8

Macromodels of Micro-Electro-Mechanical Systems (MEMS)

Anatoly Petrenko
Systems Design Department, National Technical University of Ukraine
Ukraine

1. Introduction

Micro-electro-mechanical Systems (MEMS) are components with micron-scale moving parts based on materials and processes of microelectronics fabrication. This is a good example of on-chip integration of electronics, microstructures, microsensors and microactuators. Accurate simulation of MEMS requires precise modeling of all effects of mechanical and damping forces, electrostatic forces and inner stresses, heat transfer, thermal expansion, piezoelectric stresses etc.

Modern methodology of MEMS design implies that the entire MEMS can be investigated only at higher abstraction levels such as **schematic and system** ones, where accurate macromodels can be used [1]. On the other hand, at component or device levels the physical behavior of three-dimensional continuums is described by **partial differential equations** (PDE) easily solvable by Finite Element or Finite Difference Element Methods (FEM or FDM) [2,3], available in ANSYS –like software. Component level simulations are classified in single - domain and coupled - domain simulations, both being very computer time- consuming.

The goal of this chapter is to consider methods of automatically obtaining macromodels of MEMS and their mechanical or non-electric components from ANSYS models as equivalent electric circuits or low order differential ordinary equations for further use in circuit design software. This can be done by using different model order reduction techniques developed in recent years.

When dealing with the modern MEMS, the possibility for using a single environment to simulate objects, where different physical processes such as electrical, mechanical, optical, thermal etc. take place, plays an important role. Here we have to represent different subsystems of the initial MEMS as equivalent models of the same physical nature permitting to combine them for solution in a single computational process. After that, the complete behavioral model of the entire MEMS and its subsystems can be compiled either in VHDL-AMS language (as sets of ODE) or in SPICE-like language (as equivalent electric circuits).

The Microsystems design exploits various analytical and numerical methods for virtual prototyping of MEMS. It also demands for libraries of electromechanical, optical models and microfluid components, including springs, bulks, buffers, capacitors, inductances, operational amplifiers, transistors etc. Three basic possible approaches of MEMS design procedure are illustrated below: FEM/FDM Model, Reduced Order Model (ROM), Coupled system-level model.

2. FEM/FDM model

MEMS typically involve multiple energy domains such as kinetic energy, elastic deformation, electrostatic or magneto static stored energy and fluidic interactions. The difficulty in the modeling of MEMS devices is mainly caused by the tight coupling between the multiple energy domains. Individual physical effects are governed by partial differential equations (PDE), typically nonlinear. When these equations become coupled, the computational challenges of highly meshed numerical simulation become formidable.

FEM relies on highly localized interpolation functions (or *mesh element functions*) for approximation of the solution of PDE. These mesh element functions are generated by meshing the domain of interest and parameterize the desired solution locally on each mesh element. This parameterized solution converts a continuous (PDE) problem to a coupled system of ordinary differential equations (ODE) that can be integrated in time. The resulting ODE system usually has many degrees of freedom (perhaps **several variables per mesh element**). If a fine mesh is required, the problem size grows rapidly, with a corresponding rapid growth in computational cost for explicit dynamic simulation. Consequently, it is very expensive to use FEM model in system-level simulations during MEMS iterative design. As a result, FEM models are mostly used to analyze the performance of MEMS components and to couple their multiphysics effects.

By reading ANSYS binary FULL file it is possible to assemble a MEMS component state-space model in the form of first order systems or second order ordinary differential equations (ODE)

$$E_r z' + A_r z = B_r f \ , \qquad Y = C_r z \tag{1}$$

$$\mathrm{M}x'' + \mathrm{D}x' + \mathrm{K}x = \mathrm{B}f \ , \ \mathrm{Y} = \mathrm{Q}^{\mathrm{T}}x + \mathrm{R}^{\mathrm{T}}x' \ , \tag{2}$$

where A_r, E_r ,C_r, B_r, M ,D, K, B, C- are the system matrices, B_r, B are the input and the C_r, C - output matrices, f is input force. In mechanics matrices M, D and K are known as the *mass, damping* and *stiffness* matrices correspondingly. Usually damping is included in the model as Rayleigh damping. The damping matrix D is computed as a linear combination of the stiffness K and the mass M matrices:

$$\mathrm{D} = \alpha\,\mathrm{M} + \beta\,\mathrm{K},$$

where a, β are constant coefficients.

In (1) the state space vector z is defined through the unknowns deflections $u(x,t)$ and pressures $p(x,y,t)$ into the node points being automatically generated in MEMS structure:

$$z = [u_1 \ldots u_N \ \frac{\partial u_1}{\partial t} \ldots \frac{\partial u_N}{\partial t} \ p_{11} \ldots p_{MN}]^T \tag{3}$$

By defining

$$E_r = \begin{bmatrix} D & M \\ M & 0 \end{bmatrix} \quad A_r = \begin{bmatrix} K & 0 \\ 0 & -M \end{bmatrix} \quad B_r = \begin{vmatrix} B \\ 0 \end{vmatrix} \quad C_r = \begin{vmatrix} Q \\ R \end{vmatrix} \quad z = \begin{vmatrix} x' \\ x'' \end{vmatrix} \tag{4}$$

second equations (2) can be transfer to the first (1).

The FULL file contains all the information about the system: the system element matrices, Dirichlet boundary conditions, equation constrain and the load vector. It is generated using ANSYS partial solver, which enables to assemble system element matrices for the desired analysis without solving them and it therefore computationally fast. The speed of the reading operation has been optimized taking into account that the element matrices are sparse. The load vector directly gives the matrix-vector product *Bf* and thus describes the distribution of all loads being applied. In order to obtain the *B* matrix, and thus being able to modify the inputs singularly, it is necessary to repeat the partial solution for each input of interest.

3. ROM (Reduced Order Model)

It would be easier and more intuitive for the designer to explore the design space if the MEMS model had only a few variables with a clear relationship between them and the overall device performance. *Reduced-order models (ROM)*, also called *macromodels,* lend themselves very well to these purposes. The main idea behind the reduced order model is that the number of ordinary differential equations (ODE) needed to simulate the system has been reduced from perhaps many thousands in the case of the full FEM simulation, to just a few basis function coordinates (fig.1).

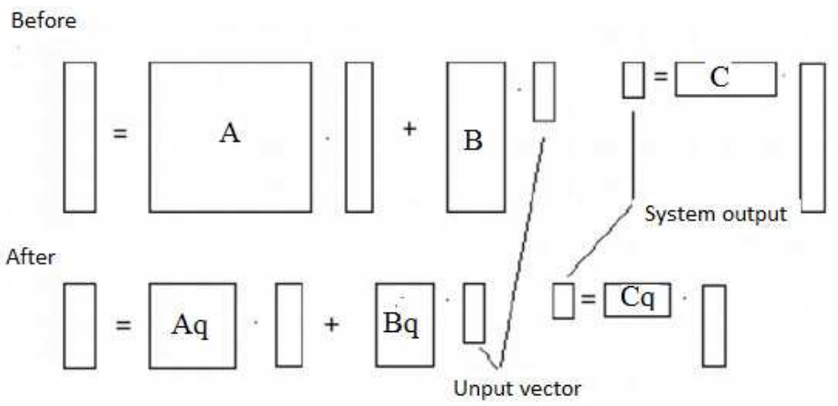


Fig. 1. Reduced order model illustration

Such the macromodel simulation can be very efficient computationally compared to the FEM model. A designer can use the FEM model for different component geometry and materials trying and the ROM model for investigation of different input forces effect (fig.2).

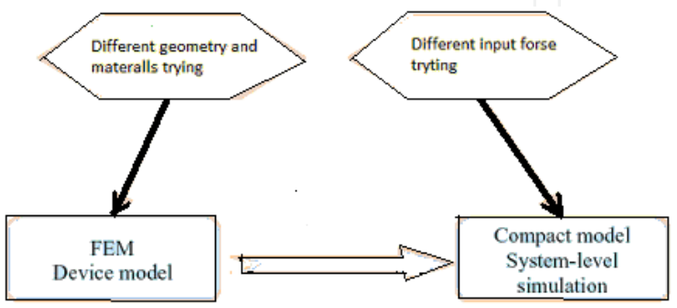


Fig. 2. Compact reduced order model in MEMS design [33]

3.1 Modal decomposition ROM

Modes (or resonances) are inherent properties of a structure. Resonances are determined by the material properties (mass, stiffness, and damping properties) and boundary conditions of the structure. Each mode is defined by a *natural (modal or resonant) frequency*, *modal damping*, and a *mode shape*.

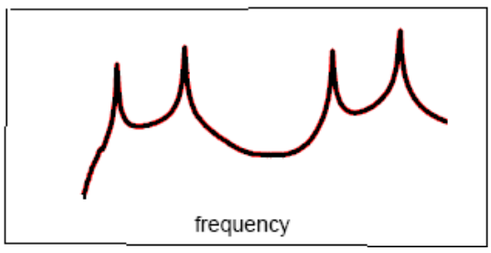


Fig. 3. Simple Plate Frequency Response Function

If either the material properties or the boundary conditions of a structure change, its modes will change as well. The overall response of a structure at any frequency is a *summation of responses due to each of its modes*. It is also evident that close to the frequency of one of the resonance peaks, the response of *one mode will dominate the frequency response* [6], fig.3.

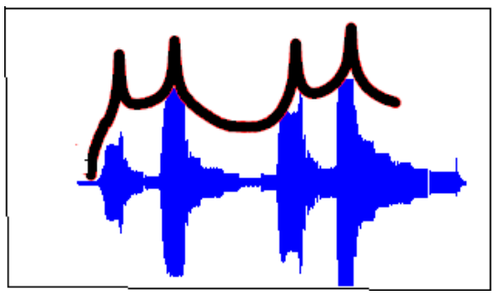


Fig. 4. Overlay of Frequency and Time Response Functions

Now if we overlay the time trace with the frequency trace it is possible to notice that maximum oscillations at the time are corresponding to the frequencies of maximum frequency response function (fig.4).

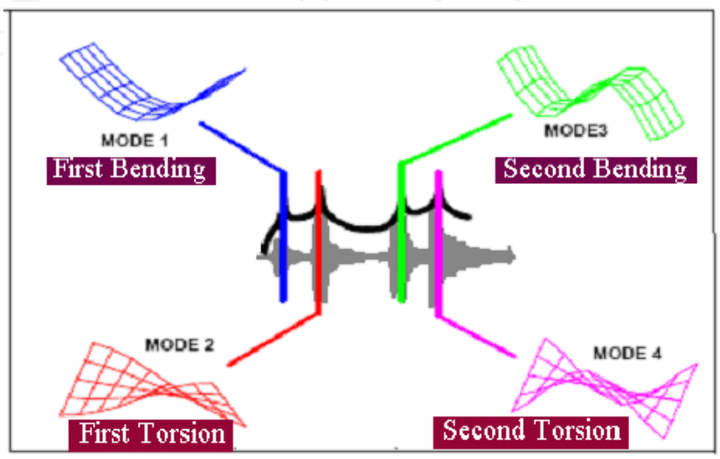


Fig. 5. Flexible Body Modes [6].

Let's see what happens to the deformation pattern on the structure at each one of these natural frequencies. Modes are further characterized as either *rigid body* or *flexible body* modes. All structures can have up to *six rigid body* modes, three translational modes and three rotational modes. Many deformation problems are caused, or at least amplified by the excitation of one or more flexible body modes. Fig.5 shows some of the common fundamental (low frequency) modes of a plate. The fundamental modes are given names like those shown in Fig.5. The higher frequency mode shapes are usually more complex in appearance, and therefore don't have common names.

Modal decomposition method uses a weighted sum of n mode shapes (modal amplitudes or eigenvalues) q_i, basis function (an eigenvector) $\varphi_i\ (x,\ y,\ z))$ of the mechanical structure to represent its deflection u:

$$u(x,y,z,t) = u_{eq} + \sum_{i=1}^{n} q_i(t)\varphi_i(x,y,z) \ , \tag{5}$$

where u_{eq} is the initial displacement produced by the initial load. For MEMS it is usually sufficient that few modes accurately describe dynamical response of the system. This approach is equivalent to the projection of the original PDE, describing the MEMS behavior, on the subspace defined by the basis functions.

By inserting (5) into the equation of motion (2), taking $\mathbf{x} = \varphi\, e^{\lambda t}$ and multiplying by φ^T_i from the left, and using the orthogonality of eigenvectors, the equation (2) can be reduced to

$$\ddot{q}_i - 2\sigma_i q_i + (\omega_i^2 + \sigma_i^2) = \varphi_i^T f \ , \quad i=1,2,\ldots n \tag{6}$$

The eigenvectors satisfy the orthogonality conditions $\varphi^T_k\, M\varphi_i = \varphi^T_k\, D\varphi_i = \varphi^T_k\, K\varphi_i = 0$ for $\lambda_k \_\neq \lambda_i$. With the normalization of the eigenvectors $\varphi^T_k\, M\varphi_k \equiv 1$, it can be shown that

$$\varphi^T_k\, D\varphi_k = -2\sigma_k \text{ and } \varphi^T_k\, K\varphi_k = \omega^2_k + \sigma^2_k \text{ if } \lambda_k = \sigma_k + j\omega_k.$$

Note that the orthogonality condition does not apply in the case of multiple eigenvalues. In such case, special modal analysis techniques are needed for decoupling of modes [7].

The decoupling of modes yields that transfer functions can be written as a sum of *modal transfer functions*. Using Fourier transforming expansion and equations (6), the transfer matrix can be derived as

$$H(\omega) = \sum_{i=1}^{n} H_k(\omega) = \sum_{i=1}^{n} \frac{\varphi_i \varphi_i^T}{(j\omega - \sigma_i - j\omega_i)(j\omega - \sigma_i + j\omega_i)} \tag{7}$$

This relation is the basis of modal analysis. It relates the *measurable* transfer functions to the modal properties ω_i, σ_i, and φ_i. Each i- th mode contributes with a modal transfer matrix H_i to the complete transfer matrix. Fig.6 shows an example of a theoretical transfer function (7) with three *modal peaks* corresponding to modes at 1, 2, and 4 Hz which are all damped with a logarithmic decrement of 1% ($-\sigma_i/f_i = 0.01$). Also the individual modal transfer functions (7) are plotted from which the complete transfer function is computed.

