IEICE TRANS. FUNDAMENTALS, VOL.E91–A, NO.9 SEPTEMBER 2008

PAPER Special Section on Nonlinear Theory and its Applications

Sensitivity Analysis and Optimization Algorithm —— Based on Nonlinear Programming ——

Masayoshi ODA[†a)], ***Nonmember***, **Yoshihiro YAMAGAMI**[†b)], **Junji KAWATA**[††c)], **Yoshifumi NISHIO**[†d)], ***Members***, ***and*** **Akio USHIDA**[††e)], ***Fellow***

SUMMARY We propose here a fully Spice-oriented design algorithm of op-amps for attaining the maximum gains under low power consumptions and assigned slew-rates. Our optimization algorithm is based on a well-known steepest descent method combining with nonlinear programming. The algorithm is realized by equivalent RC circuits with ABMs (analog behavior models) of Spice. The gradient direction is decided by the analysis of sensitivity circuits. The optimum parameters can be found at the equilibrium point in the transient response of the RC circuit. Although the optimization time is much faster than the other design tools, the results might be rough because of the simple transistor models. If much better parameter values are required, they can be improved with Spice simulator and/or other tools.

key words: *sensitivity analysis, steepest descent method, optimization algorithm, spice-oriented*

1. Introduction

Amplifiers are widely used as the building blocks of many analog circuits [1]–[4]. In this paper, we propose a fully Spice-oriented parameter optimization algorithm for attaining the maximum gains of operational amplifiers. There are many kinds of designing items such as the maximum gain, low power consumption, slew-rate, phase margin, GBW (gain band width) and so on. The circuits are usually designed by choosing the suitable resistors, bias voltages and/or sizes (lengths, widths) of MOSFETs as the optimization parameters. Traditionally, the parameters have been found by trial and error methods with Spice simulations and by designer's experimental knowledge. However, it is really time-consuming task for the circuits containing a large number of parameters. Thus, there have been proposed some designing tools, recently. ASCO (A Spice Circuit Optimizer) [5] is based on the differential evolution algorithm, where it takes about 15 [hours] for designing a 3-stage CMOS op-amp. On the other hand, FSCODE [6] is based on both the differential evolution [7] and GA, where it takes 1,255 [secs] for designing a 2-stage CMOS op-amp. Although their methods can precisely design the circuits satisfying many design items, they are still time-consuming, especially for circuits having many optimization parameters. Because they use both the transient and AC analysis of Spice to decide the optimization direction at each step of the evolution algorithms. Hence, we propose here a simple fully Spice-oriented steepest descent optimization algorithm combining with nonlinear programming [8], [9], which can rapidly find out the rough solutions*. The rough solutions are mainly due to the simple models of bipolar transistors and MOSFETs such as Ebers-Moll and Shichman-Hodges models neglecting the parasitic elements. Therefore, if we want better solutions, they should be improved by the applications of Spice simulator and/or other tools with the accurate models.

Firstly, we define the *objective function* to be minimized as follow;

$$\Phi(\mathbf{x}, \mathbf{p}), \quad \mathbf{x} \in R^n, \quad \mathbf{p} \in R^k, \tag{1}$$

where

Φ: consists of the maximum gain, low power consumption, slew-rate and so on.

$\mathbf{x}$: circuit variables such as voltages and currents.

$\mathbf{p}$: optimization parameters such as bias voltages, resistor's values and the dimensions of MOSFETs such as W's [width, μm] and L's [length, μm], and so on.

The gradient direction for attaining the maximum gain is given by

$$\frac{dp_i}{ds} = -\frac{dS_v(\mathbf{p})}{dp_i}, \quad i = 1, 2, \ldots, k, \tag{2a}$$

where $S_v(\mathbf{p})$ has minus sign to the gain estimated by the sensitivity circuit, and p_i is an optimization parameter. Unfortunately, it is impossible to evaluate $\frac{dS_v(\mathbf{p})}{dp_i}$ from the sensitivity circuit. Therefore, we introduce here a *numerical differentiation technique* as follows;

$$\frac{dp_i}{ds} = -\frac{S_{v,i}(\mathbf{p} + \Delta\mathbf{p}) - S_v(\mathbf{p})}{\Delta p_i}, \quad i = 1, 2, \ldots, k,$$
$$\Delta\mathbf{p} = (0, 0, \ldots, \Delta p_i, \ldots, 0)^T \tag{2b}$$

with a sufficiently small variational parameters $\Delta\mathbf{p}$. The

Manuscript received November 26, 2007.
Manuscript revised March 12, 2008.

[†]The authors are with the Department of Electrical and Electronic Engineering, Faculty of Engineering, Tokushima University, Tokushima-shi, 770-8506 Japan.
[††]The authors are with the Department of Mechanical and Electronic Engineering, Faculty of Engineering, Tokushima Bunri University, Sanuki-shi, 769-2193 Japan.
a) E-mail: oda@ee.tokushima-u.ac.jp
b) E-mail: yamagami@ee.tokushima-u.ac.jp
c) E-mail: kawata@fe.bunri-u.ac.jp
d) E-mail: nishio@ee.tokushima-u.ac.jp
e) E-mail: ushida@fe.bunri-u.ac.jp
DOI: 10.1093/ietfec/e91–a.9.2426

*For comparisons between Ref. [9] and our method, see Appendix A.

relation (2b) means that "s" is an auxiliary variable, such that p_i component of the gradient vector to $-S_v(\mathbf{p})$ is equal to dp_i/ds. Replacing the auxiliary variable "$s = kt$" with time "t" in (2b), our steepest descent algorithm can be realized by the equivalent nonlinear RC circuits using ABMs of Spice [12]†. The circuits are driven by the controlled-current sources obtained from the sensitivity circuits. The optimum parameters can be found at the equilibrium point of the transient analysis. Moreover, for attaining both the conditions of low power consumption and assigned slew-rate, we need to modify the algorithm (2b) as shown in Sect. 2.2. The sensitivity modules of MOSFETs and bipolar transistors are given in Sect. 3. Our algorithm can be only depend on the analysis of DC sensitivity circuits. Therefore, the capacitive elements must be replaced by equivalent resistive elements as shown in Sect. 4. We show two illustrative examples in Sect. 5. First example is a two-stage CMOS amplifier for attaining maximum gain under low power consumption, where the size (widths and lengths) of MOSs and a reference current are chosen as optimization parameters. Second is the design of an operational amplifier for attaining the DC maximum gain, where the resistors are chosen as the optimization parameters. The solutions of two examples are found quickly, which are equal to the solutions from transient analysis.

