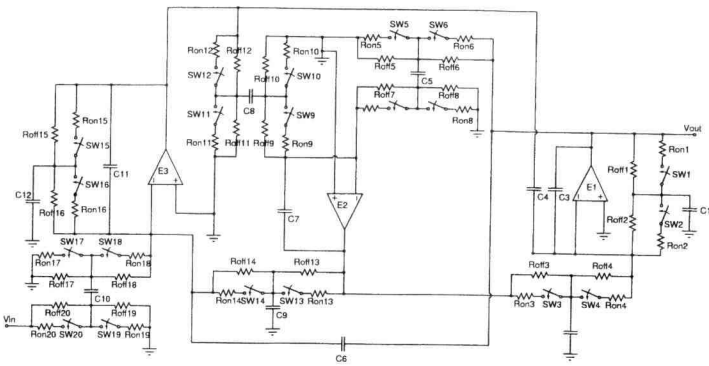


図 7 Simulation result for various on-resistances in band-pass SCF



$C_1 = 196.06p, C_2 = 196.06p, C_3 = 2.206n, C_4 = 694.02p,$
$C_5 = 468.35p, C_6 = 2.206n, C_7 = 2.200n, C_8 = 468.35p,$
$C_9 = 197.48p, C_{10} = 394.97p, C_{11} = 2.222n, C_{12} = 699.05p$

図 8 Third-order SCF circuit

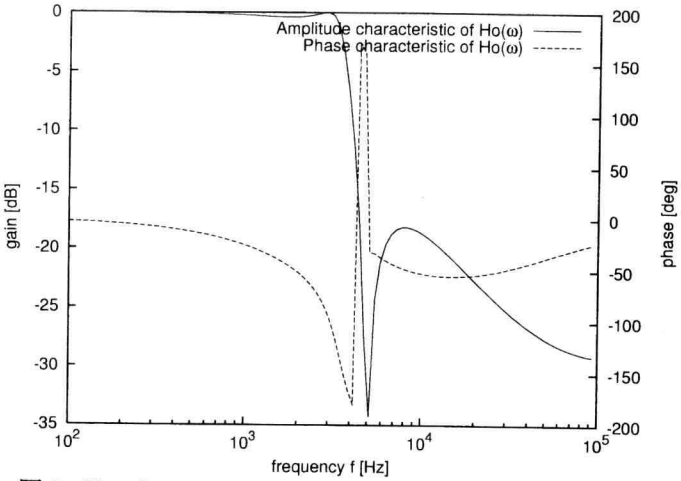


図 9 Simulation results for third-order SCF circuit

参考文献

[1] S. C. Fang, Y. P. Tsividis, O. Wing, "SWITCAP:A Switched-Capacitor Network Analysis Program Part I:Basic Features," IEEE Circuits and Syst. Mag. Vol.5, No.3, pp.4–10, Sept.1983.

[2] M. Vlach, J. Vlach, K. Singhal, R. Chadha, E. Christen, "WATSCAD - A program for the analysis and design of switched capacitor networks," IEEE Int. Symp. Circuits Syst. Vol.1985, No.Vol.3, pp.1177–1178, 1985.

[3] Yuh Sun,"Direct analysis of time-varying continuous and discrete difference equations with application to nonuniformly switched-capacitor circuits," IEEE Trans. CAS, Vol.CAs-28, No.2, pp.93-100, Feb. 1981.

[4] M. Okumura, T. Sugawara and H. Tanimoto, "An efficient small signal analysis method of nonlinear circuits with two frequency excitations," *IEEE Trans. Computer-Aided Des. Circuits & Syst.*, Vol.9, No.3, pp.225–235, Mar. 1990.

[5] Jaijeet Roychowdhury, "Reduced-order modeling of linear time-varying systems,"*Proc.* of the 1998 IEE/ACM, pp.92–95, 1998.

[6] Xin Li, P. Li, Y. Xu, L. T. Pileggi,"Analog and RF Circuit Macromodel for System-Level Analysis," *DAC*, p.478, 2003.