By the way the eigenvalues q_k and the corresponding eigenvectors φ_k for $k = 1, 2, \ldots, n$ can be found as an solution of the modified eigenvalue problem

$$(M\lambda^2 + D\lambda + K)\varphi = 0 \ . \tag{8}$$

The relationships between natural frequencies f_k, logarithmic decrements δ_k and the eigenvalues are $f_k = 2\pi\omega_k$ and $\delta_k = -\sigma_k/f_k$.

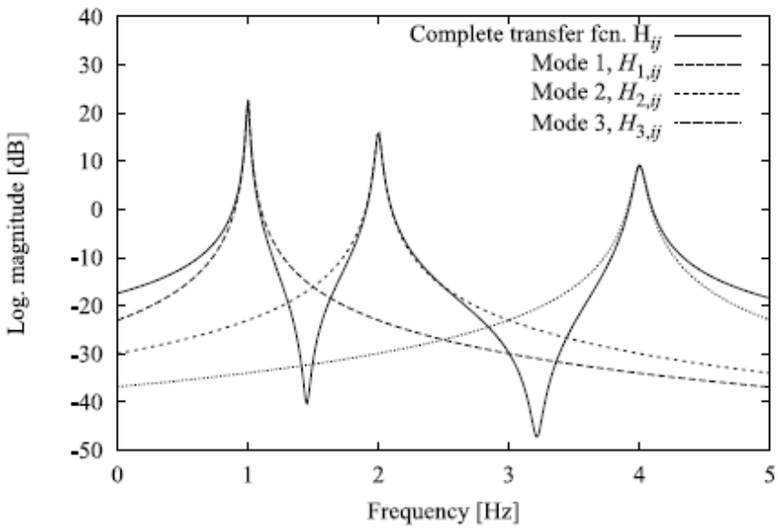


Fig. 6. Example of the modal decoupling in a transfer function [7].

3.1.1 Modal ROM based on proper orthogonal decomposition methods

Reduced order modeling using modal basis functions was originally developed by [11-13] and has been continuously improved by several authors. The basis functions $\varphi_i(x,y,z)$ in (5) may be chosen in two ways by:

- Using the undamped linear mode shapes of the undeflected microstructure as basis functions. For simple structures with simple boundary conditions, the mode shapes can be found analytically. For complex structures or complex boundary conditions, the linear mode shapes are obtained numerically using the finite element method. However, it is usually difficult to determine, *a priori*, an optimum set of eigenvectors φ_k, particularly when irregular geometries are involved.
- Using snapshots obtained from experiments under a training signal or full FEM model runs. Then the proper orthogonal decomposition method (Singular Value Decomposition –*SVD* , Karhunen -Loève decomposition -*KL* and neural networks-based generalized Hebbian algorithm -*GHA*) is applied to the time series for extracting the mode shapes of the device structural elements.

The choice of orthogonal basis functions φ_k can be done by the following way [8]. First the MEMS dynamics are simulated using a slow but accurate technique such as FEM or FDM. Sets of runs may be used to suitably characterize the operating range of the device. The spatial distributions of each state variable $u(x, y, z, t)$ are then sampled at a series of N_s different times during these simulations, and the sampled distributions are stored as a series of vectors, $\{u_i\}$, where each of them corresponds to a particular "snapshot" in time. Now suppose we would like to pick orthogonal basic n functions $\{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ in order to represent the observed state distributions as closely as possible. One way to do this is to attempt to minimize a least squares measure of the "error" distances between the observed states and the basis function representation of those states:

$$\sum_{i=1}^{n}[u_i - proj(u_i, span\{\varphi_1,...,\varphi_n\})]^2 = \sum_{i=1}^{n}\sum_{j=1}^{n}\varphi_i\varphi_j^T u_i \,, \tag{9}$$

where $proj(v,S)$ is the projection of the vector v into subspace S. In other words, we minimize a least squares measure of the "error" distances between the observed states and the basis function representation of those states.

Singular value decomposition (SVD) takes a rectangular matrix of modal experimental data (defined as U, where U is a N x n matrix) in which N is a number spatial points of each snapshot, n is a number of snapshots. The *SVD* theorem states:

$$U_{Nxn} = W_{NxN}\, S_{Nxn}\, V^T_{nxn} \,,$$

where the columns of W are the left singular vectors (*spatial points vectors*); S (the same dimensions as U) has singular values and is diagonal (*mode amplitudes*); and V^T has rows that are the right singular vectors (*snapshots numbers vectors*).

The *SVD* represents an expansion of the original data in a coordinate system where the covariance matrix is diagonal. Calculating the *SVD* consists of finding the eigenvalues and eigenvectors of matrices UU^T [NxN] and U^TU [n x n]. The eigenvectors of U^TU make up the columns of V, the eigenvectors of UU^T make up the columns of W. Also, the singular values in S are square roots of eigenvalues from UU^T or U^TU. The singular values are the diagonal entries of the S matrix and are arranged in descending order: $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n \geq 0$ .The singular values are always real numbers. If the matrix U is a real matrix, then W and V are also real.

It was shown [11] that the proper orthogonal basis functions $\{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ minimizing (9) can be chosen by setting

$$\phi_i = v_i \,,\ i = \{1,2,...,n\}$$

where v_i is the columns of V.

After finding basis functions φ_k the linear equations for modal amplitudes q_k estimation are formed for each temporal snapshot on the base of selected before "snapshot" spatial points and equation (5).

Karhunen-Loève decomposition (KL) can be viewed as a statistical procedure [9] .One initially supposes that the observed system dynamics can be modeled as a second-order ergodic stochastic process. The method consists then in constructing a spatial autocorrelation tensor from data obtained through numerical or physical experiments and performing its spectral decomposition.

KL differs from *SVD* in way of finding basis functions φ_i through the temporal snapshot correlation matrix entries $a_{ij} = \frac{1}{n}\sum_{\substack{k=1 \\ m=1}}^{n}[u_i(x,t_m)u_j(x,t_k)]$

as $$\varphi_i(x) = \sum_{k=1}^{n} b_{k,i}u_k(x) \,,$$

where n is number of temporal snapshots and $b_{k,i}$ are eigenvectors of the matrix A, or the solutions of the equation $Ab=\lambda b$.

Generalized Hebbian algorithm (GHA) is an Artificial Neural Network (ANN) approach of performing Principal Component Analysis (PCA) on a set of data and can be also used as a learning procedure for the approximation of *PDE* solutions by the expression (5) [10] .
The modal approach to MEMS macromodeling is illustrated by sensor device (fig.7), which is described by coupling a 1-D elastic beam equation with electrostatic force and 2-D compressible isothermal squeeze-film Reynold's equation [11]. This device consists of a deformable elastic beam microstructure that is electrostatically pulled in by an applied voltage waveform. The dynamics of the beam are first simulated using a finite-difference analysis. A quarter of the beam is initially meshed using a 20 x10 node 2-D grid.

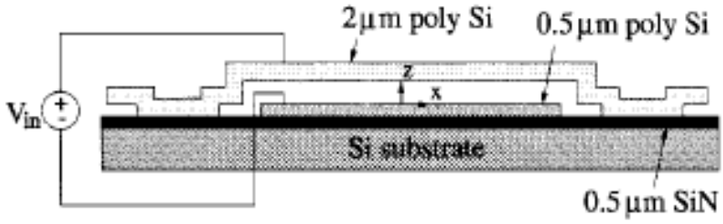


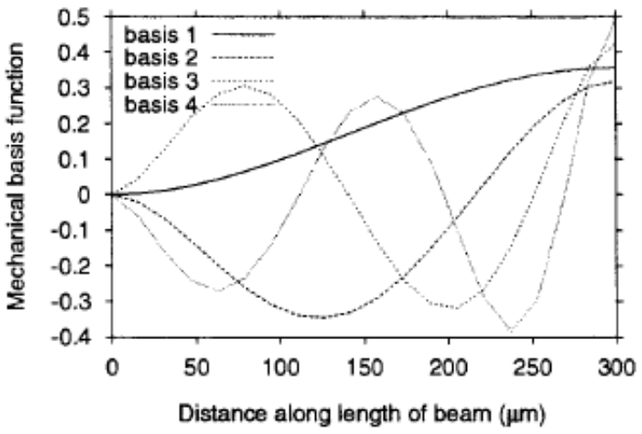
Fig. 7. Fixed-fixed beam pull-in time pressure sensor device [11]

The state at each node consists of three quantities: *z, dz/dt* and *p*. Since *z* and *dz/dt* are simulated in 1-D, this results in coupled nonlinear ODE which must be integrated in time. Basis functions are generated for pressure *p* and displacement *z* based on runs of the finite-difference code for an ensemble of four different step voltages: 9V, 10V, 12V, 16V. One hundred samples of pressure and displacement are taken during these four runs at fixed time intervals. These samples are used to generate the basic functions. The resulting basis functions for displacement and pressure are shown in Fig.8 [11].

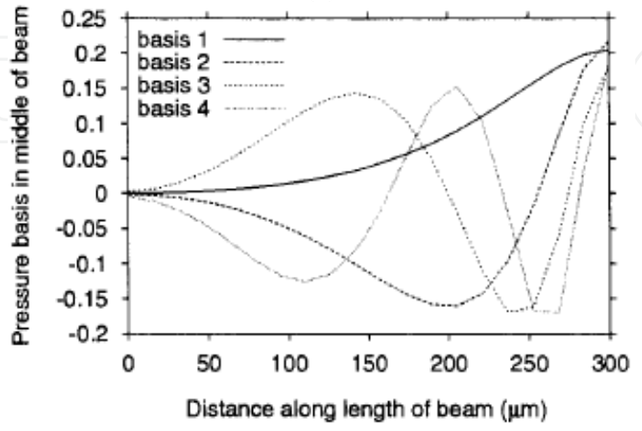
3.1.2 Nonlinear modal ROM

The device governing equations are generally derived from Lagrange equations, after expressing the internal (elastic and kinetic) and external (electrostatic) energy of the system in terms of modal amplitudes and symbolical calculation of the gradients. Assuming that the device undergoes small displacement, the basis chosen results in diagonal mass and stiffness matrices, which can be pre-computed. Linear elastic undamped normal modes of the undeflected device have been often chosen as basic functions to approximate the solution of an electromechanical problem discretized using finite element methods. Modal representation is very efficient since it *requires only one equation per mode* and involved conductor to describe the entire system.
The nonlinear energy terms, instead, are generally expressed as analytical functions of the modal coordinate. In [13] a single static full finite element simulation is used to determine the number of modal functions needed to capture the device behavior and their expected amplitude. Then this information is used to construct the electrostatic energy term. A 3D full model electrostatic simulation is run for values of the modal amplitude that span the operating range of the device, in order to compute the capacitance/deformation curve by using ANSYS' transducer element TRANS126. The results are fitted with a rational fraction of multivariate polynomials using a nonlinear function fitting scheme. In order to model large-displacement behavior of the device, the strain energy is derived by fitting data from a

set of full finite element simulations [13]. A similar procedure is proposed in [17]. In this case, the force-displacement function and the modal strain energy are still derived from a series of FEM simulations, but a polynomial multi-variable fit is used. In order to reduce the complexity of the fitting step, dominant and relevant modes are first characterized. Both the procedures in [13] and [17] can be partially automated and extended to include other conservative energy domains. Other algorithms have also been proposed for the approximation of dissipative energy terms [18, 19]. Dedicated methods have been demonstrated for actuated microbeams, still using the linear undamped mode shapes of the device as basic functions in the Galerkin procedure [20], where two expressions of the nonlinear electrostatic term were proposed as a function of modal coordinates, each including all nonlinearities up to the fifth order, obtained via mathematical manipulation [21]. A new fuzzy-logic model (FLM) for MEMS is presented in [22] in which for reducing the number of data needed for macromodel identification, cluster estimation of a model structure and back propagation method of structure parameters adaptation are chosen to fit the data. As a result the dynamic coupled simulation of a magnetic microactuator takes only several minutes and the force macromodel yielded errors is less than 1.5% for a 5-μm displacement. These FLMs combine fuzzy sets with fuzzy rules that have the capability to model the complex nonlinear behavior.



(a)



(b)

Fig. 8. Basis functions for (a) displacement *z(x, t)* and (b) pressure *p(x, y, t)* [11]

Macromodels obtained via modal basis functions methods have been demonstrated to reproduce results obtained with full physical level simulation with an accuracy of some percentage points and a reduction of the computational complexity simulation. It was shown that even when the problem is mechanically nonlinear, the linear normal modes can serve as basic functions. The approach can be outlined as follows [13]:

1. Compute the linear modes φ_i of the elastic problem.
2. Substitute $u\ (x\ ,y,\ z\ ,t)$ in the governing equation for the deflection (e.g., the Euler–Bernoulli or Timoshenko equations for beams).
3. Obtain a system of n-coupled second-order ordinary differential equations for the $q_i(t)$.
4. Solve the equations to compute the dynamic response either numerically or analytically.

3.1.3 VHDL-AMS export of modal ROM

In general, the equation (5) describes a coordinate transformation of finite element displacement coordinates (mesh element coordinate) to modal coordinates of the macromodel (basic functions or a degree of freedom-d.o.f.):

$$u(x,y,z,t) = u_{eq} + \sum_{i=1}^{n} q_i(t)\varphi_i(x,y,z)\ .$$

The deformation state of the structure given by N nodal displacements u_i ($i=1,2,\ldots,N$) is now represented by a linear combination of n modes weighted by their amplitudes q_j ($j=1,2,\ldots,n$) where $n << N$.