2. Sensitivity Analysis and Spice-Oriented Optimization Algorithm

2.1 Sensitivity Analysis

The steepest descent method is the most basic optimization approach [8], [11], where the gradient direction is decided by the solutions of sensitivity circuit. Let us derive the sensitivity circuit via tableau approach [10]. The *tableau equation* is given by;

$$\begin{bmatrix} \mathbf{K}_i & \mathbf{K}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & -\mathbf{A}^T \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{i} \\ \mathbf{v} \\ \mathbf{v}_n \end{bmatrix} - \begin{bmatrix} \mathbf{g}(\mathbf{v}, \mathbf{i}) \\ \mathbf{E} \\ \mathbf{AJ} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \quad (3)$$

The first row corresponds to the Ohm's law, where matrices $\mathbf{K}_i, \mathbf{K}_v$ consist of 1 and/or 0 elements, the second and third rows are respectively Kirchhoff's voltage and current laws, where $\mathbf{A}$ is an incidence matrix. $\mathbf{E}$ and $\mathbf{J}$ show the voltage and current source vectors. Now, let us define the *sensitivity* as follow;

Sensitivity of x_k to a parameter p_i:

$$S_{k,i} \equiv \lim_{\Delta p_i \to 0} \frac{\Delta x_k}{\Delta p_i}. \quad (4)$$

Differentiating (3) by an optimization parameter p_i, we can derive the sensitivity tableau equation as follows;

$$\begin{bmatrix} \mathbf{K}_i & \mathbf{K}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & -\mathbf{A}^T \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{S}_{i,p_i} \\ \mathbf{S}_{v,p_i} \\ \mathbf{S}_{v_n,p_i} \end{bmatrix} - \begin{bmatrix} \left.\frac{\partial \mathbf{g}(\mathbf{v}, \mathbf{i})}{\partial \mathbf{v}}\right|_{\mathbf{v}_0, \mathbf{i}_0} \mathbf{S}_{v,p_i} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

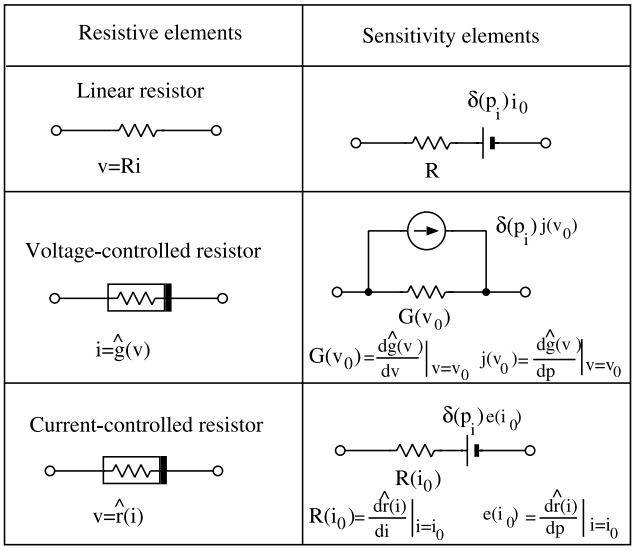

Fig. 1 Sensitivity resistors, where $\delta(p_i)$ denotes that, if the resistor is chosen as an optimization parameter, $\delta(p_i) = 1$. Otherwise, $\delta(p_i) = 0$.

$$-\begin{bmatrix} \left.\frac{\partial \mathbf{g}(\mathbf{v}, \mathbf{i})}{\partial \mathbf{i}}\right|_{\mathbf{v}_0, \mathbf{i}_0} \mathbf{S}_{i,p_i} \\ \delta(i) \\ \mathbf{A}\delta(i) \end{bmatrix} = \begin{bmatrix} \left.\frac{\partial \mathbf{g}(\mathbf{v}, \mathbf{i})}{\partial \mathbf{p}}\right|_{\mathbf{v}_0, \mathbf{i}_0} \delta(i) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad (5)$$

where $\delta(i)$ means a delta function satisfying

$$\delta(i) = [0, \ 0, \ \ldots, \ \underset{i}{1}, \ \ldots, \ 0]^T \ , \quad (6)$$

where $\mathbf{S}_{i,p_i}$ means the sensitivity of $\mathbf{i}$ for p_i and so on. Thus, we can develop the sensitivity circuit for p_i parameter as the netlist or schematic diagram. Observe that the structure of sensitivity circuit is equal to the original circuit except that the nonlinear elements are replaced by the linear incremental resistors at the operating points $\{\mathbf{v}_0, \mathbf{i}_0\}$ and the modified sources.

Now, we summarize the algorithm for deriving the sensitivity circuit.

1. When a voltage source is chosen as an optimization parameter, the source is set to "1 [V]." The other voltage sources are removed by the shorted-circuits.
2. When a current source is chosen as an optimization parameter, the source is set to "1 [A]." The other current sources are removed by the opened-circuits.
3. When resistor is chosen as an optimum parameter, it is replaced by the linear incremental resistor coupled with a controlled source $\left.\frac{\partial \mathbf{g}(\mathbf{v},\mathbf{i})}{\partial \mathbf{p}}\right|_{\mathbf{v}_0,\mathbf{i}_0} \delta(i)$. Other resistors are only replaced by the incremental resistors. These transformation algorithms to resistors are shown in Fig. 1.

Example 1: Now, consider a simple example of nonlinear circuit shown in Fig. 2(a), where the nonlinear resistor is given by

†When we choose a larger "k," the convergence ratio will become larger.

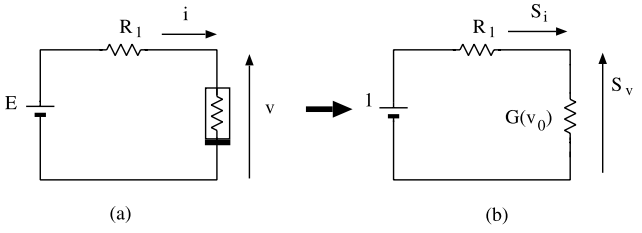

Fig. 2 (a) Simple nonlinear resistive circuit, (b) the sensitivity circuit to optimization parameter E.

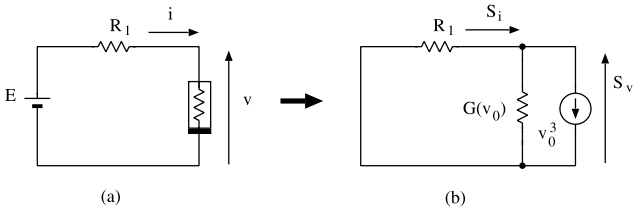

Fig. 3 (a) Simple nonlinear resistive circuit, (b) the sensitivity circuit for the optimization parameter c_3.

$$i = c_1 v + c_3 v^3. \tag{7}$$

Let us calculate the sensitivity $S_{v,E} = dv/dE$. The sensitivity circuit as shown in Fig. 2(b) can be obtained by the above transformation shown in Fig. 1. The source voltage is replaced by a unit voltage, and the nonlinear resistor is replaced by an incremental conductance

$$G(v_0) = c_1 + 3c_3 v_0^2 \tag{8}$$

at the operating voltage v_0 in Fig. 2(a).
The sensitivity is given by

$$S_{v,E} = \frac{1/G(v_0)}{R_1 + 1/G(v_0)} = \frac{1}{1 + R_1 G(v_0)}. \tag{9}$$

This relation is equal to the result obtained by the differentiation dv/dE from Fig. 2(a).

