Vol. 05, No. 3 (2023) 425-432, doi: 10.24874/PES05.03.007

Proceedings on Engineering Sciences

www.pesjournal.net

APPROXIMATION OF THE NONLINEAR DEPENDENCIES IN HARMONIC BALANCE EQUATIONS

Vladimir Lantsov[1]

Received 12.01.2023.
Accepted 05.08.2023.
UDC 510.53

Keywords:

Harmonic Balance, Electronics, Model Order Reduction, Singular Values.

A B S T R A C T

A new algorithm is presented to reduce computational costs in solving harmonic balance equations obtained by separating state variables. In the author's previous works, an approach was proposed where the vector (matrix) of unknowns is replaced by two matrices of small dimension, which leads to two systems of balance equations that are solved iteratively. The first equation reduces the number of harmonics in the balance equations, the second equation reduces the number of circuit nodes. In this paper, it is proposed to further reduce computational costs by approximating part of the elements of the balance equations using the decomposition procedure based on singular values. It is proposed to construct a matrix of sets of responses of nonlinear dependencies of circuit models before solving the problem by an iterative method. This matrix reflects all the main changes in nonlinear dependencies with changes in the amplitudes of the input effect and over time. The resulting matrix is then approximated by applying decomposition based on singular values. Comparison of the proposed algorithm with the standard harmonic balance method and algorithms developed by the author earlier showed its high efficiency.

1. INTRODUCTION

Harmonic balance (HB) methods are widely used for modeling nonlinear circuits in CAD systems in electronics (Rizzoli et al., 1988, Gilmore & Steer, 1991, Kundert, 1999). The main problems of algorithms and software tools of CAD systems based on HB methods are significant memory requirements and huge computational costs for simulation of complex nonlinear electronic circuits containing thousands of electronic components and hundreds of thousands for circuit model equations (White & Sangiovanni-Vincentelli, 1987, Kundert et al., 1988, Rizzoli et al., 2011).

For example, for circuits containing about 10 thousand of circuits nodes and taking into account about one thousand harmonics (with multifrequency excitation), the number of variables (unknowns, equations) of the model will be approximately 10 million. If we use for solving the harmonic balance equations the Newton method when the square of this number of unknowns or the size of Jacobian matrix will be 100 million. All this suggests that new more economical methods of storing and solving of harmonic balance equations in modern electronic CAD systems are needed (Nastov et al., 2007, Lantsov, 2020).

[1] Corresponding author: Vladimir Lantsov
Email: lantsov@ieee.org

In this paper, we propose a new method and algorithm for solving and storing harmonic balance equations using the ideas of model order reduction (MOR) algorithms (Bond & Daniel, 2007). MOR methods have gained popularity in recent years (Bond & Daniel, 2009, Lantsov & Dolinina, 2016). This MOR methods allow to significantly reduce the number of unknowns and the size of memory for storing of electronic circuits model equations for dynamic mode analysis. The main problems of MOR methods for modeling electronic circuits are associated with very small reductions in computational costs (while reducing the dimensions of the equations and required memory for the model).

In (Lantsov, 2020a), a new method for solving HB equations was proposed, based on dividing the equations into two components of very small dimension. The reduction of memory and computational costs in the new method was determined by the significantly smaller dimension of the equations and the fact that they were used in solving sequentially. In (Lantsov, 2020a), a simple iteration method was used to solve the equations. Further development of the method was proposed in (Lantsov, 2020b), where relations were obtained and an algorithm for implementation in software tools based on the use of Newton's iterative method was developed. Further development of the method and implementation algorithms was proposed in (Lantsov & Papulina, 2021), where detailed algorithms for implementing of method in CAD software tools for electronics are presented.

The method and algorithms of its implementation in the software for modeling complex electronic circuits have shown high efficiency, especially when solving problems of very high dimension (Lantsov, 2020, Lantsov & Papulina, 2021).

The experience of using previously developed software tools has shown that the main costs of computer memory and simulation time are associated with the storage and processing of the Jacobian of full dimension. In this paper, an algorithm is proposed to significantly reduce computational costs using an effective method of approximation of the Jacobian of the harmonic balance equations when solving by the Newton method.

2. BASIC EQUATIONS OF HARMONIC BALANCE METHOD

The basic relations and algorithms for solving the HB equations for electronic circuits by the new method are given in detail in (Lantsov, 2020b, Lantsov & Papulina, 2021), so here we will give only some basic relations necessary for the presentation of the proposed approach.