The governing equation of motion describing the ROM of electrostatic actuated MEMS structures in modal coordinates:

$$m_j\ddot{q}_j + 2\xi_j\omega_j m_j\dot{q}_j + \frac{\partial}{\partial q_j}W_{st}(q_1,\ldots,q_n) = \frac{1}{2}\sum_r \frac{\partial}{\partial q_j}C_{ks}(q_1,..,q_n)\cdot(V_k - V_s)^2 + \sum_{i=1}^{n}\varphi_i f_i\ , \qquad (10)$$

where m_j is the modal mass, ω_j is the eigenfrequency, ξ_j the linear modal damping ratio, W_{st} is the modal strain energy function, C_{ks} is the modal capacity-stroke function, r is the number of capacities involved for microsystems with multiple electrodes, V is the electrode voltage applied and f_i is a local force acting at the i-th node. The current I_k at each electrode k is defined by:

$$I_k = \frac{\partial Q_k}{\partial t} = \sum_r (C_{ks}\cdot(\frac{\partial V_k}{\partial t} - \frac{\partial V_s}{\partial t}) + \frac{\partial C_{ks}}{\partial t}\cdot(V_k - V_s)) \qquad (11)$$

An essential prerequisite to establish (10) and (11) are proper modal strain energy and capacity-stroke functions. Both are derived from a series of FEM runs at various deflection states in the operating range. The received data are used for polynomial functions fitting in order to compute the local derivatives, which describe force and stiffness terms. As a matter of fact, shape function methods can be applied to nonlinear systems, too [13]. Geometric nonlinearities, as, for instance, stress stiffening, can be regarded if the modal stiffness is computed from the first derivative of the strain energy function with respect to the modal amplitudes. Capacitance-stroke functions provide non-linear coupling between each eigenmode and the electrical quantities (i.e. electrostatic modal forces, electrical current) if

stroke is understood as modal amplitude. Damping parameters are assigned to each eigenmode.

The first step of the ROM generation is to determine which modes are really significant, and to estimate a proper amplitude range for each mode. Several criteria can be applied, for instance, the lowest eigenmodes of a modal analysis, modes in operating direction, or modes, which contribute to the deflection state at a typical test load. Next the dependencies of the strain energy W_{st} and capacities are described by polynomial functions being fitted. The necessary data points are obtained by imposing each eigenmode with varying amplitude on the mechanical mode11 for the non-linear strain energy and on an electrostatic space model for capacitance. This process is computationally expensive but has to be done just once. The result is a black-box model that can be applied to any load situation. In the concept of the modal superposition method, each eigenmode represents a single independent resonator with modal mass m_i and modal damping ξ_i.

The export of the ROM to VHDL-AMS is performed in two steps [16]. At first, an initialization file containing all necessary information of the macromodel, such as the fitted polynomial coefficients and orders, is generated. Then, the source code in VHDL-AMS is automatically generated. The main problem of exporting the ROM in VHDL-AMS is to express the fitted functions of the non-linear strain energy and of the capacities which are part of coefficients of the differential algebraic equations (DAE), which can be mapped to the simultaneous statements of VHDL-AMS. If simulators support description in matrix notation properly (as MATLAB), the exported VHDL-Models will become more compact and clear.

The Modal ROM approach was implemented as the available ROM-Tool in ANSYS/Multiphysics since Release 7 **(ROM144).** It contains some terminals (fig. 9) and provides necessary functions:

- the master node terminals which describe the displacement u_i and the inserted forces $F_{N,i}$ at these nodes;
- the modal terminals with the modal amplitude q_i and modal force $F_{M,i}$ for the chosen modes;
- the electrical terminals which provide the voltages V_i and currents I_i for the electrodes of the system.

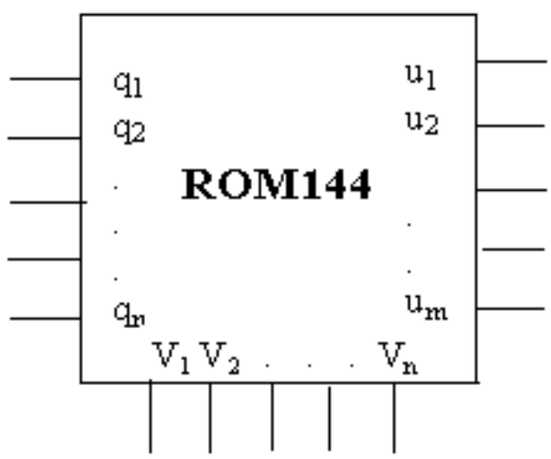


Fig. 9. ROM144 functional block

The ROM144 takes input equations (5),(10),(11) with characteristic constants (modal masses, modal damping ratios and eigenvectors of the master nodes) and provides the special functions of calculating the strain energy W_{mech} (q_i, q_j , q_k) and capacitances C_{op}(q_i, q_j , q_k)as well as their first derivatives with respect to the modal amplitudes q_i, q_j and q_k using the information of the polynomials degrees defined in other packages.
Modal ROM144 approach speeds up computations in 40 times in comparison with the FEM model while pull-in time errors is less than 2%.
Modal ROM approach was implemented also in *INTEGRATOR system of CoventorWare* (http:// www.coventor.com), *MEMS Pro* (http://www.memspo.com) and *MEMSCAP* (http:// www.memsscap.com).

3.1.4 Galerkin's approximated ROM

As in (5) the desired PDE solution $u(x,y,z,t)$ can be approximated by spatially varying arbitral basis functions $\beta_i(x,y,z)$ with time varying coefficients $\alpha_i(t)$:

$$u(x,y,z,t) = \sum_{i=1}^{n} \alpha_i(t)\beta_i(x,y,z) \tag{12}$$

For the Galerkin's method the PDE residual $(L(u)-f)$ is orthogonal to each a_i of the basic functions in the operating range H:

$$(\alpha_i, L(u)-f) = \int \alpha_i^T (L(u)-f)dt = 0, \quad i = 1,n. \tag{13}$$

where L is a differential operator (possibly nonlinear), and f is an input vector.
The basic functions $\beta_i(x,y,z)$ can be chosen arbitrarily, as long as their elements satisfy all of the boundary conditions and are sufficiently differentiable. It means that they can be not only eigenmodes as it was shown before, but they may be Tchebychev, Legendre, Hermit polynomials or even wavelets functions, which were introduced in the past two decades and are gaining increasing popularity. Indeed wavelets have many excellent properties: such as orthogonality, compact support, exact representation of polynomials to a certain degree, and flexibility to represent functions at different resolution levels. The wavelet-Galerkin method is a Galerkin scheme using wavelet functions as the basic functions. However, wavelet functions do not satisfy the boundary conditions. Thus the treatment of general boundary conditions is a major difficulty for the application of the wavelet- Galerkin method. The wavelet interpolation Galerkin method is a Galerkin scheme where basic functions are a class of interpolating functions generated by autocorrelation of the usual compactly supported Daubechies scaling functions [23]. Daubechies' functions are easy to construct. For an even integer L, we have the Daubechies' scaling function $\varphi(x)$ and wavelet $\psi(x)$ satisfying

$$\begin{aligned} \varphi(x) &= \sum_{i=1}^{L-1} p_i \varphi(2x-i) \\ \psi(x) &= \sum_{i=2-L}^{1} (-1)^i p_{1-i} \varphi(2x-i) \end{aligned} \tag{14}$$

The scaling function $\varphi(x)$ is supported in the interval $[0, L-1]$ while the corresponding wavelet $\psi(x)$ is supported in the interval $[1-L/2, L/2]$. The parameter L will be referred to as the degree of the scaling function $\varphi(x)$. The coefficients p_i are called the wavelet filter coefficients and they satisfy the same conditions. The constructed scaling function $\varphi(x)$ and wavelet $\psi(x)$ have also the prescribed properties.
The autocorrelation functions $\theta(x)$, which are used for generating basic functions, can be defined as follows:

$$\theta(x) = \int_{-\infty}^{\infty} \varphi(\tau)\varphi(\tau - x)d\tau \tag{15}$$

and act as the scaling function

$$\theta_{J,k}(x) = \theta(2^J x - k)\,. \tag{16}$$

Wavelets have proven to be an efficient tool of analysis in many fields including the solution of PDE. In [23] a new wavelet interpolation Galerkin method is used for the numerical simulation of MEMS devices under the effect of squeeze film damping. The air film pressure is expressed as a linear combination of a class of basic functions generated by autocorrelation of the usual compactly supported Daubechies scaling functions, which are the first- generation wavelets. The wavelet interpolation Galerkin method was used to predict the frequency response of the accelerometer with a spring mass-damper model with a parallel-plate electrostatic force. Various numerical tests have been conducted by changing the degree of the Daubechies wavelet L and the number J of the scale. Better accuracy can be achieved by increasing L and J. The higher L is, the smoother the scaling function becomes. The price for the high smoothness is that its supporting domain gets larger. The higher J is, the more accurate the solution becomes. The number of differential equations and the CPU time increase significantly as J increases. The solutions for $L = 6$ and $J = 4$ have results higher than the finite difference method.
The present wavelet interpolation Galerkin method is not suitable to solve problems defined on nonrectangular domains, since higher-dimensional wavelets are constructed by employing the tensor product of the one-dimensional wavelets, so their application is restricted to rectangular domains. But usage of the second-generation wavelets which are constructed in the spatial domain can expend in future PDE solutions to complex domains.

3.2 Moment matching based ROM

Moment matching model order reduction is based on the approximation of the original n-dimensional system transfer function $F(s)$ of the original n-dimensional system with a rational function with a lower degree $q<<n$ [24]. This is done by matching some terms of the Taylor expansion of $F(s)$ around a certain expansion point.
For a state space equation

$$\begin{aligned} \dot{X}(t) &= AX(t) + Bv(t) \\ Y(t) &= C^T X(t) \end{aligned} \tag{17}$$

let's perform a Laplace transform to obtain its frequency domain transfer function

$$F(s) = C^T (sI - A)^{-1} B \tag{18}$$

and expand it into Taylor series as

$$F(s) = -\sum_{k=0}^{\infty} m_k s^k \tag{19}$$

where $m_k = C^T A^{-k}(A^{-1}B)$.

Moment matching is directly connected to the Krylov subspace formed by the pair of matrices (A^{-1}, $A^{-1}B$) [24]. The Krylov subspace is spanned by the column vectors in the following collection of matrices:

$$\{A^{-1}B, (A)^{-1}A^{-1}B, \ldots, (A)^{-i}A^{-1}B, \ldots\} \tag{20}$$

where the column vectors are called the Krylov vectors. The q-th order Krylov subspace is denoted by

$$K_q(A^{-1}, A^{-1}B) \tag{21}$$

which is spanned by the leading q linearly independent Krylov vectors in (20). Let $V \in R^{n\times q}$ be any matrix which columns span the Krylov subspace $K_q(A^{-1}, A^{-1}B)$

$$\text{Spancolumn}\{V\} = \text{span}\{A^{-1}B; A^{-2}B; \ldots; A^{-q}B\}.$$

Here V is the orthogonal projection matrix that maps the n-dimensional state-space into a q dimensional state-space and satisfies $V^TV = I$. If the columns of V are orthogonal and B is a column vector, it can be shown that the following identities hold [24]:

$$(A)^iA^{-1}B = V(A_q)^iA_q^{-1}B_q \tag{22}$$

for $i = 0, 1, \ldots, q - 1$. These identities can be used to verify that at least the q leading moments of the full-order and reduced-order transfer functions are matched. Finally, we get the reduced order system of much smaller order (or state-space dimension) by performing variable change $x(t) = V\tilde{x}(t)$ and multiplying on V^T both sides of the equations (17):

$$\begin{aligned} \tilde{x}'(t) &= A_q\tilde{x}(t) + B_q v(t) \\ y(t) &= C_q{}^T\tilde{x}(t) \end{aligned} \tag{23}$$

where $A_q = (V^TAV)$, $B_q = V^TB$, $C_q = C^TV$.

As a common practice, the block vectors forming the Krylov subspace are orthogonalized by using the Arnoldi algorithm for a numerical stability. Lets describe the block Arnoldi algorithm for a single column input matrix, i.e., B = b, where $b \in R^n$, so that the algebraic operations involved can be seen.

3.2.1 Arnoldi Algorithm

i. LU factorize matrix A: $A = LU$.

ii. Solve $\tilde{v}_1$ from: $A\,\tilde{v}_1 = b$.

iii. Compute $h_{11} = \|\tilde{v}_1\|$ and $v_1 = \tilde{v}_1 / h_{11}$.

iv. For j = 2, . . . , q:

Solve $\tilde{v}_j$ from: $A\tilde{v}_j = Av_{j-1}$.

For $i = 1, \ldots, j-1$: $h_{ij} = v_i^T \tilde{v}_j$.

$$w_j = \tilde{v}_j - \sum_{i=1}^{j-1} v_i h_{ij}$$

$h_{jj} = \|w_j\|$, $v_j = w_j / h_{jj}$.

Note that the Arnoldi algorithm terminates when $h_{jj} = 0$, which means that the subsequent vectors belong to the subspace already generated. The Arnoldi algorithm is basically a Gram–Schmidt procedure for orthogonalizing the Krylov vectors. The variant of the Arnoldi algorithm (also known as *PRIMA*) with some extra computational effort preserves the passivity of the original system [27].

The Moment matching method of model order reduction can be extended on nonlinear systems. In these cases the original nonlinear systems has to be changed previously (linearized or piecewise-linearized, approximated by a quadratic systems, divided into several linear systems, etc.) [28].

3.2.2 Second order systems

A size of a second order equations system (2) can be also reduced by transforming it to the first order system (1), and then applying the methods described before. However, the reduction of second order systems by such transformation ignores the physical meaning of the original matrices and gives a reduced order model in a first order form. It is desirable for the reduced system to preserve the form of the original system (2). Approaches, that deal directly with the system (2) reduction have been proposed in the framework of Krylov subspaces methods [29, 30].