Example 2: Now, consider the same circuit as Fig. 2(a), but c_3 in (7) is chosen as an optimization parameter. In this case, the nonlinear resistor in Fig. 3(a) is transformed into the sensitivity element according to the third row in Fig. 1, and we have Fig. 3(b). Thus, the sensitivity $S_{v,c_3} = dv/dc_3$ is given by

$$S_{v,c_3} = \left.\frac{dv}{dc_3}\right|_{v=v_0} = -\frac{v_0^3}{G(v_0) + \frac{1}{R_1}}. \tag{10}$$

Remark that the voltage source E is removed by the shorted-circuit.

2.2 Spice-Oriented Optimization Technique

Now, consider our optimization algorithm for attaining the maximum gain under low power consumption and assigned slew-rate. In this case, the steepest descent algorithm for attaining the maximum gain given by (2b) should be modified

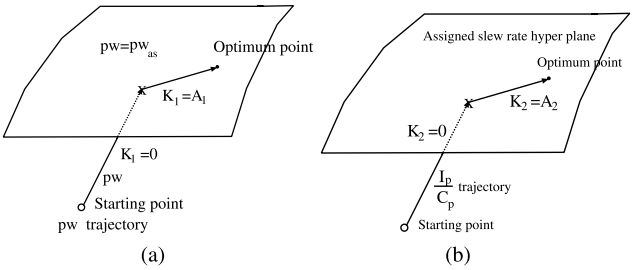

Fig. 4 Schematic diagrams of trajectory.

as follows;

$$\frac{dp_i}{ds} = K_1(\mathrm{pw}_{as} - \mathrm{pw}(\mathbf{p} + \Delta\mathbf{p})) + K_2(\mathrm{slw}_{as} - \mathrm{slw}(\mathbf{p} + \Delta\mathbf{p})) - \frac{S_{v,i}(\mathbf{p} + \Delta\mathbf{p}) - S_v(\mathbf{p})}{\Delta p_i}. \tag{11}$$

$$K_1 = \begin{cases} 0 & : \ \mathrm{pw}_{as} > \mathrm{pw} \\ A_1 & : \ \mathrm{pw}_{as} \le \mathrm{pw} \ \text{ for sufficiently large } A_1 \end{cases}$$

$$K_2 = \begin{cases} 0 & : \ \mathrm{slw}_{as} > \mathrm{slw} \\ A_2 & : \ \mathrm{slw}_{as} \le \mathrm{slw} \ \text{ for sufficiently large } A_2 \end{cases}$$

$$\Delta\mathbf{p} = (0, 0, \ldots, \Delta p_i, \ldots, 0)^T{}^{\dagger},$$

where symbols pw_{as} denotes an assigned low power consumption, and slw_{as} an assigned large slew-rate. The first and second terms in the right hand side are added to (2b). The terms restrict the low power consumption to pw_{as} and assigned slew-rate to slw_{as}. The variational power and slew-rate are given by $\mathrm{pw}(\mathbf{p} + \Delta\mathbf{p})$, and $\mathrm{slw}(\mathbf{p} + \Delta\mathbf{p})$, respectively. The trajectory in this algorithm moves along the steepest descent direction until the curve has reached to the hyper-plane satisfying $\mathrm{pw} = \mathrm{pw}_{as}$ and/or $\mathrm{slw} = \mathrm{slw}_{as}$. After then, it moves on the plane because the variables $-\mathrm{pw}(\mathbf{p} + \Delta\mathbf{p})$ and $-\mathrm{slw}(\mathbf{p} + \Delta\mathbf{p})$ in (11) are multiplied by sufficiently large constants A_1, and A_2. Thus, the trajectory still moves to the optimum point on the hyper-plane.

These schematic diagrams are shown in Figs. 4(a) and (b). Thus, we can find out the optimum point satisfying the maximum gain under the given low power consumption and assigned slew-rate, where the slew-rate is defined [1] by

$$\text{slew-rate} = \mathrm{Max}\left|\frac{dv_{out}}{dt}\right|. \tag{12}$$

The slew-rate usually corresponds to the derivative of v_{out} at $t = 0$, which is given by the functions of $I_k/C_k, k = 1, 2, \ldots, n$ for the capacitors[††]. Replacing the variable "s" by time "t" in (11), the algorithm is realized by RC circuit as shown in Fig. 5. where $C = 1$, and R_0 is a sufficiently large dummy resistor to avoid the C-J cut-set. The controlled

[†] Δp_i should be a small number. In our examples, we have chosen around 0.1% of p_i. If it becomes larger, the steepest descent direction may be erroneous.

[††] Assume that a capacitor C_p gives the large effect to slew-rate (12). Then, the second term in (11) is approximately replaced by $K_2(\mathrm{slw}_{as} - I_p(\mathbf{p} + \Delta\mathbf{p}/C_p)$, for an assigned slew-rate slw_{as}. If it does not satisfy the assigned slew-rate, we need to replace the capacitor to the other.

current sources at the left hand side in Fig. 5 are obtained from the sensitivity and the variational circuits with $p_i + \Delta p_i$. Thus, the optimization parameters satisfying the above conditions can be found at the equilibrium point of the transient analysis of (11) using Spice. Note that although C's in Fig. 5 are set equal to "1" because of (11), the convergence ratio will be largely increased by choosing the smaller values.
Our steepest descent algorithm based on nonlinear programming is shown by the flowchart of Fig. 6.

Algorithm

1. Firstly, set the upper and lower boundaries of optimization parameters $p_i, i = 1, 2, \ldots, k$. The lower boundary of power consumption pw_{as} and higher boundary of slew-rate slw_{as} should be also set.
2. The frequency is set at the highest value in the GBW. The capacitive elements are replaced by the equivalent resistors whose the transformation technique is given in the Sect. 4.
3. Formulate "k+1" sensitivity circuits corresponding to (11)†. Choose sufficiently large constants A_1 and A_2.
4. Set the initial parameter values of $p_{i0}, i = 1, 2, \ldots, k$ within the boundaries.
5. Execute the transient analysis of Spice with $K_1 = K_2 = 0$. After arriving at one of the hyper-planes shown in Fig. 4(a) or (b), K_1 or K_2 is set to a sufficiently large constant A_1 or A_2.
6. The equilibrium point gives the optimization parameters satisfying the power consumption and slew-rate.
7. The phase margin is calculated by AC analysis of Spice. If the above optimization parameters do not satisfy the phase margin, we need to change the boundaries $p_i, i = 1, 2, \ldots, k$, and go to 4. Otherwise stop.