Consider the nonlinear circuits which described by the system of nonlinear integra-differential equations

$$f(v(t),t) = i(v(t)) + \frac{dq(v(t))}{dt} + \\ + \int_{-\infty}^{t} y(t-\tau)v(\tau)d\tau + i_E(t) = 0. \qquad (1)$$

Here, $v(t)$ is a vector of node voltages with dimension N; $i(v(t))$ is a vector of currents for resistive elements; $q(v(t))$ is a vector for capacitor charges; $u(t)$ are a input sources; $y(t)$ describes linear part of circuits; i and q describe a nonlinear elements; the size of equations is N (Lantsov, 2020a).

In the HB method, it is assumed that functions v and f are represented as a Fourier series (so, the HB method is referred to as methods in the frequency domain):

$$x(t) = \sum_{k=-\infty}^{\infty} X(k)e^{j\omega_k t},$$

where $\omega_k = k\lambda, \quad \lambda = 2\pi/T$ is the base frequency.

The equations (1) then can be transformed into next form of nonlinear system of equations

$$F(V,k) = I(V,k) + j\omega_k Q(V,k) + \\ + Y(k)V(k) + I_E(k) = 0.$$

In vector-matrix form

$$F(V) = I(V) + YV - I_{\mathrm{E}} = 0, \qquad (2)$$

where V is a vector of unknowns (nodal voltages in the circuit); the first term $I(V)$ describes nonlinear elements; the second term YV characterizes the linear part of the circuit; the element I_{E} is a vector of input sources. The dimension of the equations is equal to [N×(2K+1)], where N is the number of nodes in the circuit, K is the number of harmonics taken into account (Lantsov, 2020a).

The solution of harmonic balance equations (2) in the frequency domain is most often performed by the Newton iterative method

$$J(V^i) \cdot \Delta V^{i+1} = -F(V^i), \qquad (3)$$

where $J(V^i)$ is a Jacobian matrix; $\Delta V^{i+1} = V^{i+1} - V^i$; i is iteration number. The Jacobian matrix can be found as

$$J(V^j) = {\partial F}/{\partial V}\,|_{V^i} = Y + {\partial I}/{\partial V}\,|_{V^i}. \qquad (4)$$

In HB method the dimension of Jacoby matrix ([(2K+1)×N] × [(2K+1)×N]) which for complex malty components circuits becomes too large and needs excessive computer memory an computational costs.

Model order reduction seeks to reduce the computational complexity and computational time of large-scale dynamical systems by approximations of much lower dimension that can produce nearly the same input/output response characteristics.

The equations (1) for simplicity we present in form

$$dv(t)/_{dt} = Av(t) + F[v(t)]. \qquad (5)$$

Here $t \in [0, T]$ denotes time, $v(t)=[v_1(t),..., v_N(t)]^T \in \mathrm{R}^N$, $\mathrm{A} \in \mathrm{R}^{N\times N}$ is a constant matrix, and F is a nonlinear function evaluated at $v(t)$ component wise, i.e., $F = [F(v_1(t)),...,F(v_N(t))]^T$. The matrix A is the discrete approximation of the linear operator, and F is a nonlinear function of a variable v.

In MOR methods projection-based techniques are commonly used for constructing a reduced-order system. They construct a reduced-order system of order $k \ll N$ that approximates the original system from a subspace spanned by a *reduced basis* of dimension k in R^N. Let $V_k \in \mathrm{R}^{N\times k}$ be a matrix whose orthonormal columns are the vectors in the reduced basis. Then, by replacing $v(t)$ in (5) by $v(t) = V_k\ \hat{v}(t)$, $\hat{v}(t) \in \mathrm{R}^k$ and projecting the system (5) onto V_k, the reduced system of (5) is of the form

$$d\hat{v}(t)/_{dt} = V_k^T A V_k \hat{v}(t) + V_k^T F[V_k \hat{v}(t)]. \qquad (6)$$

The choice of the reduced basis clearly affects the quality of the approximation. The techniques for constructing a set of reduced basis use a singular value decomposition (SVD) of snapshots, which are discrete samples of trajectories associated with a particular set of boundary conditions and inputs. It is expected that the samples will be on or near the attractive manifold. Once the reduced model has been constructed from this reduced basis, it may be used to obtain approximate solutions for a variety of initial conditions and parameter settings, provided the set of samples is rich enough.

The paper (Chaturantabut & Sorensen, 2010) illustrates the computational inefficiency that occurs in solving the reduced-order system that is directly obtained from SVD approach. Equation (5) has the nonlinear term

$$NE(\hat{v}(t)) = V_k^T F[V_k \hat{v}(t)].$$

$NE(\hat{v}(t))$ has a computational complexity that depends on N, the dimension of the original full-order system (5). It requires a full evaluation of the nonlinear function F at the N-dimensional vector $V_k\hat{v}(t)$. As a result, solving this system might still be as costly as solving the original system.

One of way to overcome the difficulty is to approximate the nonlinear function in $NE(\hat{v}(t))$ by projecting it onto a subspace that approximates the space generated by the nonlinear function and that is spanned by a basis of dimension $m \ll N$. The discrete empirical interpolation method was proposed in paper (Chaturantabut & Sorensen, 2010) for these purposes.

Another idea described in (Lantsov, 2020) was to replace the vector of variables V of the HB equations (2) with two matrices of reduced dimension (Figure 1)

$$V = V_\mathrm{H} \cdot V_\mathrm{N},$$

where the matrix V_H reduces the number of harmonics and has dimension $[N\times R]$, V_N - reduces the number of nodes of the circuit and has dimension $[R\times(2K+1)]$, R is the reduced dimension of the equations, $R<<N$, $R<<(2K+1)$.

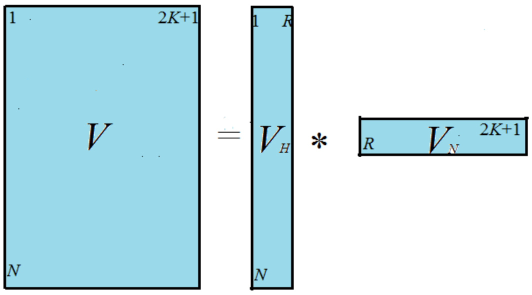

Figure 1. Replacing the matrix V with two new matrices of reduced dimensions

Replacement of (2) leads to two systems of balance equations of reduced dimension as it was obtained in (Lantsov, 2020a)

$$F(V_\mathrm{H}) = I(V_\mathrm{H}V_\mathrm{N}) \cdot V_\mathrm{N}^\mathrm{T} + YV_\mathrm{H} - I_\mathrm{E} \cdot V_\mathrm{N}^\mathrm{T} = 0 \qquad (7)$$

and

$$F(V_\mathrm{N}) = V_\mathrm{H}^\mathrm{T} \cdot I(V_\mathrm{H}V_\mathrm{N}) + Y V_\mathrm{N} - V_\mathrm{H}^\mathrm{T} I_\mathrm{E}\ . \qquad (8)$$

The equation (7) reduces the number of harmonics in the balance equations and has dimension of equations as $[N\times R]$. The equation (8) reduces the number of circuit nodes and has dimension of equations as $[R\times(2K+1)]$. Equations with reduced dimension are solved sequentially.

In the traditional circuit simulators, the more convenient form of the equations for Newton's method is on the new value V^{i+1} instead of the increment ΔV^{i+1} in (3)

$$J(V^i) \cdot (V^{i+1} - V^i) = -F\ (V^i),$$
$$J(V^i) \cdot V^{i+1} = I_E + \partial I/_{\partial V}\,|_{V^i} - I(V^i),$$

As a result, the final expression for calculations (7) by Newton's method gives us the following iterative formula in the form of the system of linear algebraic equations (SLAE) (Lantsov, 2020b)

$$J(V_\mathrm{H}^i) \cdot V_\mathrm{H}^{i+1} = I_\mathrm{H}^i. \qquad (9)$$

Here the Jacobi matrix is defined as

$$J(V_\mathrm{H}^i) = \partial F/_{\partial V_\mathrm{H}} = \partial I/_{\partial V} \cdot V_\mathrm{H}^\mathrm{T} + Y. \qquad (10)$$

The vector of the right parts of the system of equations (9) will have the form (Lantsov, 2020b)

$$I_{\mathrm{H}}^{i} = I_{\mathrm{E}} V_{\mathrm{N}}^{\mathrm{T}} - {}^{\partial I}/_{\partial V}\, V_{\mathrm{H}}^{i} - I(V)\, V_{\mathrm{N}}^{\mathrm{T}}. \qquad (11)$$

Similarly, to solve the system of equations (8), we have the following basic iterative formulas (Lantsov, 2020b)

$$J(V_{\mathrm{N}}^{i}) \cdot V_{\mathrm{N}}^{i+1} = I_{\mathrm{N}}^{i}\,, \qquad (12)$$

$$J(V_{\mathrm{N}}^{i}) = {}^{\partial F}/_{\partial V_{\mathrm{N}}} = V_{\mathrm{H}}^{T} \cdot {}^{\partial I}/_{\partial V} + Y, \qquad (13)$$

$$I_{\mathrm{N}}^{i} = V_{\mathrm{H}}^{\mathrm{T}}\, I_{\mathrm{E}} - {}^{\partial I}/_{\partial V}\, V_{\mathrm{N}}^{i} - V_{\mathrm{H}}^{\mathrm{T}}\, I(V). \qquad (14)$$

3. APPROXIMATION OF DERIVATIVES FOR NONLINEAR DEPENDENCIES

Note that in equations (11) and (14) there is an element $I(V)$ that determines the dependence of current on voltages on nonlinear elements in the frequency domain. In standard of circuit simulation tools, all models of nonlinear devices (elements) are described by dependencies in the time domain $i_{\mathrm{NE}}(t) = f\,[v_{\mathrm{NE}}(t)]$. The basic HB equations are solved in the frequency domain. Therefore, at each iteration of Newton method, when solving equations, it is necessary to make transformations from the frequency domain to the time domain and back using the Fast Fourier Transform (Lantsov & Papulina, 2021, Carvalho et al., 2006, Nastov et al., 2007, Lantsov & Papulina, 2021)

$$I(V) = \Gamma\,\{i(t) = f[v(t)]\}\,\Gamma^{-1}. \qquad (15)$$

Here Γ is the forward and Γ^{-1} is the inverse Fourier transform. The relationship between the representation of the signal in the time and frequency domains will be defined as $V = \Gamma v$ and $v = \Gamma^{-1} V$. Similarly, in the iterative formulas there is an element of calculating the derivatives of currents of nonlinear elements according to the corresponding voltages

$${}^{\partial I}/_{\partial V} = \Gamma \left\{ {}^{\partial i(t)}/_{\partial v(t)} \right\} \Gamma^{-1}. \qquad (16)$$

Earlier, in previous works (Lantsov, 2020, Lantsov & Papulina, 2021), the standard part of traditional HB-based software tools and the full dimensions of these equation elements were used to calculate (15) and (16). Unfortunately, the experience of using software tools implementing previously developed methods and algorithms has shown that the main computational costs (memory and time) are associated precisely with the calculation of element $I(V)$, and especially with the calculation of the derivative for the Jacobian ${}^{\partial I}/_{\partial V}$.

Note that the element ${}^{\partial I}/_{\partial V}$. will be the same for equations (10 and 11), (13 and 14). It is proposed to approximate this element and reduce its dimension using a method widely used in recent years based on Singular Value Decomposition (SVD) (Antoulas & Sorensen, 2001, Chaturantabut & Sorensen, 2010, lantsov, 2012, Charumathi et al., 2019).

This approximation will correspond to the methods of averages used in the algorithms for solving the HB equations to approximate the Jacobian in (Filicori & Monaco, 1988, Kundert et al., 1990, Ushida et al., 1992), where equation (3) can be replaced as

$$J(V^{j}) = {}^{\partial F}/_{\partial V}\,|_{V^{j}} = Y + G_{av}.$$

In (Kundert et al., 1990) average value was obtained as

$$G_{av} = \frac{1}{T_0} \int_{0}^{T_0} \frac{\partial i[v(t)]}{\partial v(t)}\, dt.$$

It is important to note that the approximation is performed once and before the start of the main iterations.

In this paper we use another approach. The collection of output waveforms (snapshots or samples of trajectories) is discretized and the times at which the derivatives are crossed are recorded. The discretized input and output waveforms can be collected in one single matrix form (snapshot matrix) of responses of nonlinear elements in the form (Ramalingam, 2007)

$$G(t) \coloneqq [g_1(t), \ldots, g_n(t)]^{\mathrm{T}} \in R^{n},$$

where $g_k(t) = {}^{\mathrm{d}i(t)}/_{\mathrm{d}u(t)}\,|_{u_{k(t)}}$ are derived currents of nonlinear elements (usually in circuit simulators, these derivatives are in analytical form); time $t \in [0, T]$, T is the period of the lowest harmonic spectrum at the output of the circuit; $n = N_{\mathrm{NE}} \cdot N_{\mathrm{IN}}$; N_{NE} is a number of nonlinear dependencies; N_{IN} is a number of analysis when we change the amplitude of the input.

The total dimension of the matrix of sets G will be defined as $n{\cdot}N$t, where Nt is the number of time (discrete) samples on the period T. The correct and sufficiently large for selection of a set of snapshots is a decisive factor in building of the SVD algorithm. This choice can greatly affect the approximation of the initial description (Ramalingam, 2007).

It is assumed that the sample set will correspond to the dominant states of the model and the sample set is large enough. The SVD method creates a reduced basis, which is optimal in the sense that the approximation error with respect to data sets is minimized (Ramalingam, 2007).

The SVD algorithm with respect to the matrix of sets G gives the following decomposition

$$G = U\,\Sigma\,V^{\mathrm{T}}, \qquad (17)$$

where the matrices $U := (u_1, \dots, u_r) \in R^{r\times r}$ and $V := (v_1, \dots, v_r) \in R^{Nt\times r}$ are orthogonal, i.e. *U V =I*, where *I* is the unit matrix, $\Sigma = diag(\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r \geq 0)$ is a diagonal matrix of singular values.

The matrix *U* is called the left singular matrix and its columns provide the orthogonal basis for the columns of *G*. The matrix *V* is called the right singular matrix and its columns provide the orthogonal basis for the rows of *G*. As noted above, the columns of *V* provide the orthogonal basis for the rows of *G* and since each row contains a discretized waveform, the columns of *V* turn out to be the orthogonal basis for the waveforms in matrix *G*.

Now (17) can be rewritten by post-multiplying both sides by *V*. Since *V* is orthonormal ($V^TV = I$) we get $GV = U\Sigma$. We denote the resultant product matrix as *M* called the moments matrix (Ramalingam, 2007) because this is another way to represent time points just like an equivalent representation of any function by its moments:

$$M = GV = U\Sigma.$$

The moments matrix defined here is a linear combination of time points weighed by the right singular vectors (rsv) *V.j*:

$$m_{ij} = G_{i.} V_{.j} = \sum_{k=1}^{n} g_{ik} v_{kj},$$

where

$$G = \begin{pmatrix} g_{1,1} & g_{1,2} & \dots\ g_{1,n} \\ g_{2,1} & g_{2,2} & \dots\ g_{2,n} \\ & \dots & \\ g_{Nt,1} & g_{Nt,2} & \dots\ g_{Nt,n} \end{pmatrix}.$$

The right singular vectors transform a waveform from time domain $t = (t_1, t_2, \dots, t_n)$ to moments domain $m = (m_1, m_2, \dots, m_n)$ through $m = tV$ and vice-versa through $t = mV^T$.

This equivalent representation leads to an interesting possibility in the context of timing analysis. If a waveform can be represented accurately using a few moments then by propagating these moments, one can do an accurate waveform analysis instead of propagating all the *n* time points.

Suppose we represent a waveform by *r* moments, where $r < n$ then the last $n - r$ moments are set to zero. The process of setting the last $n - r$ moments to zero is equivalent to setting the last *n* - *r* singular values to zero since zeroing singular value will force the corresponding moment to zero. But zeroing out singular values is equivalent to approximating a matrix *G* with another matrix $\hat{G}$ having a smaller rank. To measure the goodness of approximation, Frobenius norm is used. This norm measures the goodness of fit in a root mean square fashion. The Frobenius norm of *G* is defined as

$$||G||_F = \sqrt{(\sum_{i=1}^{Nt} \sum_{j=1}^{n} g_{ij}^2)}.$$

There is an equivalent way to compute the Frobenius norm of a matrix by using the singular values of a matrix.

The above discussion can be summarized by saying that the following statements are equivalent (Ramalingam, 2007).

- Approximating a waveform G_i. using the first *r* moments.
- Approximating a matrix considering the first *r*-singular values of matrix *G*.
- A rank-*r* approximation of matrix *G* in Frobenius norm.

A reduced matrix of averaged values of G_{av} is obtained using the following relation (Ramalingam, 2007)

$$G_{av} = U\,\Sigma\,V^T = U\,diag(\sigma_1,\ \sigma_2, \dots, \sigma_r,\ 0, 0, 0)\ V^T.$$

To coordinate the dimensions in the balance equations, the value of *N* can be taken as the value of *r*, which is easily done in most cases.

The final expressions for calculation by iterative formulas will look like this. For equations (10-11)

$$J(V_H^i) = Gav \cdot V_N^T + Y, \tag{18}$$

$$I_H^i = I_E V_N^T - Gav \cdot V_H^i - I(V)\, V_N^T. \tag{19}$$

For equations (13-14)

$$J(V_N^i) = V_H^T \cdot Gav + Y, \tag{20}$$

$$I_N^i = V_H^T\, I_E - Gav \cdot V_N^i - V_H^T\, I(V). \tag{21}$$

Algorithm 1 (Solution of equations 9)

```
Input: $V_H^i, V_N^i, Gav$;
$V_N^T = (V_N^i)^T$;
Access to the library of standard non-linear
    elements. Calculation of $I(V^i)$, equations (15);
Calculation (18) and (19);
Solution of SLAE (9).
```

Algorithm 2 (Solution of equations 12)

```
Input: $V_H^i, V_N^i, Gav$;
$V_H^T := (V_H^i)^T$;
Access to the library of standard non-linear
    elements. Calculation of $I(V^i)$, equations (15);
Calculation (20) and (21);
Solution of SLAE (12).
```

Algorithm 3 (Newton Iterations)

```
1:  DC analysis
    $V_H^0 = V_H^{DC};\ V_N^0 = V_N^{DC};$
2:  Calculation $Gav$;
3:  $i$ = 0;
4:  Algorithm 1;
5:  Algorithm 2;
6:  if $(V_H^{i+1} - V_H^i \leq \varepsilon) \wedge (V_N^{i+1} - V_N^i \leq \varepsilon)$ stop.
7:  $i = i + 1$;
8:  go to 4.
```

All the developed algorithms are implemented as a supplement to the general-purpose circuit modeling program SMORES (Bond, 2010), implemented as open code in the Matlab/Simulink system.

Verification of the proposed algorithms based on the circuit simulation software tools (Lantsov, 2020, Lantsov & Papulina, 2021) was performed on a well-known example from the author's previously published works and examples known from publications in the world literature (Fig. 3, Bond & Daniel, 2007, Fig. 1, Lantsov & Dolinina, 2016).

This example is characterized by the fact that by changing the number of nonlinear cascades of the circuit, it is possible to increase the dimension of the circuit to very high dimensions determined by the memory of a particular computer.

The comparison was performed both with the standard HB method and with previously published results. Since the convergence in all the examples coincided with the calculations by the standard HB method, we do not provide graphs of the results.

The table 1 shows the results of the comparison.

Table 1. Comparison of methods by calculation time

	Calculation options: the number of repeating cascades of the circuit, the number of nodes, the number of harmonics taken into account	New algorithm	Standard HB method	The algorithm is in paper (Lantsov, 2020a)]
1	3, 11, 3	5 s	4 s	6 s
2	3, 11, 99	9 s	9 s	11 s
3	14,42, 3	20 s	22 s	23 s
4	14, 42, 99	112 s	135 s	128 s
5	45, 135, 3	175 s	206 s	196 s
6	45, 135, 99	271 s	342 s	302 s

4. CONCLUSION

The results of modeling of circuits that differ in the number of electronic circuit nodes and the number of harmonics taken into account showed that the gain was obtained for almost all variants.

The new algorithm is suitable for solving large and super-large harmonic balance equations.

References:

Antoulas, A. C., & Sorensen, D. C. (2001). Approximation of large-scale dynamical systems. An overview. *International Journal of Applied Mathematics and Computer Science, 5(11)*, 1093-1121.

Bond, B. N., & Daniel, L. (2007). A piecewise-linear moment-matching approach to parameterized model-order reduction for highly nonlinear systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 12(26)*, 2116-2129. Doi:10.1109/TCAD.2007.907258.

Bond, B. N., & Daniel, L. (2009). Stable reduced model for nonlinear descriptor systems through piecewise-linear approximation and projection. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 10(28)*, 1467-1480.

Bond, B. N.(2010). SMORES: A Matlab tool for Simulation and Model Order Reduction of Electrical Systems [Preprint]. Massachutes Institute of Technology.

Carvalho, N. B., Pedro, J. C., Jang, W., & Steer, M. B. (2006). Nonlinear RF circuits and systems simulation when driven by several modulated signals. *IEEE Transactions on Microwave Theory and Techniques, 2(54)*, 572-579. Doi:10.1109/TMTT.2005.862713.

Charumathi, V., Ramakrishna, M., & Vasudevan, V. (2019). Fast proper orthogonal decomposition using improved sampling and iterative techniques for singular value decomposition. Retrieved from https://arxiv.org/abs/1905.05107.

Chaturantabut, S., Sorensen, D.C. (2010). Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal of Science and Computing. 5(32)*, 2737-2764.

Filicori, F., & Monaco, V. A. (1988). Computer-aided design of non-linear microwave circuits. *Alta Frequenza, 7(57)*, 355-378.

Gilmore, R. J., & Steer, M. B. (1991). Nonlinear circuit analysis using the method of harmonic balance. A review of the art. Part 1. Introductory concepts. *International Journal of Microwave- and Millimeter-Wave Computer-Aided Engineering, 1*(1), 22-37.

Kundert, K. S., Sorkin, G. B., & Sangiovanni-Vincentelli, A. (1988). Applying harmonic balance to almost periodic circuits. *IEEE Transactions on Microwave Theory and Techniques, 2*(36), 366-378. doi:10.1109/22.3525.

Kundert, K., White, J., & Sangiovanni-Vincentelli, A. (1990). *Steady-State Methods for Simulating Analog and Microwave Circuits*. New York, NY: Kluwer Academic Publishers.

Kundert, K. (1999). Introduction to RF simulation and its application. *IEEE Journal of Solid-State Circuits, 9(34)*, 1298-1319. doi: 10.1109/4.782091.

Lantsov, V.N. (2012). Order-reduction methods for models of complex electronic circuits (review). *Radio and telecommunication systems. 3*, 59-65.

Lantsov, V. N., & Dolinina, A. A. (2016). Method of lowering the order of models based on Volterra series. *Dynamics of Complex Systems, 3*(10), 50-54.

Lantsov, V. (2020a). A new algorithm for solving of harmonic balance equations by using the model order reduction method. *Biomedical Engineering, Radioelectronics and Information Technology* (pp. 295-297). Ekaterinburg, Russian Federation: UFU.

Lantsov, V. N. (2020b). A new algorithm of solving harmonic balance equations. *Journal of IOP Conference Series: Material Science and Engineering, 862(5)*, paper № 052031. doi: 10.1088/1757-899X/862/5/052031

Lantsov, V., & Papulina, A. (2021). Algorithms for software tools with harmonic balance method. *Biomedical Engineering, Radioelectronics and Information Technology* (pp. 228-230). Ekaterinburg, Russia: UFU.

Nastov, O., Telichevsky, R., Kundert, K., & White, J. (2007). Fundamentals of fast simulation algorithms for simulation algorithms for RF circuits. *Proceedings of IEEE. 3(95)*, 600-621. doi: 10.1109/JPROC.2006.889366.

Ramalingam, A., Singh, A. K., & Nassif, S. R. (2007). Accurate waveform modeling using Singular Value Decomposition with applications to timing analysis. *IEEE Design Automation* (pp. 148-153). San Diego, USA: ACM.

Rizzoli, V., Cecchetti, C., Lipparini, A., & Mastri, F. (1988). General-purpose harmonic balance analysis of nonlinear microwave circuits under multitone excitation. *IEEE Transactions on Microwave Theory and Techniques, 12*(36), 1650-1659. Doi:10.1109/22.17396.

Rizzoli, V., Masotti, D., Mastri, F., & Montanari, E. (2011). System-oriented harmonic-balance algorithms for the circuit-level simulation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2*(30), 256-269. Doi: 10.1109/TCAD.2010.2092250.

Ushida, A., Adachi, T., & Chua, l. O. (1992). Steady-state analysis of nonlinear circuits based on hybrid methods. *IEEE Transactions on Circuits and Systems, 8*(39), 649-661.

White, J. K., & Sangiovanni-Vincentelli, A. (1987). *Relaxation techniques for the simulation of VLSI circuits*. New York, NY: Kluwer Academic Publishers.

Vladimir Lantsov
Vladimir State University,
Vladimir,
Russian Federation
lantsov@ieee.org
ORCID 0000-0003-4927-2640