The transfer function for the system (2), with zero initial conditions, is given by:

$$H(s) = C^T (s^2 M + sD + K)^{-1} B \tag{24}$$

If the system is **undamped**, i.e. $D = 0$, the Arnoldi process can be applied for the computation of a basis for the Krylov subspace $K_q\ (K^{-1}M, K^{-1}B)$ which is used for the projection matrix V building. The transfer function can be expanded into Taylor series as

$$H(s) = -\sum_{k=0}^{\infty} m_k s^{2k} ,$$

where $m_k = C^T (K^{-1}M)^{-k} (A^{-1}B)$.

Then the reduced system matrices with variable change $x(t) = V\tilde{x}(t)$ are obtained by the orthogonal projection:

$$M_q = V^T M V,\ K_q = V^T K V,\ B_q = V^T B,\ C_q = C^T V \tag{25}$$

and the matching properties of the method are conserved [31].

It is worth noting that, starting from a second order system in the form (2), Krylov subspace methods require the knowledge of K^{-1}. For high dimensional systems, the explicit calculation of the inverse of K is computationally not affordable. Its computation is therefore

replaced by the solution of linear systems of equations through a *LU*- decomposition and defying $K^{-1} = U^{-1} L^{-1}$.

If the system is **damped**, i.e. $D \neq 0$, in case of Rayleigh damping, it demonstrates that the damping matrix can be neglected during the reduction process and it can be computed afterwards, as a linear combination of the reduced stiffness and mass matrices [29]. That is why it is possible to recalculate the matrix D also similar to matrices M and K as in (25):

$$D_q = V^T D V.$$

The reduction of a nonlinear system (2) can be done by using linear model order reduction techniques and considering nonlinearities as inputs. In the general case, the stiffness matrix is nonlinear since its entries dependent on the nodal displacements x. So damping matrix entries also are nonlinear functions of the nodal displacements x and the applied voltages V. If the nonlinearities are confined in the input function f, the system can be reduced using linear model order reduction technique. The only complication is that, after the reduction, the argument of x has to be recovered by the projection $x = Vx_q$ [31,32].

As a typical example Fig. 10 displays the transient simulation and frequency response of the original and reduced models of the microgyroscope being received via usage of the Arnoldi process [33].

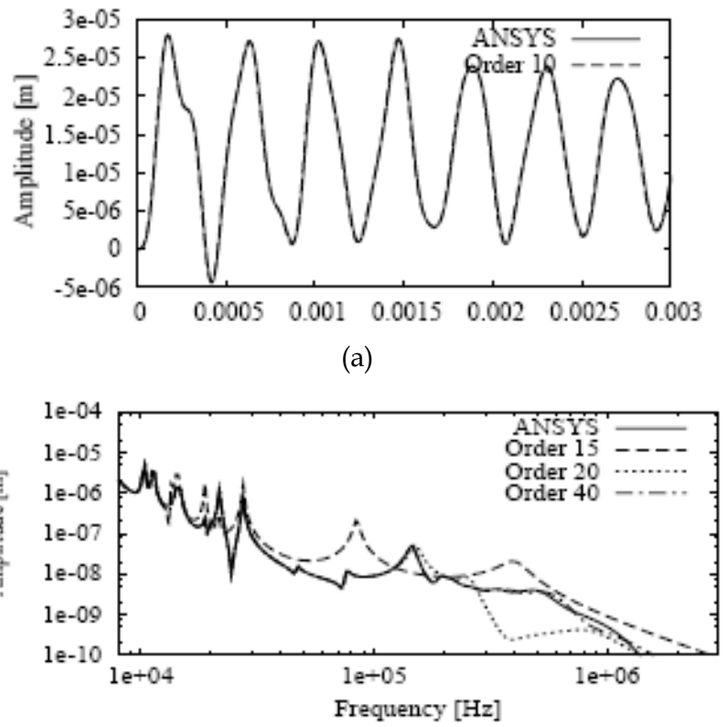


Fig. 10. Comparison of transient behavior (a) and transfer functions (b) for the full and reduced microgyroscope models [33].

We see that the solution obtained by reduced model of order 10 is already very close to the true ANSYS excitation (fig.10,a) while the reduced models of order 15 up to 20 show

considerable deviations at the high frequency range (fig.10,b). The model with order of 40 shows a perfect match for the lower eigenfrequencies and it is quite closer for higher frequencies, though this is not so important for the gyroscope.
Using Krylov/Arnoldi approach, only a postprocessor to ANSYS is necessary to generate a macromodel in one of the well established model description languages: pure C code, HDL-A, MAST, Modelica and the new standardized VHDL-AMS which are supported by powerful system simulators. Such approach was implemented in *mor4ansys* (pronounced "more for ANSYS") that was developed by IMTEK [33] and which provides the reduced model of order 20 - 30 with an accuracy of a few percents when the dimension of an original FEM is up to 100 000.

3.3 Equivalent-circuit ROM

Taking in to account relations between displacements x, velocities v and accelerations a: $a = dv/dt, \quad x = \int v dt$, it is possible to present the equation (2) in the form [34,35]:

$$\frac{d}{dt}(Mv) + Dv + \int Kv dt = F(t) \quad \text{or} \quad \tilde{C}\dot{v} + \tilde{G}v + \tilde{L}v = F(t) \ , \tag{26}$$

where $\tilde{C} = M, \tilde{G} = D, \tilde{L} = K$ are equivalent matrices of capacitances, conductance and inductances.

$$\tilde{C}, \tilde{G}, \tilde{L}$$

The elements of matrices $\tilde{C}, \tilde{G}, \tilde{L}$ are formed from the elements of the mass, damping and stiffness matrices in the following ways:

$$C_{ij} = -m_{ij}, \quad i,j = 1(1)N, \quad i \neq j; \quad C_{ii} = \sum_{j=1}^{N} m_{ij}, \quad i = 1(1)N; \quad L_{ij} = -1/k_{ij}, \quad i,j = 1(1)N, \quad i \neq j;$$

$$L_{ii} = 1/\sum_{j=1}^{N} k_{ij}, \quad i = 1(1)N; \quad G_{ij} = -d_{ij}, \quad i,j = 1(1)N, \quad i \neq j; \quad G_{ii} = \sum_{j=1}^{N} d_{ij}, i = 1(1)N. \tag{27}$$

where N is a number of equations or nodes of the MEMS structure.
In this approach a capacitive-inductive-resistive model of the circuit is built which correctly reflects mass, damping and stiffness matrices. Nodal potentials in this model correspond to the displacement velocities v.
Connected inductors, capacitors and conductors are placed in parallel between each two nodes and each node and ground (for the case i = j), fig.11. Values of these elements are defined by equations (27). Similar approach was used for solving thermo-structural analysis [35].
Displacements x are defined by the current of inductances L_{ii}, which are connected between a node and ground. Such approach suggests significant advantage if the coefficients of the mass and stiffness matrices become time-dependant.
The task of reduction of MEMS model order turns now into reduction of the equivalent RLC circuit size. There are two ways of performing such circuit size decreasing with the minimal accuracy loss:

- sequential excluding of the internal circuit nodes by application of Y-Δ (star-triangle) transformation [36, 39];
- building a macromodel as a four- terminal [40].

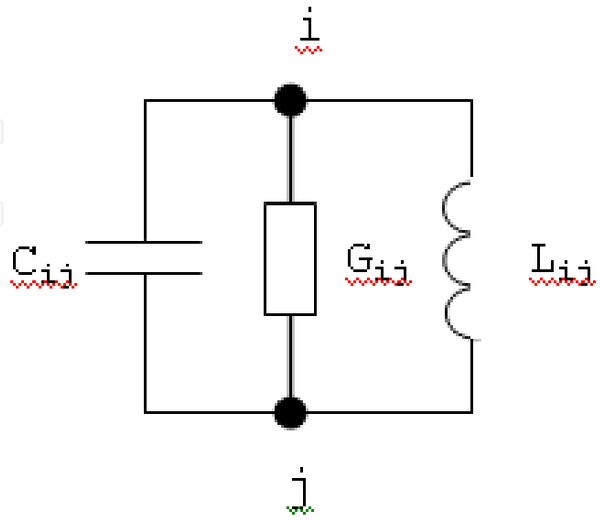


Fig. 11. Circuit element for composing the equivalent-circuit model

3.3.1 Y-Δ transformation based circuit size reduction methods

The essence of the methods based on the Y-Δ transformation consists in the following. Let's *i*-th node and *k* its neighbors are located as shown on fig.11. Then the component equation of *i*-th row will look like

$$Y_i V_i - y_1 V_1 - y_2 V_2 - \ldots - y_n V_n = 0 \ , \tag{28}$$

where $Y_i = \sum_{j=1}^{k} y_j$.

Let's define V_i as

$$V_i = \left(\sum_{j=1}^{k} y_j V_j \right) \Big/ Y_i \tag{29}$$

and replace V_i in other *k* equations , that is equivalent to excluding *i*- th node from a circuit. Then, for example, the equation of the first node transfers to:

$$(\bar{Y}_1 + y_1 - y_1^2 / Y_i) V_1 - \left(\sum_{j=2}^{k} y_1 y_j V_j \right) / Y_i \ - \sum_{\substack{r=1 \\ r \neq i}}^{k1} y_r V_r = 0 \tag{30}$$

where $\bar{Y}_1 = \sum_{\substack{r=1 \\ r \neq i}}^{k1} y_r$ is a sum of all node conductance excluding *i*- th node, *k1* is a number of nodes being connected to node 1.

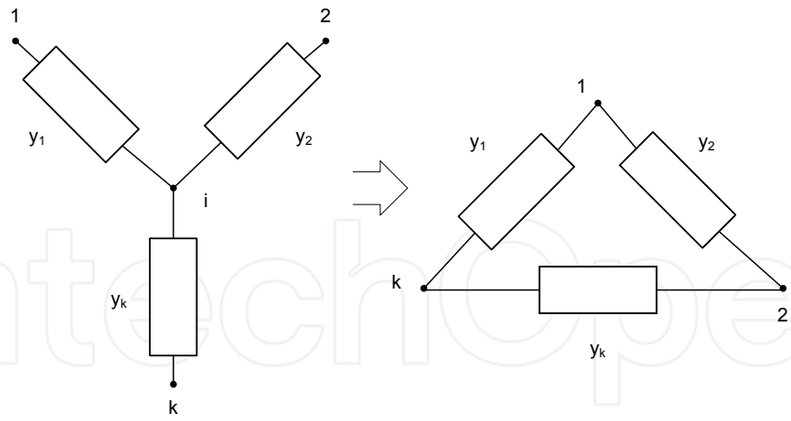


Fig. 12. A working node of the RLC-circuit: conductance are added between node 1 and all neighbors nodes

The equation (30) can be simplified:

$$(\bar{Y}_1 + \left(\sum_{j=2}^{k} y_1 y_j\right) / Y_i) V_1 - \left(\sum_{j=2}^{k} y_1 y_j V_j\right) / Y_i - \sum_{\substack{r=1 \\ r \neq i}}^{k1} y_r V_r = 0 \tag{31}$$

Note that it is equivalent to adding k-*1* new elements between first node and *k-1* former neighbors of *i*-th node on Fig. 12.
For any two *a*-th and *b*-th nodes, which are neighbors to *i*- th node, elimination of *i*- th node will add a new element between these two nodes, which is equal to:

$$y_{ab} = (y_a y_b) / Y_i \tag{32}$$

or in the p-polynomial form, taking into account existence of R, L and C elements, as shown on fig. 11:

$$y_{ab} = \left(g_a + \frac{b_a}{p} + pc_a\right)\left(g_b + \frac{b_b}{p} + pc_b\right) \Big/ \left(G_i + \frac{B_i}{p} + pC_i\right) \tag{33}$$

where $C_i = \sum_{j=1}^{k} C_j$, $B_i = \sum_{j=1}^{k} B_j$, $G_i = \sum_{j=1}^{k} G_j$ are sums of all the capacitances, reciprocal inductances and conductance connected to *i*-th node.
In order to simplify (33) two constant time values $\tau_{RCi} = C_i / G_i$ and $\tau_{LCi} = \sqrt{C_i / B_i}$ are introduced for any node in the circuit. The time constant of *i*-th node is defined as $\tau_i = \max(\tau_{RC}, \tau_{LC})$ and it is considered to be fast if

1. $\tau_i < \tau_{\min} = 2\pi / \omega_{\max}$,

where $\tau_{\min}$ - a time constant which depends on maximal circuit frequency $\omega_{\max}$ being defined by user. So, a fast node will satisfy the following conditions:

$$\omega_{\max} C_i < G_i,\ G_i < B_i / \omega_{\max} \text{ i } \omega_{\max} C_i < B_i / \omega_{\max}.$$

In order to eliminate a fast node from *RLC* circuit, let us consider the following two cases.

2. If $\tau_{RCi} >> \tau_{LCi}$, the equation (33) can be transformed into:

$$y_{ab} = (g_a + pc_a)(g_b + pc_b) / (G_I + pC_i) \tag{34}$$

and its decomposition into Taylor series will look like:

$$y_{ab} = \frac{g_a g_b}{G_i} + p\frac{c_a g_b + c_b g_a}{G_i} + p^2 \frac{4c_a c_b}{G_i} . \tag{35}$$

For typical *R,L,C* values (*R*=0,1÷1000 ohm, *L*=0,01÷10 nH, *C*=0,001÷10 pF, ω=0,1÷10 GHz) the contribution of the last term in (35) will be significantly smaller that previous ones. The constant term in (35) gives the value of the conductance, which appears between *a*- th and *b*- th nodes during the elimination of *i*- th node, and the second term in (35) defines the value of capacitance.

3. If $\tau_{LCi} >> \tau_{RCi}$, the equation (33) is transformed into:

$$y_{ab} = \frac{1}{p}\frac{b_a b_b}{B_i} + p\frac{c_a b_b + c_a b_b}{B_i} . \tag{36}$$

The first term with $1/p$ in (36) defines the value of the reactive conductance (the inductances' reciprocal value), which appears between *a*- th and *b*- th nodes during the elimination of *i*- th node, and the second term in (36) gives the value of the capacitance.