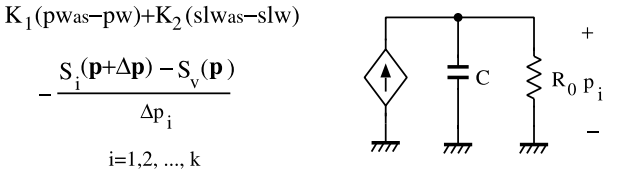

Fig. 5 A circuit configuration for attaining the maximum gain under low power consumption and the slew-rate.

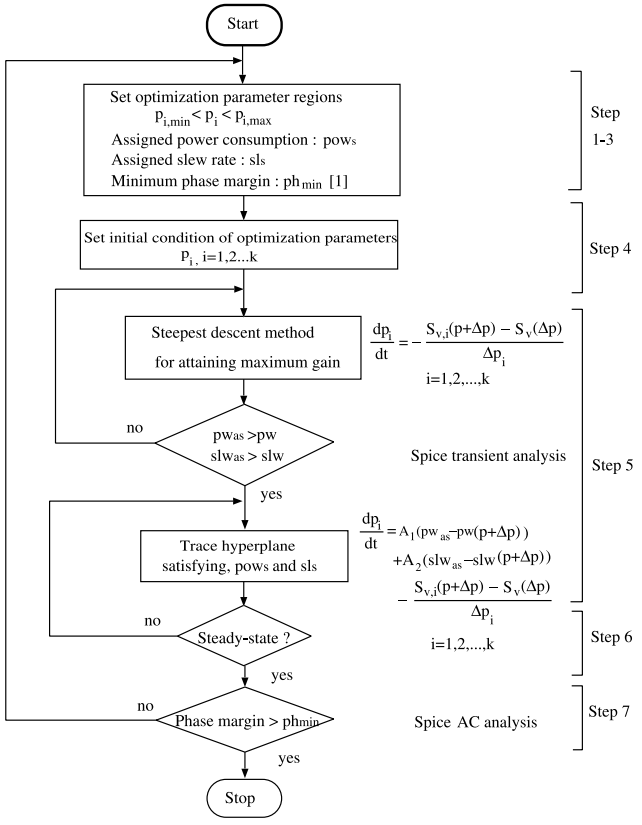

Fig. 6 Flowchart of our steepest descent algorithm based on the nonlinear programming.

3. Sensitivity Modules of MOSFETs and Bipolar Transistors

Now, let us develop the sensitivity modules of MOSFETs and bipolar transistors. Once these modules are stored in our computer library as the packages, we can easily formulate the sensitivity circuits as the netlist and/or schematic diagram††.

3.1 Sensitivity Modules of MOSFETs

We consider here a simple Shichman-Hodges model [1]–[4] of nMOS shown in Fig. 7(a);

1. **Linear region** ($v_{GS} > v_T,\ 0 < v_{DS} < v_{GS} - v_T$)

$$i_D = \frac{k_n W}{L}\left[(v_{GS} - v_T) - \frac{v_{DS}}{2}\right] v_{DS}(1 + \lambda_n v_{DS}) \quad (13a)$$

2. **Saturation region** ($v_{GS} > v_T,\ v_{DS} \geq v_{GS} - v_T$)

$$i_D = \frac{k_n W}{2L}(v_{GS} - v_T)^2(1 + \lambda_n v_{DS}) \quad (13b)$$

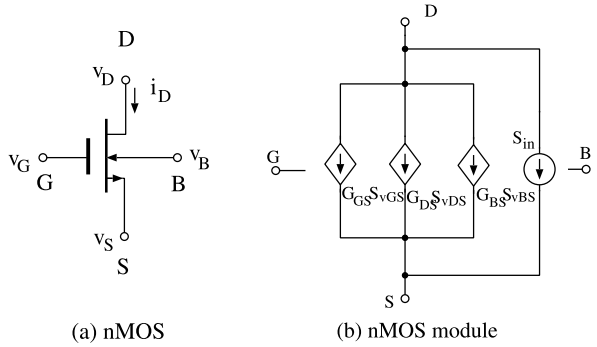

Fig. 7 (a) nMOS, (b) nMOS sensitivity module.

†Since our algorithm uses numerical differentiation to "k" variables, we need "k+1" sensitivity circuits to estimate the current sources in Fig. 5.
††See Appendix B.

Where the threshold voltage is given by

$$v_T = v_{T0} + \gamma\left(\sqrt{\phi - v_{BS}} - \sqrt{\phi}\right), \tag{13c}$$

and "W" and "L" are the width and length of nMOS. For simplicity, we rewrite (13) in the following form;

$$i_D = \hat{i}_D(v_{GS}, v_{DS}, v_{BS}, W, L) \tag{14}$$

Then, we have the following variational equation;

$$\begin{aligned}\Delta i_D = &\frac{\partial \hat{i}_D}{\partial v_{GS}}\Delta v_{GS} + \frac{\partial \hat{i}_D}{\partial v_{DS}}\Delta v_{DS} + \frac{\partial \hat{i}_D}{\partial v_{BS}}\Delta v_{BS} \\ &+ \frac{\partial \hat{i}_D}{\partial W}\Delta W + \frac{\partial \hat{i}_D}{\partial L}\Delta L.\end{aligned} \tag{15}$$

Therefore, the sensitivity of drain current i_D is given for each optimization parameter p_i

$$\begin{aligned}S_{i_D,pi} = &\frac{\partial \hat{i}_D}{\partial v_{GS}} S_{v_{pi,GS}} + \frac{\partial \hat{i}_D}{\partial v_{DS}} S_{v_{pi,DS}} + \frac{\partial \hat{i}_D}{\partial v_{BS}} S_{v_{pi,BS}} \\ &+ \frac{\partial \hat{i}_D}{\partial W}\delta(p_i, W) + \frac{\partial \hat{i}_D}{\partial L}\delta(p_i, L),\end{aligned} \tag{16}$$

where δ means

$$\delta(x_1, x_2) = \begin{cases} 1 & \text{for } x_1 = x_2 \\ 0 & \text{for } x_1 \neq x_2 \end{cases}$$

Thus, we have MOSFET module shown in Fig. 7(b), where

$$G_{GS} = \frac{\partial \hat{i}_D}{\partial v_{GS}},\ G_{DS} = \frac{\partial \hat{i}_D}{\partial v_{DS}},\ G_{BS} = \frac{\partial \hat{i}_D}{\partial v_{BS}}$$

$$S_{in} = \frac{\partial \hat{i}_D}{\partial W}\delta(p_i, W) + \frac{\partial \hat{i}_D}{\partial L}\delta(p_i, L).$$

where, for simplicity, we set the sensitivity voltages as follows;

$$S_{VGS} = S_{v_{p_i},GS},\ \ S_{VDS} = S_{v_{p_i},DS},\ \ S_{VBS} = S_{v_{p_i},BS}.$$

The sensitivity module of pMOS is obtained in the same manner as nMOS. These modules can be easily developed with ABM of Spice [12].

3.2 Sensitivity Modules of Bipolar Transistors

Bipolar transistors are usually modeled with Ebers-Moll model or Gummel-Poon model [1]–[4]. The collector current i_C and base current i_B are functions of v_{BE} and v_{BC} as follows;

$$\begin{aligned} i_C &= \hat{i}_C(v_{BE}, v_{BC}) \\ i_B &= \hat{i}_B(v_{BE}, v_{BC}) \end{aligned}. \tag{17}$$

Therefore, the sensitivities for optimization parameters p_i are given by

$$\begin{aligned} S_{i_C,p_i} &= \frac{\partial \hat{i}_C}{\partial v_{BE}}\frac{\partial v_{BE}}{\partial p_i} + \frac{\partial \hat{i}_C}{\partial v_{BC}}\frac{\partial v_{BC}}{\partial p_i} \\ &\equiv G_{i_C,v_{BE}} S_{v_{pi,BE}} + G_{i_C,v_{BC}} S_{v_{pi,BC}} \\ S_{i_B,p_i} &= \frac{\partial \hat{i}_B}{\partial v_{BE}}\frac{\partial v_{BE}}{\partial p_i} + \frac{\partial \hat{i}_B}{\partial v_{BC}}\frac{\partial v_{BC}}{\partial p_i} \\ &\equiv G_{i_B,v_{BE}} S_{v_{pi,BE}} + G_{i_B,v_{BC}} S_{v_{pi,BC}} \end{aligned}. \tag{18}$$

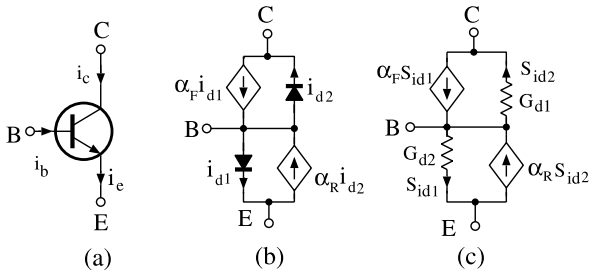

Fig. 8 (a) NPN bipolar transistor, (b) Ebers-Moll model, (c) sensitivity module of transistor.

Thus, the sensitivity module of NPN bipolar transistor modeled by Ebers-Moll model† is given by Fig. 8(c)††, where

$$G_{d1} = \frac{\partial i_{d1}}{\partial v_{BE}},\ \ G_{d2} = \frac{\partial i_{d2}}{\partial v_{BC}},$$

and the sensitivity currents $S_{id1},\ S_{id2}$ are set by

$$S_{id1} = \frac{\partial i_{d1}}{\partial v_{BE}},\ \ S_{id2} = \frac{\partial i_{d2}}{\partial v_{BE}}.$$

In the same manner, we can develop PNP bipolar sensitivity module.

4. Resistive Models of Capacitors

Our Spice-oriented steepest descent method can be executed using the circuit shown in Fig. 5, where the controlled-current sources in the left hand side are decided by the sensitivity circuits. The circuits consist of all the resistive elements, that means our optimization method does not permit to contain any dynamical elements such as capacitors and inductors. On the other hand, many kinds of amplifiers [4] have some capacitors, so that we need to transform them into the approximate equivalent resistive elements as follows;

$$Z_C = \left|\frac{1}{j\omega C}\right| = \frac{1}{\omega C}. \tag{19}$$

Note that it is a function of frequency ω.

Example 3: Consider a simple circuit containing a series and a parallel RC circuits as shown in Fig. 9(a), where C_1 and C_2 correspond to the coupling and bypass capacitors in the case of amplifiers. The frequency response curves using the resistive models are almost exact in the mid-frequency region except for lower and higher cut-off frequency regions. Though the model (19) might affect the accuracy of our optimization technique, we have to use the model since the steepest decent circuit of Fig. 5 is simulated by DC analysis of Spice.

†Ebers-Moll model is given by Eq. (22) in Sect. 5.

††Remark that, if the optimization parameters are different from the transistor parameters, the sensitivity module corresponds to the linear incremental model as shown in Fig. 8(c).

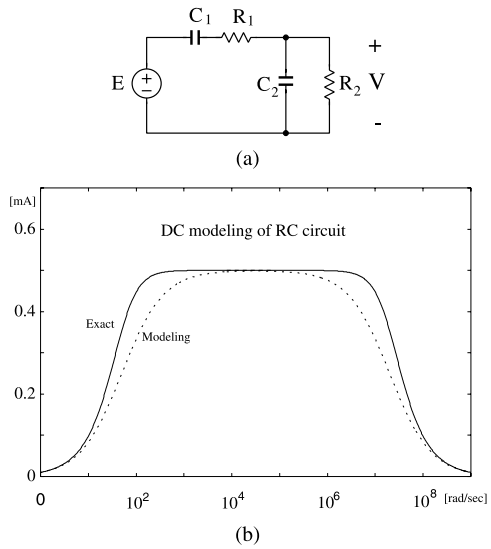

Fig. 9 (a) A simple RC circuit. (b) Frequency response curves. $R_1 = R_2 = 10\,[\mathrm{k}\Omega], C_1 = 1\,[\mu\mathrm{F}], C_2 = 10\,[\mathrm{pF}]$.

5. Illustrative Examples

5.1 CMOS Two-Stage Amplifier

Consider a two-stage CMOS amplifier as shown in Fig. 10(a). Let us design the circuit for attaining the maximum gain under low power consumption and a larger slew-rate. We have modeled the MOSFETs using Schichman-Hodges models and their sensitivity modules, where the circuit parameters are given as follows [4]:

$K_p = 19.344\,[\mu\mathrm{A}]$, $pV_{T0} = -0.8311\,[\mathrm{V}]$, $\lambda_p = 0.1$
$pW9 = 10\,[\mu\mathrm{m}]$, $pW8 = 50\,[\mu\mathrm{m}]$, $pW1 = 50\,[\mu\mathrm{m}]$,
$K_n = 74.209\,[\mu\mathrm{A}]$ $nV_{T0} = 0.6081\,[\mathrm{V}]$, $\lambda_n = 0.2$
$nW3 = 1\,[\mu\mathrm{m}]$ $nW6 = 3\,[\mu\mathrm{m}]$,
$pL = nL = 0.1\,[\mu\mathrm{m}]$ $J = 90\,[\mu\mathrm{A}]$

Note that for these parameter values, the gain was -1.956 in Ref. [4].