Final formulas for additional elements between *a*-th and *b*-th nodes during the elimination of *i*- th node for all possible cases are given in tables 1 and 2. All these formulas can be deducted from (35) and (36). The only exception is the case when *a*- th and *b*- th nodes are connected to *i*- th node through a capacitor when $C_{ab} = C_a C_b / C_i$.

In order to eliminate a node from equivalent MEMS circuit with the smallest inaccuracy, the following two criteria should be taken into account:

- a node should be fast;
- time constants τ_{LCi}, τ_{RCi} should not be compared (have almost the same values).

In practice, usually lower eigenfrequencies for mechanical systems are most interesting. Therefore, a compromise between accuracy and size of the received circuit models can be reached by proper selection of $\tau_{\min}$ value. So the algorithm of equivalent-circuit ROM can be described as following:

1. Time constants should be calculated for all the nodes of the circuit.
2. All the nodes should be put into a priority queue, sorted by time constants, apart from input, output and ground nodes.
3. Take the first i- th node in the queue (the one with the smallest time constant $\tau_i < \tau_{\min}$).
4. Find the neighbors *k* of *i*- th node.
5. Eliminated *i*- th node according to the rules described in tables 1, 2 (depending on τ_{LCi}, τ_{RCi} values).
6. Time constants in the set *k* (which consists of the neighbor of i-th node) should be recalculated
7. Take i- th node from the head of the priority queue and jump to step 3.

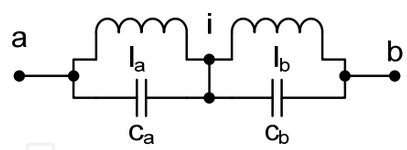
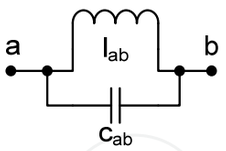
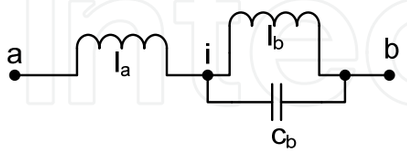
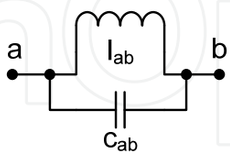
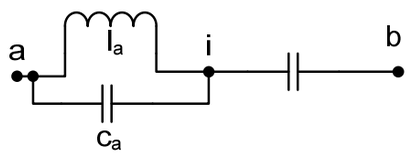
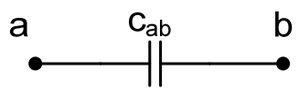
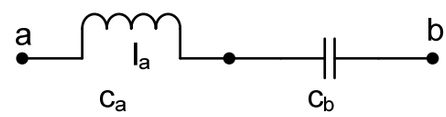
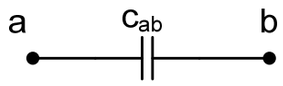
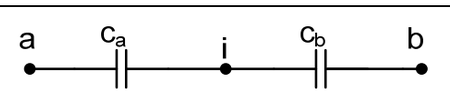
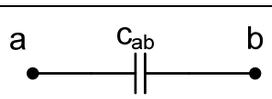
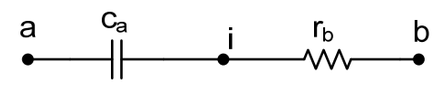
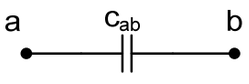
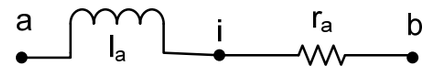
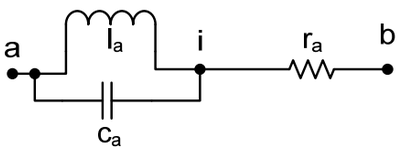
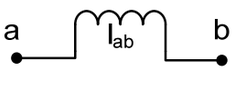
Original branch	Substitution	Formula
		$b_{ab}=b_a b_b/B_i$ $C_{ab}=(b_a C_b++b_b C_a)/B_i$
		$b_{ab} = b_a b_b/B_i$ $C_{ab} = b_a C_b/B_i$
		$b_{ab}=b_a b_b/B_i$
		$C_{ab} = b_a C_b/B_i$
		$C_{ab}=b_a*C_b/B_i$
		$C_{ab}=C_a C_b/C$
		$C_{ab}=C_a g_b/G_i$
		$g_{ab}=g_a g_b/G_i$
		$g_{ab}=b_a g_b/G_i$
		$g_{ab}=b_a g_b/G_i$

Table 1. Transformations for $\tau_{LCi} >> \tau_{RCi}$ case

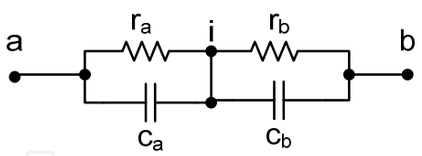
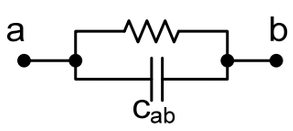
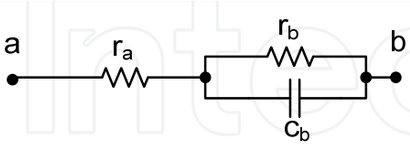
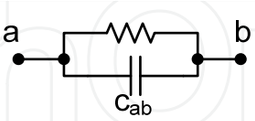
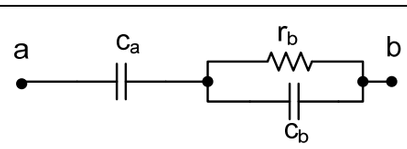
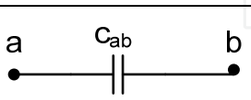
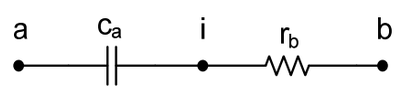
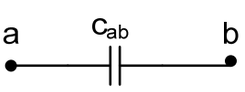
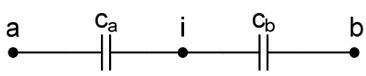
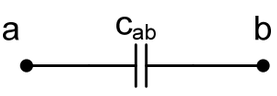
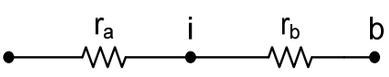
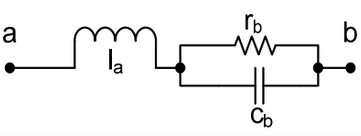
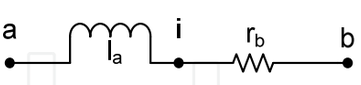
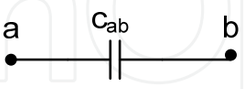
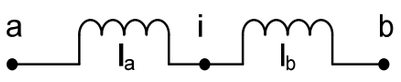
Original branch	Substitution	Formula
		$g_{ab}=g_a g_b/G_i$ $C_{ab}=(g_a C_b+{}_b C_a)/G_i$
		$g_{ab}=g_a g_b/G_i$ $C_{ab}=g_a C_b/G_i$
		$C_{ab}=g_b C_a/G_i$
		$C_{ab}=g_b C_a/G_i$
		$C_{ab}=C_a C_b/C$
		$g_{ab}=g_a g_b/G_i$
		$g_{ab}=b_a g_b/B_i$
		$g_{ab}=b_a g_b/B_i$
		$g_{ab}=b_a C_b/B_i$
		$g_{ab}=b_a b_b/B_i$

Table 2. Transformations for $\tau_{RCi} >> \tau_{LCi}$ case

3.3.2 Building MEMS macromodel as a four- terminal

The main idea of this method is to develop a MEMS macromodel as a four-terminal circuit (n-terminal in a general case) as:

$$\begin{bmatrix} I_a \\ I_b \end{bmatrix} = \begin{bmatrix} Y_{aa} & Y_{ab} \\ Y_{ba} & Y_{bb} \end{bmatrix} \begin{bmatrix} U_a \\ U_b \end{bmatrix}, \tag{37}$$

where I_a, U_a are current and voltage at the macromodel input, I_b, U_b are current and voltage at the output. It is suggested to obtain equation (37) directly from the general matrix of the circuit (e.g. admittance matrix Y) according to the expression [40]:

$$Y_{tp} = \frac{1}{\Delta_{aa,bb}} \begin{bmatrix} \Delta_{bb} & -\Delta_{ba} \\ \Delta_{ab} & \Delta_{aa} \end{bmatrix} \tag{38}$$

where Δ_{ij} – an algebraic complement to a_{ij} element of the initial circuit admittance matrix Y, which is equal to a determinant of the matrix being obtained from the matrix Y after eliminating i – th row and j- th column on the crossing of which the given element a_{ij} is situated. In addition the sign of Δ_{ij} is defined by the factor $(-1)^{i+j}$;

$\Delta_{aa,bb}$ – redoubled algebraic complement which equals to the determinant of the matrix obtained by eliminating a-th,b-th rows and a-th ,b-th columns and its sign is defined by the factor $(-1)^{a+a+b+b}$.

The necessary algebraic complements can be calculated by the initial matrix inversion procedure, if matrix elements are defined for the selected frequency ω_0, which is fixed by an user:

$$Y^{-1} = \frac{1}{\Delta} \begin{bmatrix} \Delta_{11} & \Delta_{21} \cdots & \Delta_{n1} \\ \Delta_{12} & \Delta_{22} \cdots & \Delta_{n2} \\ & & \\ \Delta_{1n} & \Delta_{2n} \cdots & \Delta_{nn} \end{bmatrix},$$

where Δ - is the determinant of the initial matrix Y and $\Delta_{aa,bb} = \frac{\Delta_{bb}\Delta_{aa} - \Delta_{ba}\Delta_{ab}}{\Delta}$.

So if numerical values of the inverted Y^{-1} matrix elements are computed:

$$Y^{-1} = \begin{bmatrix} g_{11} & g_{21} \cdots & g_{n1} \\ g_{12} & g_{22} \cdots & g_{n2} \\ & & \\ g_{1n} & g_{2n} \cdots & g_{nn} \end{bmatrix} \tag{39}$$

it is possible to find parameters of the equivalent four- terminal (38). By choosing elements $g_{aa}, g_{bb}, g_{ab}, g_{ba}$ of the inverted matrix (39) and computing $g_{aa,bb} = g_{aa}g_{bb} - g_{ab}g_{ba}$ it is possible to find parameters of the reduced model (37) from the relations:

$$\begin{aligned} Y_{aa} &= g_{bb} / g_{aa,bb}; \\ Y_{ab} &= -Y_{ba} = g_{ba} / g_{aa,bb}; \\ Y_{bb} &= g_{aa} / g_{aa,bb}. \end{aligned} \tag{40}$$

Since macromodel parameters are determined by sum of the real and imaginary parts:

$$Y_{ij}= a_0+i\, a_1 , \tag{41}$$

it is convenient to represent the circuit macromodel also as a sum of real and imaginary terms:

$$Y_{tp} = \begin{bmatrix} Y_{aa0} Y_{ab0} \\ Y_{ba0} Y_{bb0} \end{bmatrix} + i \begin{bmatrix} Y_{aai} Y_{abi} \\ Y_{bai} Y_{bbi} \end{bmatrix}. \tag{42}$$

Investigation the expression (42) for different frequencies confirms that for linear circuits at least for frequency range $\omega \le \omega_0$, where ω_0 – frequency at which elements of the inverted matrix (40) are initially defined, real terms of the computed parameters (42) preserve their values and imaginary terms change proportionally to the selected frequency. This confirms the possibility to use macromodel parameters being computed at once frequency in a broad frequency range.

As it was noted earlier, the macromodel form (42) is inconvenient for direct implementation in the circuit simulation software. A real part of Y_{ij} parameters for the stable passive macromodels ($a_0>0$) has to be positive while an imaginary part could be negative. If in (41) $a_o > 0$, $a_1 > 0$ the Y_{ij} component of the four- terminal may be presented by parallel connection of conductance, capacitance and inductance [41]:

$$G_{ij} = \mathrm{Re}\,(Y_{i,j}),\ C_{ij} = k\,\mathrm{Im}\,(Y_{ij})/\ \omega_0,\ L_{ij} = 1/[(k-1)\mathrm{Im}\,(Y_{ij})\ \omega_0]. \tag{43}$$

If $a_o > 0$, $a_1 < 0$ the respective component Y_{ij} may be presented in the same way as parallel connection of conductance, capacitance and inductance but:

$$G_{ij} = \mathrm{Re}\,(Y_{i,j})\ ,\ C_{ij} = (k-1)\ \mathrm{Im}\,(Y_{ij})/\ \omega_0 \ \text{ and } \ L_{ij} = 1/(k\ \mathrm{Im}\,(Y_{ij})\ \omega_0), \tag{44}$$

where *k* is defined by reactive components values ratio.

So it is possible to select priori the equivalent-circuit macromodel, for example, shown on Fig.13, and to define parameters of its components using expressions (43) and (44).

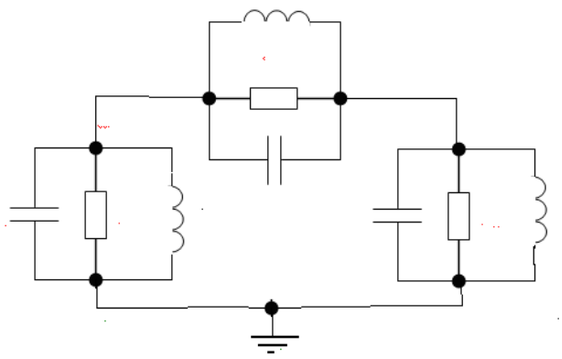


Fig. 13. Equivalent-circuit macromodel of a four-terminal

The important feature of the equivalent circuits MEMS macromodel is a possibility to use optimization procedures for accurate adjustment of the macromodel components values to meet the device characteristics being obtained by FEM model. If a whole MEMS equivalent circuit is rather large it is possible to divide it into some subcircuits and apply transformation (37) to each subcircuit.

Then to combine individual four-terminal circuits into one equivalent MEMS four-terminal circuit taking in account ways of connectivity of different subcircuits (parallel, sequence or mixed) and proper recalculating four-terminal circuits parameters systems (*y, z, h, a*) [40].

3.3.3 NetALLTED equivalent-circuit ROM subsystem

Equivalent-circuit ROM for MEMS was implemented in the circuit simulation package NetALLTED (**ALL TE**chnologies **D**esinger) which was developed not only for simulation and analysis, but for processing project procedures such as parametric optimization tasks; optimal tolerance assignments; centering availability regions; yield maximization [42]. NetALLTED is widely used for design of Nonlinear Dynamic Systems composed of either/and electronic, hydraulic, pneumatic, mechanical, electromagnetic, and other elements and it is available through the Internet (http://allted.kpi.ua/). ROM developing approach in hand provides more than 99% reduction of elements and node numbers of equivalent- circuit ROM. For example, for the accelerometer only 3 nodes and 6 elements are left from initial 1,883 nodes and 62,826 elements.

Let's consider the example of the equivalent- circuit ROM for the beam working on bending and find its eigenfrequencies [34]. The left end of the beam is fixed motionlessly, right one is free. Force *f* is applied to the right end perpendicularly to the beam axe.

The initial FEM model was constructed using ANSYS Multiphysics v.10.0 and the beam ANSYS library's BEAM3 finite element with the length of 0.5 um for following beam parameters: *L* = 25 um; beam cross-section is a square one with height of 3 um and width of 2 um. Beam material properties: coefficient of elasticity $E=2 \cdot 10^{11}$, Pa = 0.2 N/mkm^2; Poisson coefficient μ = 0.3; material density $\rho = 6 \cdot 10^3$ kg/m^3 = $6 \cdot 10^{-9}$ mg/mkm^3. The initial equivalent beam circuit contains 101 nodes and 314 elements.

The developed equivalent-circuit ROM with 5 nodes and 14 elements is presented on fig.14.

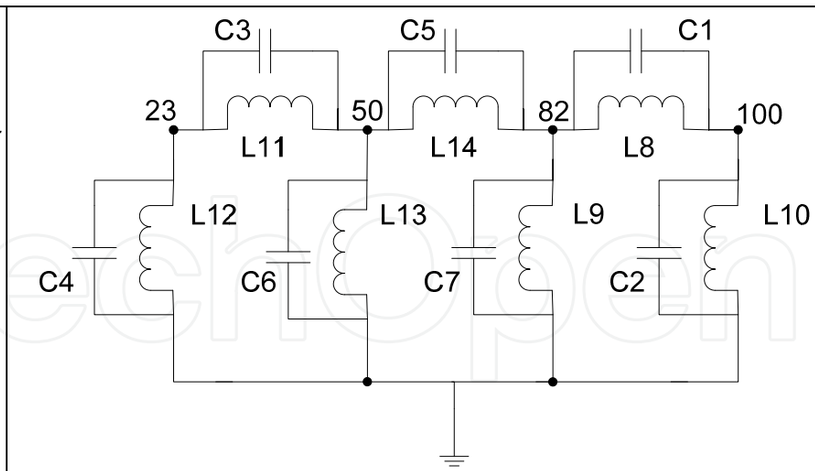
Object Circuit Beam; J1(100,0)=-100; C_1(82,100) = -0.116667; L_10(100,0) = -34161.8; L_11(23,50) = 1.35e-10; L_12(23,0) = 1.15e-10; L_13(50,0) = -16435.9; L_14(50,82) = 1.6e-10; C_2(100,0) = 6.3; C_3(23,50) = -0.116667; C_4(23,0) = 17.3833; C_5(50,82) = -0.116667; C_6(0,50) = 20.65; C_7(0,82) = 17.5; L_8(82,100) = 9e-11; L_9(82,0) = -580750; &&	

Fig. 14. Equivalent beam circuit

The two lower eigenfrequencies of the beam are defined by computation. The results of ANSYS Multiphysics frequency analysis as well as the results of the equivalent- circuit ROM simulation for different τ_{min} by NetALLTED [36,43] are given in table 3.
It is obvious that the accuracy of a macromodel obtained with $\tau_{min} = 3*10^{-5}$ is rather high and there are only 5 nodes and 14 elements in the reduced circuit. For more accurate simulation it is possible either to use a reduced circuit obtained with smaller values of τ_{min} (when a size of equivalent circuit ROM increases), or to adapt the received ROM with help of the optimization methods.

	ANSYS results	ALLTED results				
		Source circuit	Reduced circuit			Optim. circuit
τ_{min} , s	-	-	$5*10^{-6}$	10^{-5}	$3*10^{-5}$	$3*10^{-5}$
Node number	-	101	24	12	5	5
Element number	-	314	76	38	14	14
Reduction by nodes, %	-	-	76,2376	88,1188	95,0495	95,0495
Reduction by elements, %	-	-	75,7962	87,8981	95,5414	95,5414
1st peak, Hz	1336,2	1336,3	1336,1	1334,9	1327	1336,2
2nd peak, Hz	4009,3	4009,3	4009,4	3993,3	3612,1	4009,4
Maximal error, %	-	-	0,01	0,3	9,9	0,003

Table 3. Frequency beam analysis results

The macromodel of the four-terminal for the same beam is being developed in according with (38)-(44) with only 3 nodes and 6 elements that presented on fig.15.

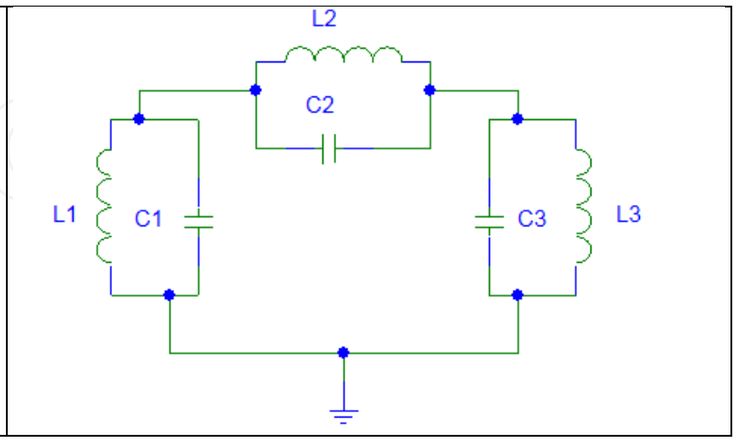
Object Circuit Beam; J1(1,0)=-100; L2(1,2) = 2.997781327698e-011; C2(1,2) = 3.155049077722e+002; L1(1,0) = 1.125969906329e-009; C1(1,0) = 1.890000000000e+001; L3(2,0) = 7.750537495099e-010; C3(2,0) = 1.220321457595e+001; &&	

Fig. 15. Equivalent beam circuit being considered as a four- terminal

The results of simulations of the four- terminal equivalent circuit by NetALLTED [36,43] are given in table 4.

	Source circuit	Reduced circuit (as n-ports)
Node number	101	3
Element number	314	6
1st peak, Hz	1336.3	1337.0(0.05%)
2nd peak, Hz	4009.3	4008.9(0.01%)

Table 4. Results of frequency analysis of the four- terminal equivalent circuit for a beam

The advantage of the equivalent circuit approach is obtaining small size of the equivalent reduced circuit as well as the possibility to get required frequencies with a high accuracy, using NetALLTED optimization possibilities (fig.16). The disadvantage is a necessity to make the additional analysis of reduced circuit in order to find the most sensitive elements and take them as variable parameters during optimization procedure.

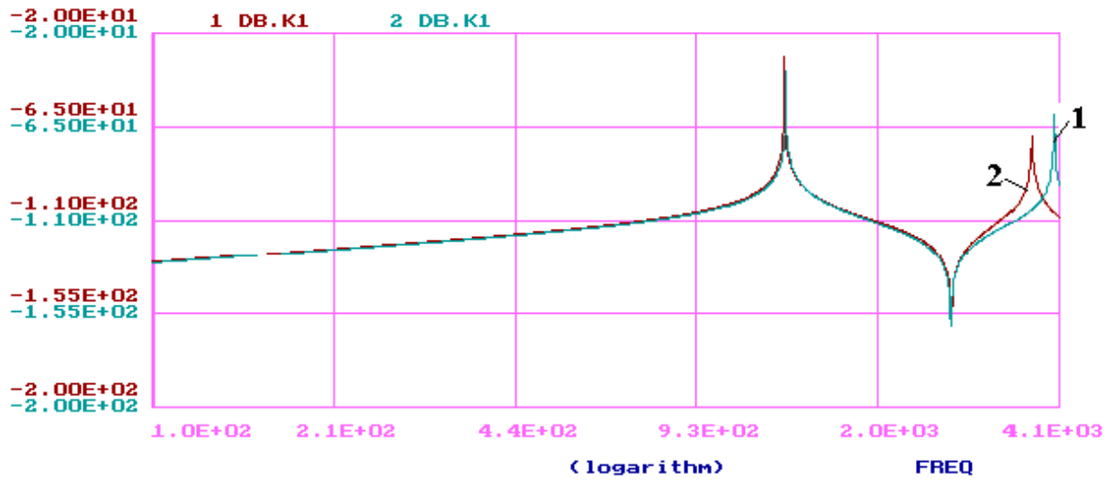


Fig. 16. ROM transfer function before (1) and after (2) optimization.

For example, for reduced circuit, shown at fig.14, four variable elements have been chosen (*L8, L11, L12, L14*) and the objective function *OF ERROR1= F8(1336.3,4009.3/T1,T2)* was constructed which contains the requirement to obtain resonance peaks at frequencies *T1=1336.3* Hz and *T2=4009.3* Hz in according to the ANSYS analysis results (table 3). The objective function contains also current values of frequency response extremes *T1=MAXA(db.K1,100,1600)* and *T2=MAXA(db.K1,1700,4100),* which are calculated with help of MAXA directive for defining time or frequency when the output (*db.K1* in our case) reaches its maximum value in the specified time or frequency range (*100-1600* Hz). Among available 12 optimization methods being incorporated in NetALLED the Random Search Method (METHOD=40) with search interval reducing has been used to optimize the beam macromodel parameters.

4. MEMS coupled system-level model

The modeling of MEMS provides a very challenging task in modern engineering. This field of research is inherently multiphysics of nature, since different physical phenomena are

tightly intertwined at microscale. Typically, up to four different physical domains are usually considered in the analysis of microsystems: *mechanical, electrical, thermal* and *fluidic*. For each of these separate domains, well-established reduced order modeling and analysis techniques are available. However, one of the main challenges in the field of microsystems engineering is to connect models for the behavior of the device in each of these domains to equivalent lumped or reduced-order models without making unacceptably inaccurate assumptions and simplifications and to couple these domains correctly and efficiently.

Micromechanical membrane devices (capacitive pressure transducers, ultrasonic transducers), surface micro machined devices (RF switches, micro optical devices) as well as bulk micro machined devices (accelerometers, inclinometers, laser scanning mirrors) are driven or sensed by nearly parallel electrode pairs in many cases. The motion of these electrodes is strictly normal to the surfaces.

It means that MEMS electrical parts of these MEMS had to be combined with mechanical ones (fig.16). Typically, block-diagram descriptions and lumped-element circuit models for components are connected into a full system. Mostly, this description is used for functional analysis of a design concept.

The coefficients and electrostatic nodal force are obtained from the capacitance-displacement function *C(w)* of the associated electrode portions and gap space. The function can be input by one of three means [44]:

- as analytical function if the electrode portions make up a plate capacitor geometry with homogeneous intermediate field;
- as polynomial approximation of a function given by data points;
- as data table wherein the element subroutine interpolates values during solution.

The electrostatic forces acting on the movable conductor of the device are included in the model as nonlinear input forces, which are applied on nodes distributed over the conductor surface. Nodes divide the surface into N smaller portions. A lumped force is applied to k- th node at the center of each portion, in its preferential direction of movement x_i. The capacitance C_k between the k-th portion and the fixed electrode of the device is computed, for its undeformed configuration, using an electrostatic analysis

The entity of each force f_k is then approximated as:

$$f_k = \frac{1}{2}\frac{\varepsilon A_k}{(d_0 + w_i^k)^2}(V_p - V_n)^2 \tag{45}$$

where d_0 is the initial distance between the conductor and the electrode, ε is the relative dielectric permittivity, w_i^k is the displacement of the k- th node along the direction x_i and $(V_p\text{-}V_n)$ is the voltage difference between the structure and the fixed electrode.

The capacitance C_k can be calculated from

$$C_k = \varepsilon A_k/(d - w_i^k) \, . \tag{46}$$

If C_k varies considerably with the deformation of the structure, then a series of electrostatic computations for different device deflections in its operation range can be performed. The results can be used for extracting the dependency $C_k(w_i^k)$ and calculating the electrostatic force.