Next, we design the circuit under the objective function;

$$\phi = \{\mathrm{Max}|gain| : I_{in} < I_s,\ I_p > I_{psl}\} \tag{20}$$

The initial conditions are set as follows;

$$pW9_0 = 2\,[\mu\mathrm{m}],\ pW8_0 = pW1_0 = nW_0 = 5\,[\mu\mathrm{m}]$$
$$nW3_0 = 10\,[\mu\mathrm{m}],\ J_0 = 5\,[\mu\mathrm{A}]$$

and $C = 0.01\,[\mathrm{F}]$, $R_0 = 1000\,[\mathrm{k}\Omega]$ of the steepest descent method in Fig. 5, where we have restricted the regions of 6 optimization parameters as follows;

$$0.5\,[\mu\mathrm{m}] < pW9, pW8, pW1, nW3, nW6 < 60\,[\mu\mathrm{m}]$$
$$10\,[\mu\mathrm{A}] < I_{in} < 200\,[\mu\mathrm{A}]$$

Then, we have obtained the results shown in Table 1. Where

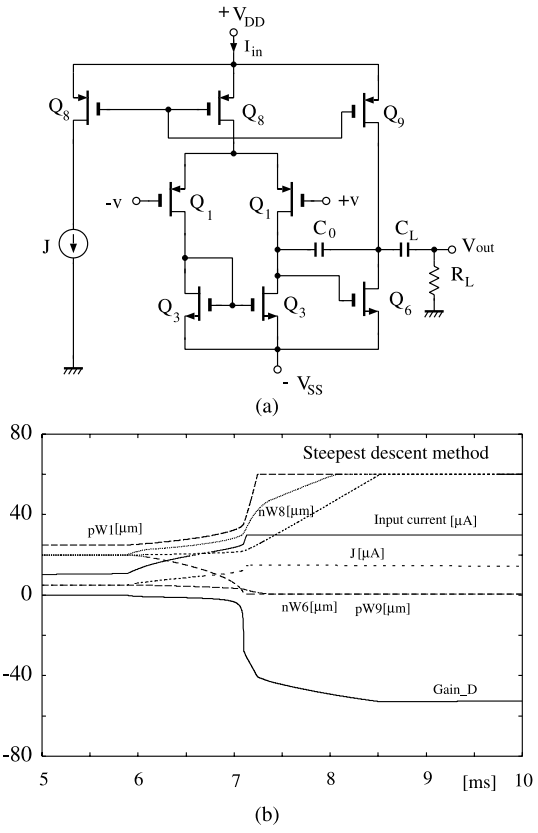

Fig. 10 (a) Two stage CMOS amplifier. (b) Optimization of two-stage CMOS amplifier. $V_{DD} = 2.5\,[\mathrm{V}]$, $V_{SS} = -2.5\,[\mathrm{V}]$, Frequency =100 [MHz], $C_0 = 0.1\,[\mathrm{nF}], C_L = 1\,[\mathrm{nF}], R_L = 1000\,[\mathrm{k}\Omega]$.

Table 1 Simulation results.

No	A_1	I_S (μA)	I_{in} (μA)	J (μA)	$Gain_D$	$Gain_T$	$pW8$ (μm)
1	-	-	100.1	50.3	−57.6	−57.4	60
2	10^4	50	54.6	26.7	−54.1	−54.0	7.2
	10^5	50	50.1	24.5	−56.8	−56.5	24.2
	10^6	50	49.6	25.0	−57.7	−57.4	60.0
3	10^6	40	39.7	19.4	−55.8	−55.3	60.0
	10^6	30	29.8	14.5	−52.7	−51.9	60.0
	10^6	20	20.0	9.4	−45.6	−43.9	26.7

A_1 is a sufficiently large constant given in (11), and the computational time was around 225–230 [sec]†. In this case, the power consumption is given by

$$\mathrm{pw}_{as} = I_S(V_{DD} - V_{SS}) \simeq 100\,[\mu\mathrm{W}], \tag{21}$$

where I_S is current from the source V_{DD}. In the table, I_{in}, $Gain_D$ are the input current and gain obtained by our optimization. We can see that I_{in} is nearly equal to I_S. $Gain_T$ is the gain obtained by the transient analysis of Spice, where the parameters are set to those from our optimization.

We found interesting results as follows;

†The slew-rates were always larger than 1.5 [V/μsec] [6], so that we did not choose them in the objective function. All the results in this paper were simulated by a PC with the following spec; CPU: PentiumM 733 (1.10 GHz), Main memory: 256 MB, OS: Windows XP professional SP2.

1. No.1 is the result from our optimization for attaining the maximum gain without the restrictions for power consumption and slew-rate. We have obtained the following MOS sizes;

   $pW8 = pW1 = nW3 = 60\,[\mu\text{m}]$,
   $pW9 = nW6 = 0.5\,[\mu\text{m}]$,

   where we have fixed $pL = nL = 0.1\,[\mu\text{m}]$ for all the MOSFETs. The input current was $I_{in} = 100\,[\mu\text{A}]$, and the maximum gain was −59.6.
2. No.2 is the results for attaining the maximum gain under the low power consumption ($I_S = 50\,[\mu\text{A}]$) without the load coupled by capacitor (C_L, R_L), where we set A_1 from 10^4 to 10^6 in (11). For $A_1 = 10^4$, we had $pW8 = 7.21\,[\mu\text{m}]$ and $I_{in} = 54.6\,[\mu\text{A}]$ which is different from the assigned current $I_S = 50\,[\mu\text{A}]$. From these results, we can conclude that the constant A_1 should be chosen larger than 10^5.
3. No.3 is the results for the different assigned currents $I_S = 40\,[\mu\text{A}], 30\,[\mu\text{A}], 20\,[\mu\text{A}]$. Note that the op-amp did not work under $I_S = 10\,[\mu\text{A}]$. In these optimizations, the capacitor C_0 is set to

   $$Z_0 = \left|\frac{1}{j\omega C_0}\right| = 15.9\,[\Omega].$$

4. We found the slew-rate becomes smaller according to the input current. However, we had enough slew-rates larger than $4.2\,[\text{V}/\mu\text{sec}]$ [6] in all the cases.
   The transient behavior of our steepest descent method is shown in Fig. 10(b), the optimum values are obtained by the equilibrium point.
5. Note that all the optimization parameter values are obtained on the boundaries because of the nonlinear programming.

5.2 Bipolar Operational Amplifier

Next, consider an operational amplifier [4] shown in Fig. 11(a). We apply our optimization algorithm to design the circuit for attaining the maximum gain, where $R_1 - R_5$ in Fig. 11(a) are chosen as the optimization parameters.

Note that, for parameters given in Ref. [4]:

$R_1 = 10\,[\text{k}\Omega],\ R_2 = 3\,[\text{k}\Omega], R_1 = 2.3\,[\text{k}\Omega], R_4 = 28\,[\text{k}\Omega],$
$R_5 = 15.7\,[\text{k}\Omega], R_6 = 3\,[\text{k}\Omega],$

the gain was 7,765, where the transistors are modeled by Ebers-Moll model

$$i_C = I_s\left(\alpha_F \exp\left(\frac{v_{BE}}{V_T}\right) - \exp\left(\frac{v_{BC}}{V_T}\right)\right), \tag{22a}$$

$$i_E = I_s\left(\exp\left(\frac{v_{BE}}{V_T}\right) - \alpha_R \exp\left(\frac{v_{BC}}{V_T}\right)\right), \tag{22b}$$

where $I_s = 10^{-12}$ [A], $V_T = 0.026$, $\alpha_F = 0.99$, $\alpha_R = 0.3$.

Firstly, we have set the optimization parameter regions as follows;

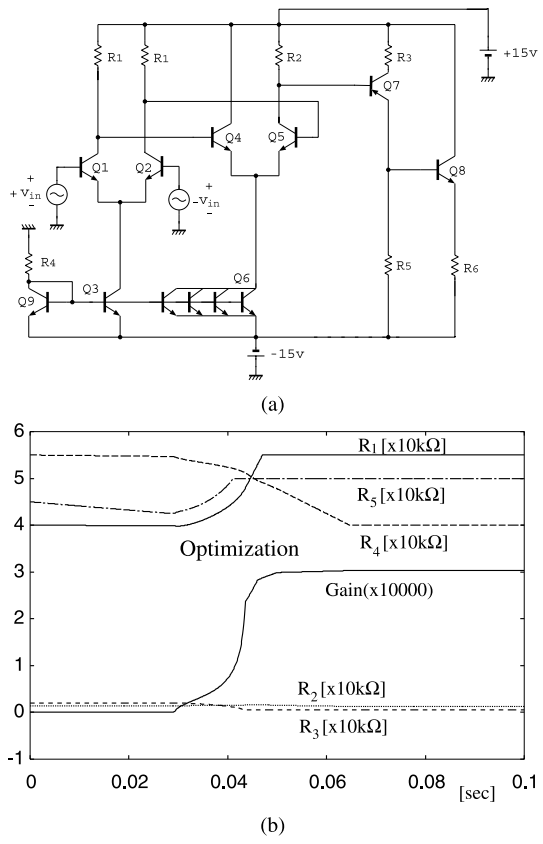

Fig. 11 (a) Bipolar operational amplifier. (b) Optimization result of operational amplifier.

$10\,[\text{k}\Omega] < R_1 < 55\,[\text{k}\Omega],\ 1\,[\text{k}\Omega] < R_2 < 6\,[\text{k}\Omega],$
$0.5\,[\text{k}\Omega] < R_3 < 5\,[\text{k}\Omega],\ 40\,[\text{k}\Omega] < R_4 < 60\,[\text{k}\Omega],$
$10\,[\text{k}\Omega] < R_5 < 50\,[\text{k}\Omega],\ R_6 = 3\,[\text{k}\Omega]$

and applied our optimization algorithm. The optimization result from our steepest descent method is shown in Fig. 11(b). The parameter values are

$R_1 = 55\,[\text{k}\Omega],\ R_2 = 1.0\,[\text{k}\Omega], R_1 = 0.5\,[\text{k}\Omega], R_4 = 40\,[\text{k}\Omega],$
$R_5 = 50\,[\text{k}\Omega], R_6 = 3\,[\text{k}\Omega],$

and the maximum gain is equal to 30,514. The computational time was 36.95 [sec].

Thus, our steepest descent method based nonlinear programming can quickly find out the optimum parameters.

6. Conclusions and Remarks

In this paper, we have proposed a fully Spice-oriented design algorithm of amplifies for attaining the maximum gain, low power consumption and assigned slew-rate. The algorithm is based on the steepest descent method which is realized by the nonlinear RC circuits using ABMs of Spice. We found that our optimization method can find out the solutions much faster than optimizers ASCO (A SPICE Circuit Optimizer) [5] and FSCODE [6]. However, our results might be rough solutions because we are using simple Schichman-Hodges models of MOSFETs, and Ebers-Moll

models of bipolar transistors. If we want more accurate solutions, we need to improve the results by the applications of Spice simulator and the other tools.

Design algorithms considering the other items such as nonlinear distortion, and pulse waveforms are future problems.

References

[1] G. Roubik, Introduction to CMOS OP-AMPs and Comparators, Wiley-Interscience Pub., 1999.
[2] F. Maloberti, Analog Design for CMOS VLSI Systems, KLuwer Academic Pub., 2001.
[3] S.M. Kang and Y. Leblebici, CMOS Digital Integrated Circuits: Analysis and Design, McGraw-Hill, 1996.
[4] A.S. Sedra and K.C. Smith, Microelectronic Circuits, Oxford Univ. Press, 2004.
[5] http://asco.sourceforge.net/
[6] FSCODE: A practical methodology for parameters design an optimization, private communication (unpublished).
[7] P. Whittle, Networks: Optimisation and Evolution, Cambridge Univ. Press, 2007.
[8] J.A. Snyman, Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithm, Springer, 2005.
[9] J. Guo, T. Nishi, and N. Takahashi, "A method for solving optimization problems with equality constraints by using Spice program," IEICE Trans. Fundamentals, vol.E86-A, no.9, pp.2325–2332, Sept. 2003.
[10] L.O. Chua and P.-M. Lin, Computer-Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques, Prentice-Hall, 1975.
[11] J. Vlach and K. Singhal, Computer Methods for Circuit Analysis and Design, Van Nostrand Reinhold Co., 1983.
[12] MicroSim: PSpice A/D circuit Analysis User's Guides, MicroSim Co., 1995.
[13] J.A. Connelly and P. Choi, Macromodeling with Spice, Prentice Hall, 1992.

Appendix A: Comparisons between Ref. [9] and Our Method

Reference [9] proposes a new optimization technique using Spice simulator, where the objective function $f(\mathbf{x})$ is minimized under subject to the equality conditions $g_k(\mathbf{x}) = 0, (k = 1, 2, \ldots, m)$. Then, they apply the Lagrange method as follows;

$$\bar{f}(\mathbf{x}) = f(\mathbf{x}) + \sum_{k=1}^{m} \lambda_k g_k(\mathbf{x}). \quad (\mathrm{A}\cdot 1)$$

The minimization is attained by solving

$$C_j \frac{dx_j}{dt} = -\frac{\partial \bar{f}(\mathbf{x})}{\partial x_j}, \quad (\mathrm{A}\cdot 2)$$

where the right hand term is described in the functional form. Then, the equivalent circuit corresponding to (A· 2) is constructed by Spice in the schematic diagram, and the optimum point is found by the equilibrium point.