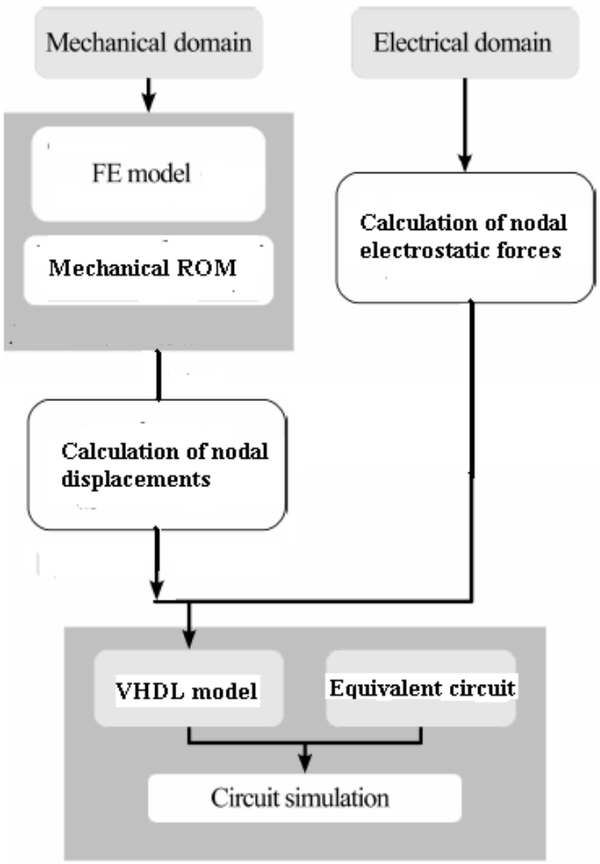


Fig. 16. MEMS system-level simulation approach [33]

According to the assumption that conductors are equipotential, all the nodes connected to a certain conductor are subjected to the same voltage boundary conditions. The total current flowing in the conductor is simply given by the sum of the currents at those nodes:

$$i_k = \frac{d}{dt}[C_k(V_k - V_n)].$$

The nodes used for electrostatic force application can be also used for monitoring the distance between the movable structure and the electrode. When this is equal to the transduction gap, the contact condition is reached and contact forces with the stiffness of the contact K_n , given by equation

$$F_{cont} = K_n(d - gap_{min}), \tag{47}$$

can be applied to nodes.

Computation of the electrostatic forces adds some complexity to the development of the MEMS system-level model, but this is largely compensated by the speed-up simulation of the full model.

There is the special ANSYS' transducer element *TRANS126* which has the possibility to calculate the capacitance of a parallel plate capacitor model (at particular nodes or as a whole) as [47]

$$C(w) = \frac{C_0 d_0}{d_0 - w} = \frac{C_0}{d_0}(1 + \frac{w}{d_0} + \frac{w^2}{d_0^2} + \frac{w^3}{d_0^3} + \frac{w^4}{d_0^4} + ...) \tag{48}$$

where d_o and w are the initial distance and the displacement between the plate.
The element has two nodes; the gap distance is calculated as the sum of the initial displacement and the difference of the nodal displacements in the direction of the element. The force is calculated by equation being similar to (44) so that in the constant voltage case

$$F = \frac{1}{2}\frac{\partial C(w)}{\partial d}V^2 = \frac{C_0 V^2}{2}(\frac{1}{d_0} + 2\frac{w}{d_0^2} + 3\frac{w^2}{d_0^3} + 4\frac{w^3}{d_0^4} + ...) . \tag{49}$$

The element has also contact capabilities (47). It is possible to specify a minimal gap and a spring stiffness K_n for the repelling force.
The drawback of this kind of element is that it is limited to the case where the electrodes are (almost) parallel plates, so that the stroke/capacitance function can be evaluated from a single degree of freedom. But the extensions for rotation plates and for 2D cases were developed. The last one with a triangular shape (element *TRANS109*) is useful for simulating structures such as comb drivers and optical MEMS, in which capacitance between the device parts is generally a function of a two-directional displacement. *TRANS126* and *TRANS109* elements enable a huge reduction of the complexity of the system-level simulation.
Let's consider for example the system-level macromodel of an ultrasonic transducer which has two plates with the bottom electrode area A_c and the plate dimension L and which can be presented by nonlinear capacitance:

$$C_{eq} = C_o + (C_{L/2} - C_0)(1 - e^{-\tau t}), \tag{50}$$

where C_0 is the smallest capacitance in the absent of voltage V: $C_0 = \frac{\varepsilon_0 A_c}{d_e}$

$C_{L/2}$ is the largest capacitance when a plate center displacement $w(\frac{L}{2}, t)$ is calculated from the ROM macromodel equations (26):

$$C_{L/2} = \frac{\varepsilon_0 A_c}{d_e - w\left(\frac{L}{2}, t\right)},$$

where d_e is an equivalent gap ($d_e = d_0 + \frac{d_1}{\varepsilon_1} + \frac{d_{ins}}{\varepsilon_{ins}}$); ε_0 is the absolute dielectric permittivity of the vacuum, ε_1 is the relative dielectric permittivity of the poly silicon , ε_{ins} is the relative dielectric permittivity of the insulator; w is a plate deflection .
The largest value of this capacitance corresponds to the value $w_{\max} = d_0$. The electrostatic force acting on the capacitor surfaces is the Coulomb force:

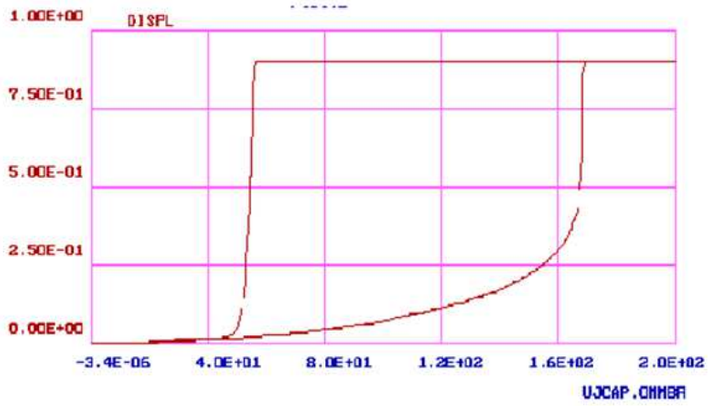


Fig. 17. Capacitive –voltage characteristic an ultrasonic transducer

$$\tilde{F}_{elec} = -\frac{\partial E}{\partial w} = \frac{V_{in}^2}{2}\frac{\partial C_{eq}}{\partial d} = \frac{\varepsilon_0 A_C V_{in}^2}{2\left(d_e - w(\frac{L}{2},t)\right)^2}.$$

As the displacement $w(\frac{L}{2},t)$ reaches the value $w_{\max} = d_0$, the hard stop will restrict its further increase, but input voltage V_n and $\tilde{F}_{elec}$ can be further increased. Assume that the input voltage V_n is a superposition of a constant voltage V_{DC} and a time dependent signal $V(t)$ and that $V_{DC} >> V(t)$. The elastic-plastic properties of the points of contact are simulated by a spring with rigidity K_n and a damper of damping factor b_0. When the beam center moves past w_{max}, it starts interacting with the spring that represents the contact. The damper is introduced to take into account the energy dissipation at the contact. The following equations are used to represent this model:

$$R = \begin{cases} 0 & for\ w \le w_{\max} \\ K_n(w_{\max} - w) - b_o w' & for\ w > w_{\max} \end{cases},$$

where R is the interaction force.

The process of interaction simulated by the force R sometimes can be highly sensitive to the values of the model parameters (rigidity and damping factor), especially if $w_{\max} \approx d_e$. The interaction can induce high-frequency motions and slow down the rate of convergence considerably. Therefore, a good deal of attention must be paid to accurately modeling and representing this process using experimental data.

If the input voltage is increased more, there will be no equilibrium and the plate collapse takes place. In this case, a hard stop or some other arrangement must be introduced to limit the plate motion. During the plate collapse, the difference between the electrostatic force and the elastic force of the spring will continue to increase. Therefore, when the plate drops down into the hard stop, it is not enough to reduce the input voltage below to release the plate. The input voltage should be reduced more to make the electrostatic force at least equal to the elastic force. Hence, the plate capacitance exhibits the *hysteretic* behavior with respect to voltage change.

But if the insulation layer is rather thick its restrictive effect should be taken into account. If the maximal center displacement $w_{\max}$ is equal to initial thickness air gap the moving plate touches the insulator top surface of the electrode when the input V_n voltage reaches the value of $V_{\max}$ ($V_{\max}$ may be calculated from simulation). But as soon as the input voltage will be decreased under this value the plate will leave the hard stop. So, its capacitance does not demonstrate the hysteretic behavior (fig.17).

Parameter τ in (50) defines an ultrasonic transducer frequency band and can be calculated through the plate displacement $w(\frac{L}{2},t)$ and its velocity $v(\frac{L}{2},t) = w'(\frac{L}{2},t)$, which are defied from ROM equations in the following way:

$$\tau = \frac{1}{3} w(\frac{L}{2},t) / v(\frac{L}{2},t) \tag{51}$$

The coefficient 3 appears in (51) due to the fact that a capacitance recharge to 98% for a time value which is equal approximately 3τ.

It is possible to see two included procedures in according to fig.18: one for development of ROM for an ultrasonic transducer plate, where a deflection and speed of central point`s deflection of transducer plate is calculated for the value V_n, and second - for determine of MEMS system- equivalent capacity value (SLM), using values of plate central point co-ordinates .Then the cycle of calculations recurs whereupon.

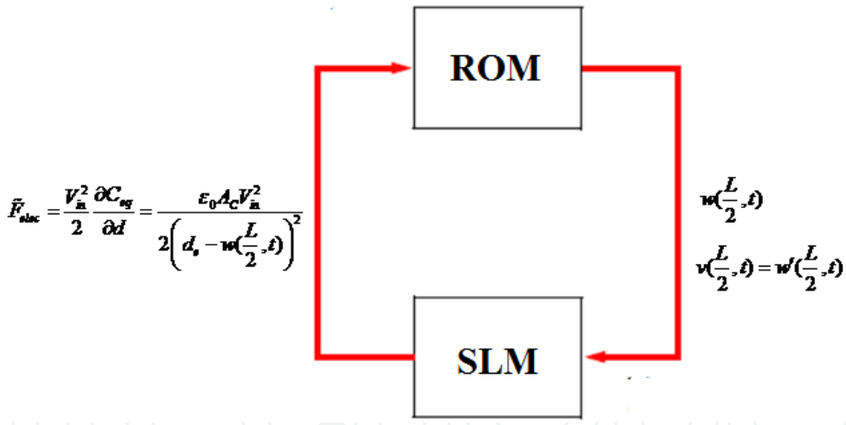


Fig. 18. ROM- system-level model coupled simulations

Instead of using two sequence procedures mentioned above it is possible using functional possibilities of the circuit simulator NetALLTED to built a single system-level equivalent circuit model for an ultrasonic transducer by introducing directly into the equivalent- circuit ROM of mechanical MEMS part the additional arbitrarily connected element (a Depended Source) with an informative function which is determined by equation (50) [42]. Optimization procedures of NetALLTED allow getting the desirable values of this transducer capacity and through it to get a desirable value of output signal of an ultrasonic transducer system-level model by the changing ROM parameters, which, in turn, are depended upon an ultrasonic transducer construction sizes and used material properties.

Providing a single representation of a MEMS operating in multiple physical domains, the electrical circuit approach is very convenient. Moreover, powerful mathematical techniques and circuit simulation programs are available for solving design tasks. It is possible to develop a library of schematic model for different MEMS elements and then use their combinations to build a system-level macromodels for entire rather complicated MEMS constructions.

5. Conclusion

In this chapter, the methods and issues encountered in the development of MEMS macromodels at the system level have been presented. System level modeling is the highest and most abstract level of modeling. This level requires various devices` linking of MEMS *component level models* – both electronic and micromechanical – into a micro-electro-mechanical system. *System-level models* of MEMS components are needed to allow a fast and sufficiently exact investigation of their behavior to simulate entire MEMS.

Starting point for the extraction of a reduced order model (ROM or a macromodel) is already its description with a large ODE system, which is typically derived using physical modeling techniques based on Finite Element Method (FEM) which is rather time consuming. Macromodels application allows the extraction of lower order ODE system that reproduces the input/output behavior with good accuracy. Particular attention has been posed in the chapter on the possibility to get a macromodel circuit presentation.

There are special methods for generating ROM for MEMS components and entire MEMS based on FEM descriptions. To derive macromodels of smaller sizes different approaches (*Modal decomposition, Moment matching, Equivalent circuit presentation)* were developed. Usage of the reduced MEMS components models allows applying successfully modern circuit simulators in workflow for MEMS design on system level.

Three automatic procedures to generate device reduced order macromodels, being based on full FEM/FDM models, were demonstrated in this chapter. Two of them are suitable for simulators with possibilities to get input information in the equation forms (ODE or OAE). The third one in opposite produces macromodels in circuit presentation and so it is more suitable for circuit simulators. The Modal ROM approach is based on using natural (modal or resonant) frequencies of MEMS structure and it is spread mostly in the USA and Asia. The Moment matching ROM approach is based on using the Krylov subspace for transfer functions and it is popular in Western Europe and Asia. The Equivalent circuit ROM approach is based on using a capacitive-inductive-resistive circuit model for mass, damping and stiffness matrices and it is used mostly in Eastern Europe. It is worth to notice that the Modal ROM approach requires some full ANSYS runs to perform a proper orthogonal decomposition during basic functions determination in opposite to the Moment matching and the Equivalent circuit ROM approaches for which it is enough to use ANSYS only for FEM model matrices building.

It seems to be interesting and perspective trying to combine mentioned approaches, for example, to start with Krylov/Arnoldi reduction of ODE dimension, then to build the proper equivalent circuit for obtained ODE systems and finally to apply Y/Δ transformation or n-port transformation for further reducing macromodel order.

6. References

[1] S. D. Senturia, "CAD challenges for microsensors, microactuators, and microsystems," *Proc. IEEE*, vol. 86, pp. 1611–1626, 1998.