On the other hand, our optimization method uses the sensitivity circuits for the calculating the steepest descent direction. It is realized by the combinations of original and the sensitivity circuits. Thus, our method is a fully Spice-oriented algorithm compared to Ref. [9]. For example 1, they correspond to Figs. 2(a) and (b). Since the incremental conductance (8) is decided by the voltage "v_0" from the original circuit, their circuits are coupled by an element with ABM of Spice[†], as shown in Fig. A· 1.

Appendix B: Fully Spice-Oriented Algorithm

In Sect. 3, we have developed the sensitivity devices of MOSFET and bipolar transistor shown in Figs. 7(b) and 8(c). Their incremental elements are estimated at the operating points of the original circuit. Therefore, it is convenient to combine them as shown in Fig. A· 2. The left hand sides correspond the original devices and the right hand sides to the sensitivities, and they are coupled in each others. Using these modules, we can formulate the circuit diagram and the sensitivity circuit in the schematic form with Spice. Thus, the optimization parameter values $p_i, i = 1, 2, \ldots, k$ are obtained by the voltages as shown in Fig. 5. We need to transform them into other parameters corresponding to resistances, widths of MOSs and so on. It can be also done with ABMs of Spice [13]. An example of the transformation of resistance is shown in Fig. A· 3, where p_i is given by voltage. Using the voltage p_i and current I_{Ri}, the input-output relation is given by

$$V_{Ri} = p_i I_{Ri}. \quad (\mathrm{A}\cdot 3)$$

Where p_i in the relation (A· 3) corresponds to the resistance. In this way, we can realize the fully Spice-oriented algorithm for designing optimum circuits.

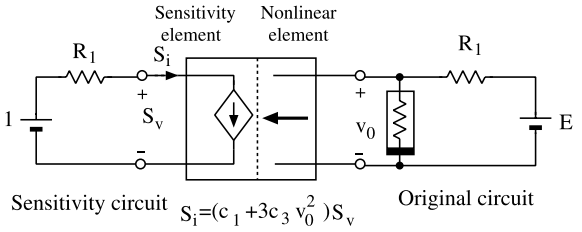

Fig. A· 1 Coupling of the original and the sensitivity circuits.

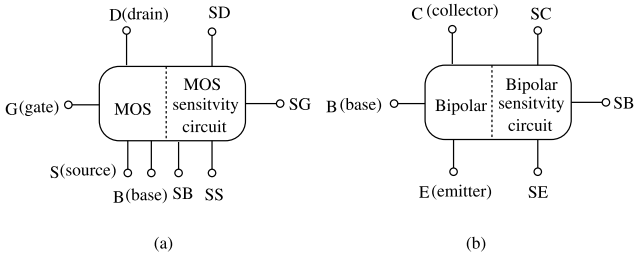

Fig. A· 2 MOSFET and bipolar transistor sensitivity modules.

[†]ABMs have many kind of built-in functions such as Fortran language, and controlled sources VCCS, VCVS, CCCS and CCVS.

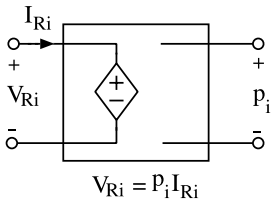

Fig. A· 3 Resistance transformation circuit.

Masayoshi Oda received the B.E. and the M.E. degrees from Electorical and Electronic Engineering from Tokushima University, Tokushima Japan, in 2005 and 2007. His research interests include the computer-aided design technology.

Yoshihiro Yamagami graduated from the Department of Electrical Engineering, Anan Technical College, Tokushima, Japan, in 1982. He joined the Department of Electrical and Electronic Engineering at Tokushima University, Tokushima, Japan, in 1984. He received Ph.D. degree in Electrical and Electronic Engineering from Tokushima University, Tokushima, Japan, in 2006. His research interests include computer-aided simulations of nonlinear circuits and high-speed interconnect circuits.

Junji Kawata received the B.E., M.E., and Ph.D. degrees from Tokushima University, Tokushima, Japan, in 1990, 1992 and 2001, respectively. Since 1992, he has been with the Department of Mechanical and Electronic Engineering at Tokushima Bunri University, where he is currently a Lecturer. His research interests are in circuit simulation and chaotic communication. Dr. Kawata is a member of the IEEE.

Yoshifumi Nishio received B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, Yokohama Japan, in 1988, 1990, and 1993, respectively. In 1993, he joined the Department of Electrical and Electronic Engineering at Tokushima University, Tokushima Japan, where he is currently an Associate Professor. From May 2000 he spent a year in the Laboratory of Nonlinear Systems (LANOS) at the Swiss Federal Institute of Technology Lausanne (EPFL) as a Visiting Professor. His research interests include analysis and application of chaos in electrical circuits, analysis of synchronization in coupled oscillatory circuits, development of analyzing methods for nonlinear circuits, theory and application of cellular neural networks, and neural network architecture. He was the Chair of the IEEE CAS Society Technical Committee on Nonlinear Circuits and Systems (NCAS) during 2004–2005, the Steering Committee Secretary of the IEICE Research Society of Nonlinear Theory and its Applications (NOLTA) during 2004–2007, and is currently the Secretary/Treasurer of the IEEE CAS Society Shikoku Chapter. He was an Associate Editor of the IEEE Transactions on Circuits and Systems–I: Regular Papers during 2004–2005, and is currently serving as an Associate Editor for the IEEE CAS Magazine, the IEEE CAS Society Newsletter, and the RISP Journal of Signal Processing. He is also the Editor of the IEICE Fundamentals Review and is a member of the Editorial Board of International Journal of Circuit Theory and Applications. He is a senior member of the IEEE, and a member of the RISP.

Akio Ushida received the B.E. and M.E. degrees in Electrical Engineering from Tokushima University in 1961 and 1966, respectively, and the Dr. degree in Electrical Engineering from University of Osaka Prefecture in 1974. He was an Associate Professor from 1973 to 1980 at Tokushima University. From 1980 to 2003, he was a Professor in the Department of Electrical and Electronic Engineering at Tokushima university. Since 2003, he has been with the Department of Mechanical and Electronic Engineering, Tokushima Bunri University. From 1974 to 1975 he spent one year as a visiting scholar at the Department of Electrical Engineering and Computer Sciences at the University, Berkeley. His current research interests include numerical methods and computer-aided analysis of nonlinear circuits. Dr. Ushida is a member of the IEEE.