[2] E. B. Rudnyi and J. G. Korvink, "Model order reduction for large scale engineering models developed in Ansys." *Lect. Notes Comput. Sc.*, vol. 3732, pp. 349–356, 2006.
[3] M. G.Mand G. Ostergaard, "Electro-mechanical trasducer for MEMS analysis in Ansys," in *Proc. Int. Conf. on Modeling and Simulation of Mycrosystems (MSM)'99*, pp. 270–273, 1999.
[4] W. Z. Lin, K. H. Lee, S. P. Lim, and Y. C. Liang, "Proper orthogonal decomposition and component mode synthesis in macromodel generation for the dynamic simulation of a complex MEMS device," *Journal of Micromechanics and Microengineering*, vol. 13, no. 5, pp. 646–654, 2003.
[5] A.I.Petrenko," Design Methology and Workflow for MEMS Design ", *Proc. MEMSTECH'2011, 11-14 May, Polyana-Svalyava (Zakarpattya), UKRAINE,* pp.12-15, 2011.
[6] Peter Avitabile. "Experimental Modal Analysis- A Simple Non-Mathematical Presentation", *Modal Analysis and Control Lab., University of Massachusetts Lowell*, 16 p.,2001.
[7] Gunner C. Larsen, Morten H. Hansen, Andreas Baumgart, Ingemar Carl. " Modal Analysis of Wind Turbine Blades", *Riso–R–1181(EN), ISBN 87–550–2697–4 (Internet), Riso National Laboratory, Roskilde*, Denmark, 72 p., February 2002.
[8] W. Z. Lin, S. P. Lim, and Y. C. Liang, "Proper orthogonal decomposition and component mode synthesis in macromodel generation for the dynamic simulation of a complex MEMS device," *J. Micromech. Microeng.*, vol. 13, pp. 646–654, 2003.
[9] H. M. Park and D. H. Cho, "The use of the Karhunen-Lo`eve decomposition for the modeling of distributed parameter systems," *Chem. Eng. Sci.*, vol. 51, no. 1, pp. 81–98, 1996.
[10] Y. C. Liang, W. Z. Lin, H. P. Lee, S. P. Lim, K. H. Lee, and D. P. Feng, "A neural-network-based method of model reduction for the dynamic simulation of MEMS," *Journal of Micromechanics and Microengineering*, vol. 11, no. 3, pp. 226–233, 2001.
[11] Elmer S. Hung and Stephen D. Senturia." Generating Efficient Dynamical Models for Micro-electro-mechanical Systems from a Few Finite-Element Simulation Runs", *Journal of Microelectromechanical Systems*, vol. 8, no. 3, September 1999.- pp.280-289.
[12] Lynn D. Gabbay, Jan E. Mehner, and Stephen D. Senturia. "Computer-Aided Generation of Nonlinear Reduced-Order Dynamic Macromodels—I: Non-Stress-Stiffened Case", *Journal of Microelectromechanical Systems*,, vol. 9, no. 2, June 2000.- pp.262-269.
[13] Jan E. Mehner, Lynn D. Gabbay, and Stephen D. Senturia. "Computer-Aided Generation of Nonlinear Reduced-Order Dynamic Macromodels—II: Stress-Stiffened Case", *Journal of Microelectromechanical Systems*, vol. 9, no. 2, June 2000.- pp.270-278.
[14] Schlegel, M.; Bennini, F.; Mehner, J.; Herrmann, G.; Muller, D.; Dozel, W.: "Analyzing and Simulation of MEMS in VHDL-AMS Based on Reduced Order FE-Models", *IEEE Sensors 2003*, Second IEEE International Conference on Sensors, Toronto, Canada, 2003.
[15] Bennini, F.; Mehner, J.; Dotzel, W. " Computational Methods for Reduced Order Modeling of Coupled Domain Simulations", *Proc. of 11 International Conference on Solid State Sensors and Actuators (Transducers 01),* Germany, 2001.
[16] M. Schlegel, F. Bennini, J. E. Mehner, G. Herrmann, D. Mueller, and W. Doetzel, "Analyzing and simulation of MEMS in VHDL-AMS based on reduced-order FE models" , *IEEE Sensors J.*, vol. 5, no. 5, 2005.
[17] J. Chen and S. M. Kang, "An algorithm for automatic model reduction of nonlinear MEMS devices," in *Proc. IEEE Int. Symp. Circuits and Syst.*, pp. 445–448, 2000.

[18] E. S. Hung, Y.-J. Yang, and S. D. Senturia, "Low-order models for fast dynamical simulation of MEMS microstructures" , *Digest of Technical Papers, IEEE Int. Conf. on Solid-State Sensors Actuators and Microsystems (Transducers'97*), pp. 1101–1104, 1997.
[19] L. H. Feng, "Review of model order reduction methods for numerical simulation of nonlinear circuits," *Appl. Math. Comput.*, vol. 167, no. 1, pp. 576–591, 2005.
[20] M. I. Younis, E. M. Abdel-Rahman, and A. Nayfeh, "A reduced order model for electrically actuated microbeam-based MEMS" , *IEEE J. Microelectromech. S.*, vol. 12, no. 5, pp. 672–680, 2003.
[21] A. H. Nayfeh, M. I. Younis, and E. M. Abdel-Rahman, "Reduced-order models for MEMS applications," *Nonlinear Dynamics*, vol. 41, no. 1–3, pp. 211–236, 2005.
[22] Chun-Hsu Ko ,Jin-Chern Chiou." Fuzzy Macromodel for Dynamic Simulation of Microelectromechanical Systems" , *IEEE Trans.on Systems, Man and Cybernetics – Part A: Systems and Humans*, vol. 36, no. 4, pp.823-830, JULY 2006
[23] Pu Li, Yuming Fang. "A Wavelet Interpolation Galerkin Method for the Simulation of MEMS Devices under the Effect of Squeeze Film Damping", Hindawi Publishing Corporation, *Mathematical Problems in Engineering*, Article ID 586718, pp.1-25,2010.
[24] Z. Bai, "Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems," *Applied Numerical Mathematics*, vol. 43, pp. 9–44, 2002.
[25] Zhaojun Bai, Daniel Skoogh," Krylov Subspace Techniques for Reduced-Order Modeling of Nonlinear Dynamical Systems", 2002
[26] Jan Lienemann. "Complexity reduction techniques for advanced MEMS actuators simulation," *Ph.D. dissertation, Albert-Ludwigs Universit at Freiburg im Breisgau*, 298 p.,2006.
[27] A. Odabasioglu, M. Celik, L. T. Pileggi, PRIMA: Passive reduced-order interconnect macromodeling algorithm, *IEEE Trans Comput Aid D*, 17 (8) pp. 645–654,1998.
[28] Lihong Feng. "Review of model order reduction methods for numerical simulation of nonlinear circuits", *Applied Mathematics and Computation, vol. 167* , pp. 576–591,2005.
[29] Z. J. Bai and Y. Su, "SOAR: A second-order Arnoldi method for the solution of the quadratic eigenvalue problem," *SIAM J. Matrix Anal. A*, vol. 26, no. 3, p. 640659, 2005.
[30] Z. Bai, P. M. Dewilde, and R. W. Freund, "Reduced-order modeling," *Numerical Analysis*, vol. 02, pp. 1–59, 2002.
[31] M. Rewienski and J.White, "A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices," *IEEE Trans. Comput.Aid. D.*, vol. 22, p. 155170, 2003.
[32] J. Chen, Sung-Mo Kang, Jun Zou,Chang Liu and José E. Schutt-Ainé. "Reduced-Order Modeling of Weakly Nonlinear MEMS Devices with Taylor-Series Expansion and Arnoldi Approach¨, *IEEE J. Microelectromech. S*, vol. 13, no. 3, pp.441-446, JUNE 2004.
[33] L. Del Tin, J. Iannacci, R. Gaddi, A. Gnudi, E. B. Rudnyi, A. Greiner and J. G. Korvink. "Non Linear Compact Modeling of RF-MEMS Switches by means of Model order rerduction", *Solid-State Sensors, Actuators and Microsystems International Conference, 2007. TRANSDUCERS 2007. Lyon,* 10-14 June 2007, pp.635 – 638.
[34] A.Petrenko, V.Ladogubets, O.Beznosyk, O.Finogenov, "Using Optimization Procedures to Calculate Parameters of MEMS Macromodels", *Proc. CADSM'2009, 24-28 February, Polyana-Svalyava (Zakarpattya), UKRAINE,* pp.511-514, 2009.
[35] Hsu J.L., Vu-Quoc . A rational formulation of termal circuit models for electrothermal simulation. -part 1: finite element method. *IEEE Trans. Circuits & Systems – I:Fund.Theory &Appl.*, 43, 9, pp.721-732, 1996.
[36] Beznosyk O., Ladogubets V., Finogenov O., Tchkalov O. "Using circuit design software to simulate microelectromechanical components // MEMSTECH 2008,

IV-th International Conference on Perspective Technologies and Methods in MEMS Design, Polyana,Ukraine, pp.130-133, 2008.

[37] H. Tilmans, "Equivalent circuit representation of electromechanical transducers: I. lumped-parameter systems," *IEEE J. Electromicromech. Syst.*, vol. 6, no. 1, pp. 157–176, Mar. 1996.

[38] T. Veijola, H. Kuisma, and J. Lahdenper.a, "Dynamic modelling and simulation of microelectromechanical devices with a circuit simulation program," in *Proc. Int. Conf. on Modeling and Simulation of Mycrosystems (MSM)'98*, pp. 245–50, 1998.38. Sheehan JB.N.

[39] TICER: Realizable Reduction of Extracted RC Circuits // *Digest of Technical Papers – IEEE/ACM Proc. of ICCAD*, pp. 200-203, 1999.

[40] Petrenko A., Sigorsky V., "Algorithmic analysis of electronic circuits", *Western Periodical Corp., San Francisco,* 618 p, 1975.

[41] Petrenko A.I.," RLC – circuits models size reduction ", *Proc. CADSM'05, Lviv-Polena*, p.3-8, 2005.

[42] Petrenko A.I., Ladogubets V.V., Tchkalov V.V., Pudlowsky Z.J. ALLTED - a Computer - Aided System for of Electronic Circuit Design, *UICEE.(UNESCO), Melbourne*, 205 p, 1997.

[43] Beznosyk O., Finogenov O., Ladogubets V. Presentation of a System of Ordinary Differential Equations as an Equivalent Electrical Circuit.- // *Perspective Technologies and Methods in MEMS Design : VI-th International Conference MEMSTECH'2010, 20-23 April 2010*, Lviv-Polyana, Ukraine: proc. – Lviv : Publishing House Vezha&Co, 2010. – P. 116–120.

[44] J.Wibbeler, J.Mehner, F.Vogel, F. Bennini. " Development of ANSYS/Multiphysics Modules for MEMS by CAD-FEM GmbH", *19-th CAD-FEM Users' Meeting 2001, International Congress on FEM Technology, Berlin, Potsdam, October* , 2001, pp.1-10.

[45] Sven Reitz, Jens Bastian, Joachim Haase, Peter Schneider, Peter Schwarz." System Level Modeling of Microsystems using Order Reduction Methods", *Journal Analog Integrated Circuits and Signal Processing, vol.37, Issue 1,* 2003, pp.7-16.

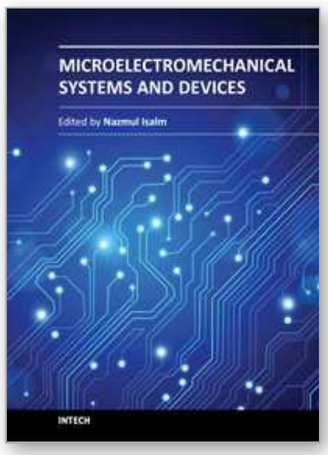
[46] L. Del Tin, R. Gaddi, E. B. Rudnyi, A. Gnudi, A. Greiner, J. G. Korvink, "Nonlinear compact modeling of RF-MEMS switches by means of model order reduction", *Proc. 14th International Conference on Solid-State Sensors, Actuators and Microsystems (Transducer'07), 10-14 June, Lyon* (France).2007.

[47] M. G.Mand G. Ostergaard, "Electro-mechanical trasducer for MEMS analysis in Ansys," in *Proc. Int. Conf. on Modeling and Simulation of Mycrosystems (MSM)'99*, 1999, pp. 270–273.

[48] J. Iannacci, L. Del Tin, R. Gaddi, A. Gnudi and K. J. Rangra,"Compact Modeling of a MEMS Toggle-Switch based on Modified Nodal Analysis", Symposium on Design, Test, Integration and Packagingof MEMS/MOEMS (DTIP) 2005, Montreux, Switzerland, 0103 June 2005.

[49] Peter Schwarz. "Microsystem CAD: From FEM to System Simulation", *Proc. Intern. Conf. "Simulation of Semiconductor Processes and Devices" (SISPAD98), Leuven, 2-4. Sept. 1998*, pp.141-148, Springer, Wien 1998.

[50] Xuan F. Zha, Ram D. Sriram. " Information and Knowledge modeling for computer supported Micro-electro-mechanical systems design and development", *Proceedings of ASME Design Engineering Technical Conference (DETC;2005) September 24-28, 2005, Long Beach, California,* USA, pp.1-11.



Microelectromechanical Systems and Devices

Edited by Dr Nazmul Islam

ISBN 978-953-51-0306-6
Hard cover, 480 pages
Publisher InTech
Published online 28, March, 2012
Published in print edition March, 2012

The advances of microelectromechanical systems (MEMS) and devices have been instrumental in the demonstration of new devices and applications, and even in the creation of new fields of research and development: bioMEMS, actuators, microfluidic devices, RF and optical MEMS. Experience indicates a need for MEMS book covering these materials as well as the most important process steps in bulk micro-machining and modeling. We are very pleased to present this book that contains 18 chapters, written by the experts in the field of MEMS. These chapters are groups into four broad sections of BioMEMS Devices, MEMS characterization and micromachining, RF and Optical MEMS, and MEMS based Actuators. The book starts with the emerging field of bioMEMS, including MEMS coil for retinal prostheses, DNA extraction by micro/bio-fluidics devices and acoustic biosensors. MEMS characterization, micromachining, macromodels, RF and Optical MEMS switches are discussed in next sections. The book concludes with the emphasis on MEMS based actuators.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Anatoly Petrenko (2012). Macromodels of Micro-Electro-Mechanical Systems (MEMS), Microelectromechanical Systems and Devices, Dr Nazmul Islam (Ed.), ISBN: 978-953-51-0306-6, InTech, Available from: http://www.intechopen.com/books/microelectromechanical-systems-and-devices/macromodels-of-micro-electro-

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen