Extraction of heat-transfer macromodels for MEMS devices*

Yao-Joe Yang and Che-Chia Yu

Department of Mechanical Engineering, National Taiwan University, Taipei, Taiwan,
Republic of China

Received 14 September 2003
Published 6 February 2004
Online at stacks.iop.org/JMM/14/587 (DOI: 10.1088/0960-1317/14/4/020)

Abstract

In this paper, a model order reduction technique for MEMS heat-transfer system-level modeling is presented. A 3D heat-transfer solver, which is appropriate for MEMS thermal analysis, is implemented using the finite-difference method (FDM). The numerical models generated by the FDM solver can be reduced into low-order macromodels by an Arnoldi-based technique. This order reduction operation has been implemented as an automatic process. Because the macromodels are generated from the finite-element or the finite-difference (FEM/FDM) approximation of the original solid models, they preserve the original characteristics for most operation conditions. Also, since the orders of the macromodels are much less than those of their original FEM/FDM models, the computational costs are significantly reduced by about two to four orders of magnitude. This performance improvement thus makes the macromodels compatible for system-level or circuit simulations, which is essential for overall performance prediction. We also demonstrate that the macromodel results are in good agreement with the experimental results. The macromodels are also converted into the circuit component modules written by the hardware description language, and are inserted into a circuit simulator for system-level simulations with other circuit components.

1. Introduction

There are numerous heat-transfer applications in MEMS, such as thermal actuation, uncooled infrared sensing, chip cooling, temperature sensing, PCR and so on [1–5]. Most of the applications need feedback loops for precise control, sensing or actuation, and thus require compact and accurate heat-transfer models for system-level modeling [6]. The typical approach for creating compact models is to use the lumped-element methods [6–8]. In order to extract the lumped elements of the system, the device has to be divided into a few heat-transfer sub-systems which have simple geometrical shapes. Then the heat capacity of each sub-system can be approximated as a lumped constant heat capacity (C_{th}), and the energy transportation mechanisms of each sub-system, such as conduction, convection and radiation effects, can be approximated as lumped heat resistors (R_{th}). Alternatively, the lumped elements can also be represented as the thermal impedance matrix [7]. The thermal lumped elements (thermal impedance) can be either derived analytically, or extracted by using finite-element (FEM) or finite-difference (FDM) simulations. Finally, the overall heat-transfer model of the whole system is formed by appropriately connecting the lumped elements of the sub-systems together. Although this approach can create a compact or simplified model for an originally complex system, it might require many costly finite-element (FEM) or finite-difference (FDM) simulations to extract appropriate C_{th} and R_{th} for each sub-system. Furthermore, the accuracy of the lumped-element approach is detrimentally affected if the original device geometry is too complicated to be partitioned into a reasonable number of thermal sub-systems whose lumped-element models can be accurately represented. In recent years, the use of model order reduction techniques to generate MEMS compact models (*macromodels*) has received great attention [9]. The Krylov-subspace methods were proposed to create thermo-electric compact models from the full FEM/FDM models for MEMS as well as semiconductor devices [10–12]. In [10], the application of the Krylov-subspace method on the

* This paper is the extension of the paper that was orally presented in *Transducers '03* (Paper ID: 4D2.4).

0960-1317/04/040587+10$30.00 © 2004 IOP Publishing Ltd Printed in the UK

thermo-electric effect for a microthruster is presented, and the comparison between this method and other order-reduction methods is also provided. The comparison of different versions of the Krylov-subspace methods for thermo-electric applications is presented in [11].

In this work, we apply the Arnoldi algorithm [12–16] to automatically create heat-transfer macromodels from complex three-dimensional (3D) FDM/FEM models, and demonstrate the applications of the technique on the MEMS transduction effects (e.g., electro-thermal actuating as well as thermo-resistive sensing) with various heat-transfer boundary conditions. The goal of this work is to provide MEMS designers compact but accurate heat-transfer macromodels, which are compatible with system-level simulators, for arbitrarily shaped micro devices. Because the macromodels are reduced from the original 3D solid model of the system, potentially they are not only as efficient as the reduced systems consisting of lumped elements, but also are capable of capturing the geometric characteristics which the lumped-element approach lacks.

This paper is organized as follows. The governing equations for heat-transfer effects and the algorithm of the Arnoldi-based technique are presented in section 2. In section 3, two simulated examples and the comparison between different approaches are demonstrated. The convergence of the model discretization and the study on the required orders of the macromodels are presented. The experimental data of thermal actuators are also provided in this section. In section 4, we will demonstrate that the generated compact models can be written in the hardware description language (HDL-AMS), and then plugged into a circuit simulator for coupled analysis with integrated driving or sensing circuitry. Finally, section 5 concludes this paper.

2. Theory

2.1. Heat-transfer governing equations

The governing equation for transient heat-transfer problems is the so-called heat equation (Fourier's equation) [17, 18], as shown below:

$$\nabla^2 T + \frac{1}{k} g(\vec{r},t) = \frac{1}{\alpha}\frac{\partial T}{\partial t}, \qquad \text{Region } \boldsymbol{R}, \quad t > 0 \tag{1}$$

where T is the temperature distribution, $g(\vec{r},t)$is the volume heat generation per unit volume at position $\vec{r}$ and at time t, $\alpha\left(=\frac{k}{\rho c}\right)$ is the thermal diffusivity, ρ is the density, c is the specific heat and k is the thermal conductivity. $\boldsymbol{R}$ is the computational domain (or the MEMS structures in our study), and the boundary and initial conditions can be represented as

$$k_i \frac{\partial T}{\partial n_i} + h_i T = f_i(t), \qquad \text{Boundary } S_i, \quad t \geqslant 0 \tag{2}$$

$$T(\vec{r},t) = F(\vec{r}), \qquad \text{Region } \boldsymbol{R}, \quad t = 0 \tag{3}$$

where$F(\vec{r})$ is the initial temperature distribution, $f_i(\vec{r},t)$ is the heat flux flowing out of the boundary S_i, k_i is the thermal conductivity on the boundary S_i and h_i is the convection coefficient on the boundary S_i.

Since the typical operating temperatures of most MEMS devices are relatively low for noticeable radiation effect [19], the radiation effect is not considered in this study. The governing equation and the boundary conditions can be approximated either using the finite-element method or the finite-difference method [17, 20, 21]. In this work, we develop a 3D transient heat-transfer solver using the finite-difference-based spatial discretization on the governing equation. The time integration is performed explicitly by the Runge–Kutta method [22]. Note that the boundary conditions implemented in the solver can be classified as the Dirichlet type and the Neumann type. The former includes the boundaries of prescribed temperature. The latter includes the boundaries of prescribed heat flux and natural heat convection. The volume condition of prescribed heat generation at each node is also implemented.

2.2. Technique of model order reduction (MOR)

Although the 3D FDM heat-transfer solver is capable of calculating transient behaviors for 3D geometries with various boundary and volume conditions, the computational cost will be very expensive if the 3D thermal computational domain is so complex (i.e., complicated shapes of devices) that a large number of nodes are required. Fortunately, the governing equation and the boundary conditions, as shown in equations (1) and (2), are linear equations, and therefore the system matrices formulated by the FDM or FEM approximation process for equations (1) and (2) can be 'reduced' by an Arnoldi-based *model order reduction* (MOR) technique. The detailed description of the model order reduction procedure is as follows.

The dynamic system equation formulated by the FDM approximation of the governing equation (equation (1)) and the boundary conditions (equation (2)) can be written as

$$\dot{\vec{x}} = \underline{\underline{A}}\vec{x} + \underline{\underline{B}}\vec{u} \qquad \vec{y} = \underline{\underline{C}}^T\vec{x} + \underline{\underline{D}}\vec{u}, \tag{4}$$

where $\underline{\underline{A}}$ is an $n \times n$ matrix, and is formulated by the finite-difference discretization, n is the total number of nodes and $\vec{x}$ is the vector which contains the unknown temperature (state) on each node. The input vector $\vec{u}$ is a set of the input functions. The input functions could be prescribed temperatures, heat flux effects, convection effects or volume heat generations on specific nodes or boundaries. The matrix $\underline{\underline{B}}$ contains the corresponding scaling factors for connecting each input function to each state. The matrices $\underline{\underline{C}}$ and $\underline{\underline{D}}$ can be deliberately designed so that the output vector $\vec{y}$ will contain the quantities of interest. For example, $\vec{y}$ can be the average temperatures of different components or the temperature distribution on a specific area. Examples of formulating the $\underline{\underline{C}}$ matrix for different MEMS modeling cases will be presented in section 3.

In the *Laplace* domain, the transfer function of the system (equation (4)) is

$$\begin{aligned} T(s) &= \underline{\underline{C}}^T(\underline{\underline{I}}s - \underline{\underline{A}})^{-1}\underline{\underline{B}} + \underline{\underline{D}} \\ &= \underline{\underline{C}}^T(\underline{\underline{I}} - s\underline{\underline{A}}^{-1})^{-1}\underline{\underline{b}} + \underline{\underline{D}} \\ \underline{\underline{b}} &= -\underline{\underline{A}}^{-1}\underline{\underline{B}}. \end{aligned} \tag{5}$$

After expanding the transfer function in Taylor series about $s = 0$, we obtain

$$T(s) = \underline{\underline{C}}^T(\underline{\underline{I}} + s\underline{\underline{A}}^{-1} + s^2\underline{\underline{A}}^{-2} + \cdots)\underline{\underline{b}} = \sum_{k=0}^{\infty} m_k s^k,$$

```
Arnoldi − based method {
  $v_1 = b / \|b\|_2$
  for( j = 1 ; j <= q ; j++) {
   /* Get new vector in space */
    solve $Ly = v_j$
    solve $Uw = y$
   /* Orthogonalize new vector */
    for(i = 1; i <= j − 1; i++) {
      $h_{i,j} = w^T v_i$
      $w = w - h_{i,j} v_i$
    }
    $v_{j+1} = w / \|w\|$
  }
}
```

Figure 1. The Arnoldi algorithm.

where m_k are the coefficients of the Taylor series, ($m_k = \underline{\underline{C}}^T(\underline{\underline{A}}^{-k})\underline{\underline{b}}$). The Taylor expansion can be truncated to approximate the transfer function $T(s)$. However, $(\underline{\underline{A}}^{-k})\underline{\underline{b}}$ lines up with a single eigenvector quickly so that this procedure is usually numerically unstable. Therefore, we apply the Arnoldi-based algorithm to stably compute the first q orthogonal bases $\{\vec{v}_i\}$ $i = 1, \ldots, q$ that spans the Krylov subspace:

$$K_q(\underline{\underline{A}}^{-1}, \underline{\underline{b}}) = span\{\underline{\underline{b}}, \underline{\underline{A}}^{-1}\underline{\underline{b}}, \underline{\underline{A}}^{-2}\underline{\underline{b}}, \ldots, \underline{\underline{A}}^{-(q-1)}\underline{\underline{b}}\}. \quad (6)$$

The procedure of generating the orthogonal bases $\{\vec{v}_i\}$ is in fact the Gram–Schmidt process with the initial basis $\{\underline{\underline{b}}, \underline{\underline{A}}^{-1}\underline{\underline{b}}, \underline{\underline{A}}^{-2}\underline{\underline{b}}, \ldots, \underline{\underline{A}}^{-(q-1)}\underline{\underline{b}}\}$. Given the matrix $\underline{\underline{V}}$ whose columns are $\{\vec{v}_i\}$, the reduced system (macromodel) is

$$\dot{\vec{x}} = \underline{\underline{A_q}}\vec{x}_q + \underline{\underline{B_q}}\vec{u} \qquad \vec{y} = \underline{\underline{C_q}}^T\vec{x}_q + \underline{\underline{D}}\vec{u}, \quad (7)$$

where

$$\underline{\underline{A_q}} = \underline{\underline{v}}^T\underline{\underline{Av}} \qquad \underline{\underline{B_q}} = \underline{\underline{v}}^T\underline{\underline{B}} \qquad \underline{\underline{C_q}} = \underline{\underline{v}}^T\underline{\underline{C}}.$$

The Arnoldi algorithm for generating $\underline{\underline{V}}$ is listed in figure 1. Note that the reduced system, as shown in equation (7), has the same input ($\vec{u}$) and the output ($\vec{y}$) as the original system (equation (4)). Since the typical ranks of $\underline{\underline{A_q}}$, $\underline{\underline{B_q}}$ and $\underline{\underline{C_q}}$ are very small, the computational efficiency for simulating transient responses and frequency responses of the reduced model is significantly increased. In other words, the reduced system still provides the results of interest (the output $\vec{y}$) based on the prescribed variables (input $\vec{u}$), but requires much less state variables ($\vec{x}_q$) than the original system. Note that the original geometric information and the boundary conditions cannot be completely retrieved from the macromodel if q is less than n, but it still provides excellent accuracy for most practical MEMS cases even though q is *much* less than n. Various MEMS examples of generating macromodels as well as the study on the macromodel accuracy will be presented in section 3. The procedure of the heat-transfer macromodeling from creating original device 3D solid models to performing system-level simulation is shown in figure 2. This procedure of generating macromodels has been implemented as an automatic process in this work.

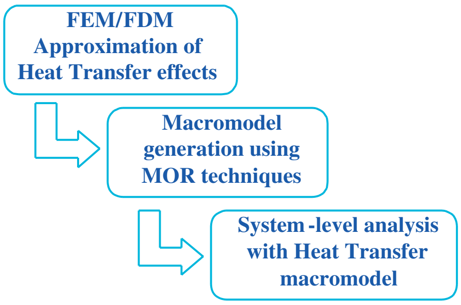

Figure 2. The procedure of the macromodeling.

It has to be pointed out that the Arnoldi method described in this work is not capable of dealing with nonlinear systems [16]. Therefore, many material properties used in this work (e.g., the electrical conductivity and the thermal conductivity) are assumed constant. In practice, however, some of these material properties are nonlinear functions of temperature, and the modeling error introduced by these nonlinearity effects becomes significant when temperature variation is large. The discussions on the temperature dependence of polysilicon properties for electrothermal microactuators can be found in [23].

3. Applications of the MOR techniques and discussions

In this section, two examples of applying the MOR technique for efficient and accurate MEMS heat-transfer modeling are presented. The first example provides the estimation of the tip deflection of a thermal actuator based on the temperature calculation from the macromodels generated by the technique. Measured results are also presented. The second example demonstrates the temperature calculation for the thermal absorption element of an infrared imager (bolometer) using a macromodel. The macromodel results are verified with the FDM solver and/or the commercial FEM package *Coventorware* [24]. The study on the accuracy versus the orders of the reduced models will also be discussed.

3.1. Case study I: thermal flexure actuator

Figure 3 shows the CCD picture of a thermal actuator [1] fabricated by the multi-user MEMS process (MUMPS) [25]. The device's movable polysilicon structure, which is designed for in-plane motion, is supported by the two anchors. With a constant electrical current flowing through the hot arm and the cold arm (when a constant voltage is applied across the two contact pads), the asymmetric shape of the device causes the current density in the hot arm to be larger than that in the cold arm. The uneven heating arising from the asymmetric current density will make the temperature in the hot arm higher than that in the cold arm. The thermal expansion difference between the two arms results in an in-plane bending of the whole structure.

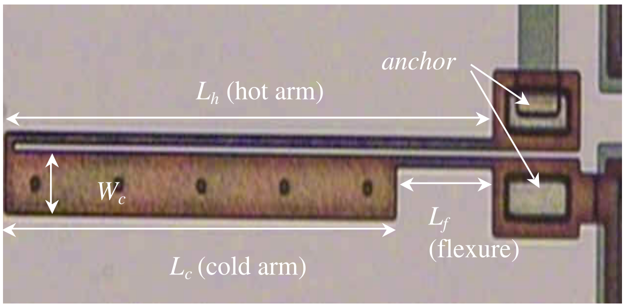

Figure 3. The CCD picture of the experimental device. Note that the contact pads are on the right side of the picture, and are not completely shown.

Table 1. The dimensions of the thermal actuator.

Geometrical dimensions	Length (μm)
Length of hot arm (L_{h})	200
Width of hot arm (W_{h})	3
Length of cold arm (L_{c})	160
Width of cold arm (W_{c})	16
Gap (g)	3
Length of flexure (L_{f})	40
Width of flexure (W_{f})	3
Thickness (t)	2

Table 2. The mechanical, thermal and electrical properties of the thermal actuator (polysilicon).

Mechanical property	
Density (kg μm^{-3})	2.23×10^{-15}
Thermal property	
Thermal conductivity (pW/μm K)	1.48×10^{8}
Specific heat (pJ/kg K)	1.0×10^{14}
Convection coefficient (pW/μm^2 K)	5
Electrical property	
Sheet resistance (Ω/sq)	10

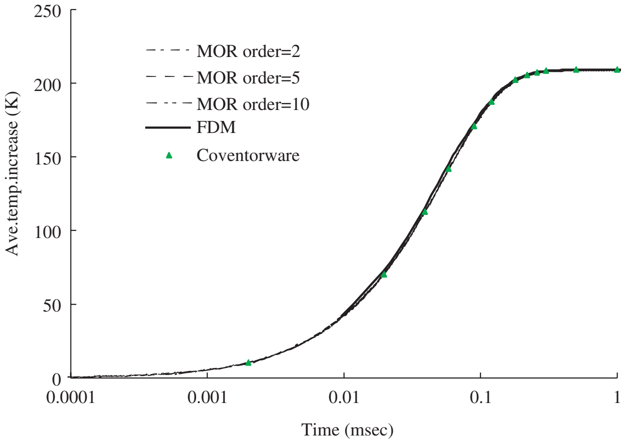

Figure 4. Comparison of the FDM, the macromodel and the Coventorware results for the average temperature increase of the thermal actuator.

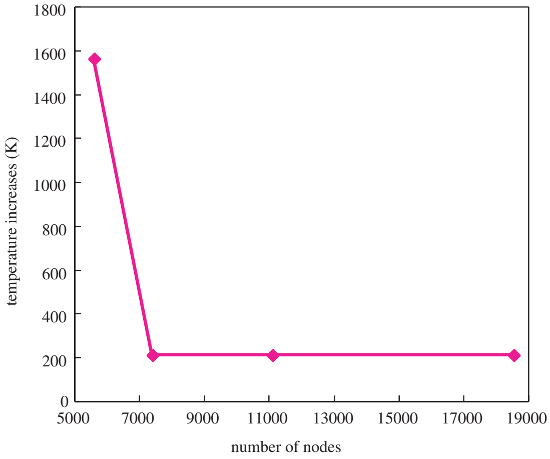

Figure 5. Average temperature increase of the device calculated by the FDM solver for different numbers of nodes.

This heat-transfer effect of the thermal actuator is modeled by imposing several thermal boundary and volume conditions on the structure. Natural convection boundaries are applied to the surfaces adjacent to the air. Constant-temperature boundaries are applied to the interfaces between the anchors and the substrate [19]. A volume condition of heat-generation distribution (W m^{-3}) is applied to the whole structure. The heat-generation distribution is determined by the electrical current density distribution, which is accurately estimated by using the Coventorware. The mechanical, thermal and electric properties as well as the dimensions of the measured and simulated device are listed in tables 1 and 2. Note that the temperature dependence of electrical resistivity is not included in the model.

Figure 4 shows the average temperature increases versus time calculated by the FDM solver, the Coventorware FEM solver and the macromodel. The temperatures on the bottom surfaces of the anchors are fixed at 300 K. The applied voltage across the contact pads is 3 V. For the FDM model, the thermal actuator is meshed into 11 127 nodes. For the Coventorware FEM model, the same structure is also meshed into 11 127 nodes (5168 linear brick elements). The order of the macromodel is 5. Figure 5 shows the average temperature of the device calculated by the FDM solver for different number of nodes, and indicates that the calculated solutions monotonically converge to a stable value for the node numbers greater than 8000. Therefore, the FDM/FEM discretizations with 11 127 nodes are satisfactory for this specific case. Under the same discretization conditions (i.e., the same number of nodes), it is well known that the FEM approaches potentially are more accurate than the FDM approaches because most FEM solvers use ‘piecewise-linear’ approximation to describe field distribution, while typical FDM solvers approximate field distributions in a ‘discrete’ manner (piecewise constant) between each node [26]. Since the temperature variation is relatively mild inside the device structure, the simulated transient results between the Coventorware (FEM) and the FDM solvers are indistinguishable (less than 0.01%). Figure 6 shows the relative average error versus time. Note that the relative average error in the figure is defined as

$$e(t) = \left| \frac{T_{\mathrm{ave}}(t) - \hat{T}_{\mathrm{ave}}(t)}{T_{\mathrm{ave}}(t)} \right|, \tag{8}$$

where $T_{\mathrm{ave}}(t)$ is the average temperature of the device calculated by the FDM solver over time, and $\hat{T}_{\mathrm{ave}}(t)$ is

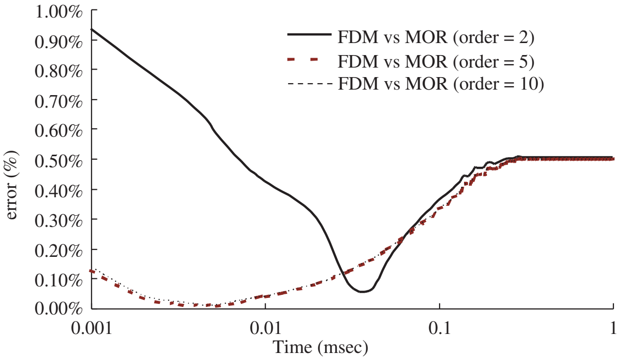

Figure 6. The discrepancy between the FDM solver and the macromodels with different orders.

Table 3. The performance comparison of the FDM solver and the macromodels.

	Simulation time (s)	Speed-up factor	Error (%)
MOR order = 2	<0.01	$>2 \times 10^6$	0.93
MOR order = 5	0.01	2×10^6	0.50
MOR order = 10	0.44	4×10^4	0.50
FDM	18743.000		N/A

the average temperature of the device calculated by the macromodels over time. Table 3 presents the comparison of the performance as well as the accuracy between the results by the FDM solver and by the macromodels with various orders. These preliminary numerical results show that the macromodels of order 5 (or higher) match full-meshed FDM/FEM results very well (1% error). The results also demonstrate that the macromodels give three to four order-of-magnitude reductions in computational times. All the simulations are performed using a Windows-based PC with single Pentium-III 800 MHz CPU and 512 MB RAM.

Note that the computational time for reducing a full-meshed model to a macromodel is about 15 s per order for this case. This computational time is also called *the model generation time*, and is in fact a 'one time charge' in the macromodeling process. Once a macromodel is generated, all the future simulations using the macromodel will not incur *the model generation time* again. Therefore, the comparison of the computational performance between the full-meshed models and the macromodels does not include effect of *the model generation time*. Moreover, the model generation times are also negligible when compared with the computational time required for a transient full-meshed simulation for a meaningful time period (e.g., about the thermal time constant of the device, 0.3 ms). Figure 7 shows the steady state temperature distribution of the thermal actuator at the time 3×10^{-4} s. This figure also indicates that the highest temperature occurs at the hot-arm segment far away from the anchor. Figure 8 shows the frequency responses of the macromodels with different orders. Note that the output for this case is the average temperature change of the device, and the excitation is the volume heat generation. The discrepancies in average temperature change are less than 3% between the results of the macromodels with orders greater than 2 (from dc to 100 MHz). On the other hand, the result of the second-order macromodel deviates significantly (more than 10%) from all other macromodels as the frequency is greater than 10 kHz.

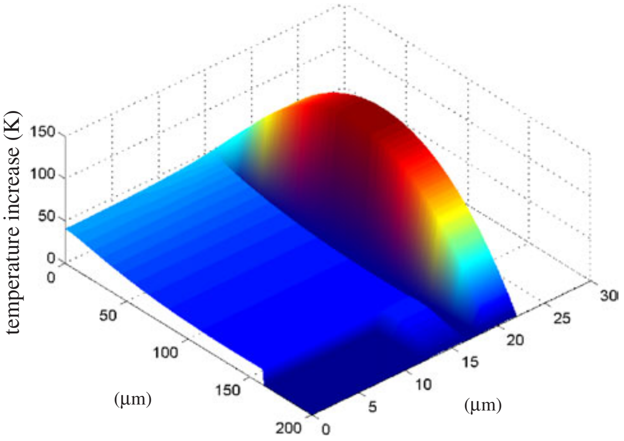

Figure 7. The temperature distribution on the top surface of the thermal actuator calculated by the FDM solver.

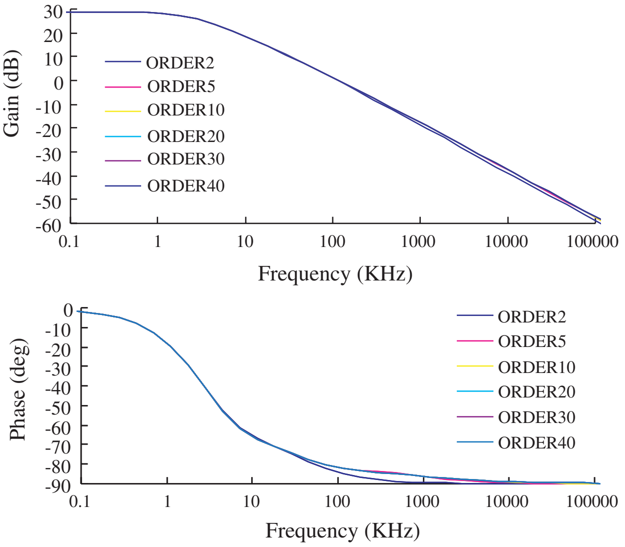

Figure 8. The frequency responses of the average temperature increase of the actuator for different orders.

The measured tip deflections of the thermal actuator are also compared with the results estimated by the macromodels. The analytical relationship between the tip deflection and the temperatures in each segment has been derived in [27]. The following derivation shows the integration of the heat-transfer macromodel, the analytical relationship in [27] and the corresponding material properties [28].

The deflection of the thermal actuator can be represented as

$$d = \frac{2{L_\mathrm{h}}^2}{Et{W_\mathrm{h}}^3}(R_1L_\mathrm{h} - 3R_3), \tag{9}$$

where E is the Young's modulus, t is the thickness of the thermal actuator, L_h and W_h can be found in table 1 and R_1 and R_3 can be obtained by solving the following equations:

$$\vec{R} = \underline{\underline{K}}^{-1}\vec{X}, \tag{10}$$

where

$$\vec{R} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}, \qquad \underline{\underline{K}} = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}$$

$$\vec{X} = \begin{bmatrix} 0 \\ \alpha_T[L_h(\overline{T}_h - T_s) - L_c(\overline{T}_c - T_s) - L_f(\overline{T}_f - T_s)] \\ 0 \end{bmatrix},$$

where L_h, L_f and L_c are described in table 1, and the analytical forms of k_{ij} can be found in appendix A. $\overline{T}_h$, $\overline{T}_c$ and $\overline{T}_f$ are the average temperature of the hot arm, the cold arm and the flexure, respectively. Finally, α_T (K^{-1}) is the coefficient of thermal expansion of polysilicon [28], and is a function of the temperature in the structure:

$$\alpha_T = \begin{Bmatrix} 3.725[1 - \exp(-5.88 \times 10^{-3}(T_{avg} - 124))] \\ +5.548 \times 10^{-6} T_{avg} \end{Bmatrix} \times 10^{-6}. \tag{11}$$

The deflection of the actuator can be evaluated by plugging into the analytical model (equations (9) and (10)) the average temperature increases of the hot arm, the cold arm and the flexure ($\overline{T}_h$, $\overline{T}_c$ and $\overline{T}_f$), which can be directly calculated by the macromodels if the $\underline{\underline{C}}$ matrix (see equation (4)) is deliberately arranged so that the elements in the output vector $\vec{y}$ will be the temperatures of interest:

$$\vec{y} = \begin{bmatrix} \overline{T}_h \\ \overline{T}_c \\ \overline{T}_f \end{bmatrix} \tag{12}$$

$$\underline{\underline{C}} = \begin{bmatrix} \dfrac{[H_1 \quad H_2 \quad \dots \quad H_n]}{n_h} \\ \dfrac{[C_1 \quad C_2 \quad \dots \quad C_n]}{n_c} \\ \dfrac{[F_1 \quad F_2 \quad \dots \quad F_n]}{n_f} \end{bmatrix} \tag{13}$$

$$\underline{\underline{D}} = 0, \tag{14}$$

where n_h, n_c and n_f are the total number of the nodes in the hot arm, the cold arm and the flexure, respectively. H_i, C_i and F_i are either 1 or 0; for the first row vector of the $\underline{\underline{C}}$ matrix, if the node i is inside the hot arm, then H_i is equal to 1. Otherwise it is 0. The same criterion is applied to the second and the third rows for the cold arm (C_i) and the flexure (F_i), respectively.

Figure 9 shows the comparison of the transient average temperature increases of the hot arm, the cold arm and the flexure for different approaches (when an input voltage of 3 V is applied). The deviation between the macromodel (order 5) and the FDM solver is within 0.5%. Figure 10 shows the average temperature increases in different parts ($\overline{T}_h$, $\overline{T}_c$ and $\overline{T}_f$, the components of the output vector $\vec{y}$) of the devices calculated by a macromodel (order 5) under various input voltages from 0 to 6.5 V. The comparison of the measured and the calculated in-plane deflection of the actuator is shown in figure 11. At low temperature, the measured results match with the FDM solver and the macromodels very well. However, at high input voltage, the simulated results over-predict the deflections. It is speculated that the discrepancy is due to the fact that the heat loss from radiation effect becomes notable when a high input voltage is applied. The same phenomenon for the thermal actuator has also been observed in [27], in which the quantitative discussion on this phenomenon for different dimensions of the actuators can be found.

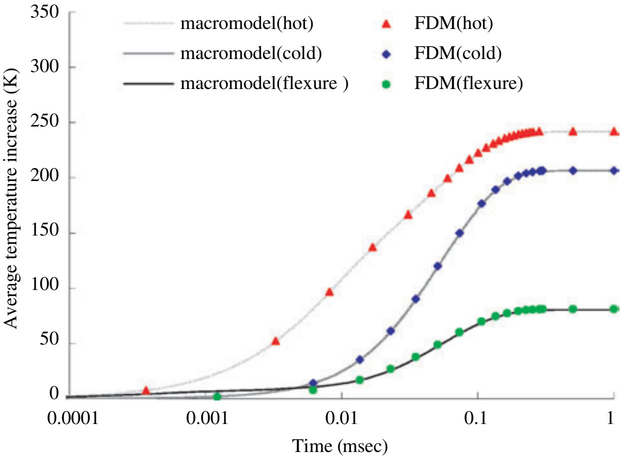

Figure 9. The average temperature increases of the hot arm, the cold arm and the flexure calculated by the macromodels and the FDM solver. These calculated temperatures are used for estimating the tip deflection of the thermal actuator.

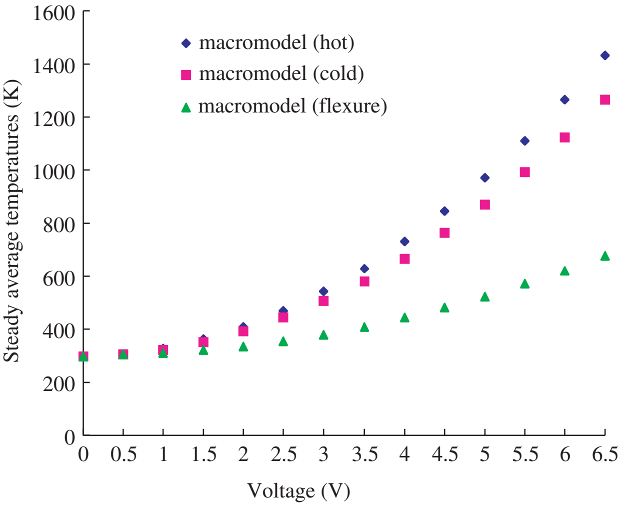

Figure 10. Steady average temperatures of the hot arm, the cold arm and the flexure calculated by the macromodels with different input voltages.

3.2. Case study II: infrared imager absorber

In this subsection, we demonstrate that the heat-transfer macromodels can be used to simulate the absorber of an infrared imager. Figure 12 shows a typical schematic of a pixel cell in an infrared imager [2]. This cell consists of a Si_3N_4 suspended structure (absorber) of 0.5 μm in thickness and two supporting tethers. A thin film (50 nm) of vanadium oxide (VO_x), which has a large temperature coefficient of resistance (TCR), is deposited on the suspended structure. The temperature at the two bottom surfaces of the two anchors is fixed at 300 K. The convection effect is ignored since typically

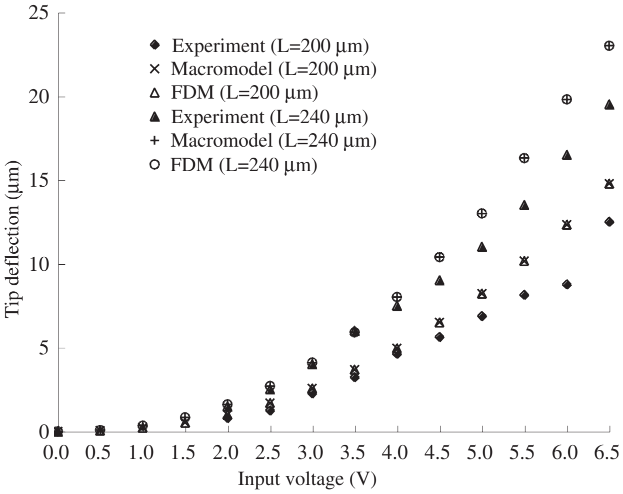

Figure 11. Comparison of the measured and the calculated tip displacements with input voltages from 0 V to 6.5 V for two different actuators. Note that the experimental data are from 2 to 6.5 V.

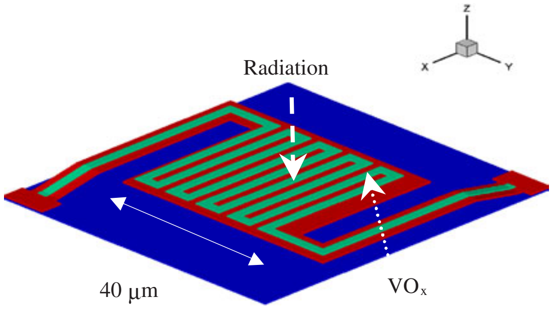

Figure 12. The schematic of an infrared imager cell.

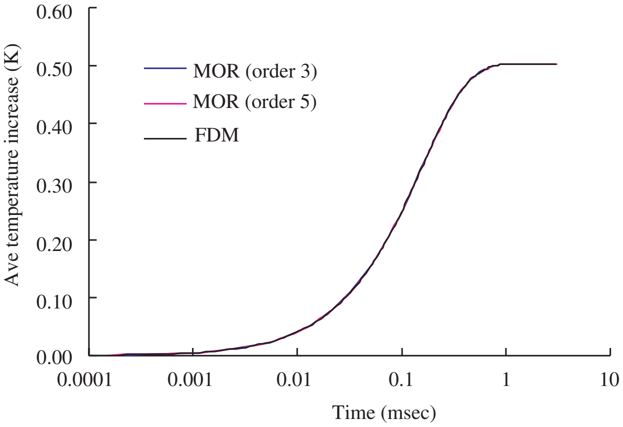

Figure 13. Comparison of average temperature increase at the top surface of the infrared sensor calculated by the FDM solver and the macromodels with different MOR orders.

an infrared imager chip is operated at pressures lower than 100 mTorr. The initial temperature of the whole device is 300 K. The size of the absorber is 40 × 40 μm^2, and the tethers are 50 μm long and 8 μm wide.

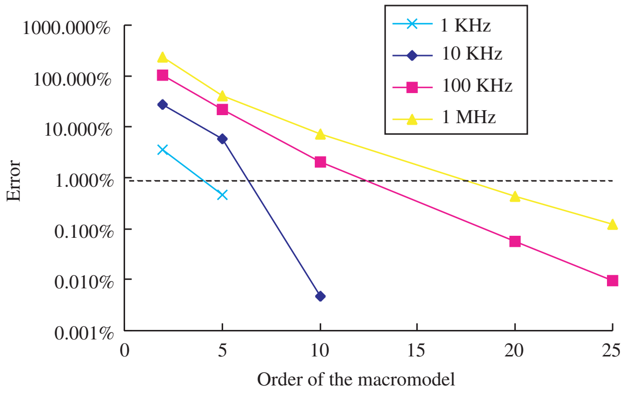

Figure 14. Error versus the order of the macromodels of the infrared imager cell for different frequencies. The definition of the error is the average difference between the results by the FDM solver and the results by the macromodels for one period of the input flux.

Table 4. The mechanical and thermal properties of the infrared sensor.

Mechanical property (silicon nitride)	
Density (kg μm^{-3})	3.18×10^{-15}
Thermal property (silicon nitride)	
Thermal conductivity (pW/μm °C)	3.01×10^{7}
Specific heat (pJ/kg °C)	1.7×10^{14}
Convection coefficient (pW/μm^2 K)	5

Table 5. The performance comparison of the FDM solver and the MOR solver.

	Simulation time (s)	Speed-up factor	Error (%)
MOR order = 3	0.44	3×10^{4}	0.868
MOR order = 5	3.37	4×10^{3}	0.397
FDM	12 421		N/A

In the FDM simulation, this device is meshed into 7395 nodes. The thermal properties are listed in table 4. Figure 13 shows the transient results calculated by the FDM solver and the macromodels. The input heat flux impinging on the top of the absorber is 10^{-9} W μm^{-2}. The figure indicates that the results by the macromodels with orders greater than 3 agree with the FDM result very well. This preliminary calculated result also shows that the intrinsic thermal time constant of the pixel is about 200 μs, which is adequate for a readout circuitry with a frame rate of 60 Hz. Table 5 shows the comparison of the computational cost and the accuracy between the FDM solver and the macromodels. About a four order-of-magnitude reduction in computation cost is achieved for the macromodel of order 3.

Figure 14 shows the relationship between the errors and the orders of macromodels for periodic input flux signals under different frequencies. The definition of the error is the average difference between the results by the FDM solver and the results by the macromodels for one period of the input flux. Obviously, the higher the operation frequency, the higher the required order of the macromodel under a pre-defined condition of error tolerance. For example, if the tolerance of error is less than 1%, the minimum order is about 20 for the device whose maximum operation frequency is below 1 MHz. Therefore, this figure can be used as a guideline for choosing appropriate orders of macromodels at different maximum operation frequencies.

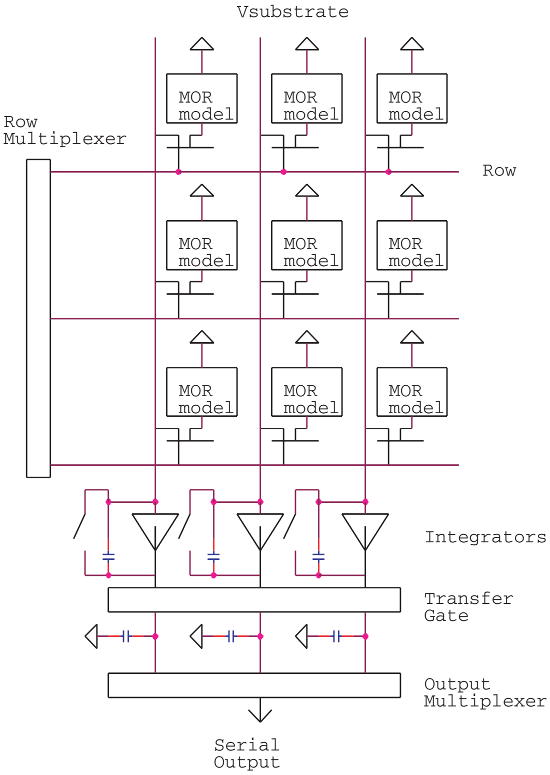

Figure 15. The readout circuit schematic with the heat-transfer macromodels (MOR models) written in HDL.

4. HDL-based simulation

In this section, we will demonstrate that the macromodel for the IR imager cell can be plugged into a circuit simulator by using the analog hardware description language (HDL-AMS) [29, 30]. The schematic of the readout circuit for the pixel array of an infrared imager is shown in figure 15 [2]. Each heat-flux-sensitive resistor element is connected between $V_{\text{substrate}}$ and a MOS switch. In order to integrate the heat-transfer macromodel with control/sensing circuitry, it is necessary to make the macromodel an 'entity' of an electric resistor written in HDL-AMS. In order to increase the sensitivity, the total resistance of the VO$_x$ material needs to be large. Therefore, the typical layout of the VO$_x$ layer is shown in figure 12, and can be approximated as a long but thin line. Based on this approximation, the total resistance of the VO$_x$ layer can be written as a function of the local average temperature variation of the layer:

$$\mathbf{R}(T) = \int_0^L \frac{\rho}{A(l)}[1 + \alpha \cdot \Delta T(l)]\,\mathrm{d}l, \tag{15}$$

where α is the temperature coefficient of resistance (TCR), L is the effective total length of the VO$_x$ layer, $\Delta T(l)$ is the local average temperature change at the location l, $A(l)$ is the cross-section area of the VO$_x$ layer at the location l, ρ is the resistivity of the VO$_x$ layer. The TCR of the VO$_x$ layer is $-0.02\ \mathrm{K}^{-1}$ [2]. In order to further simplify equation (15), we assume that the cross-sectional area of the VO$_x$ layer is uniform and is equal to A. The total resistance can be written as a function of the average temperature change of the VO$_x$ layer:

$$\mathbf{R}(\Delta T_{\text{avg}}) = \frac{\rho L}{A}(1 + \alpha \cdot \Delta T_{\text{avg}}), \tag{16}$$

where ΔT_{avg} is the average temperature increase. Consequently, the output y of equation (4) is formulated as the average temperature variation (ΔT_{avg}) of the VO$_x$ layer. The corresponding $\underline{\underline{C}}$ matrix is

$$\underline{\underline{C}}^T = \left[\frac{I_1}{n_r} \quad \frac{I_2}{n_r} \quad \cdots \quad \frac{I_{n_r}}{n_r}\right], \tag{17}$$

where I_i is equal to 1 if node-i is located on the VO$_x$ layer. Otherwise I_i is zero. n_r is the total number of the nodes which are located on the VO$_x$ layer.

It has to be emphasized that the resistance derivation described above is primarily for an intuitive demonstration of formulating the $\underline{\underline{C}}$ matrix. A more accurate resistor model, which includes the correction for the non-rectangular

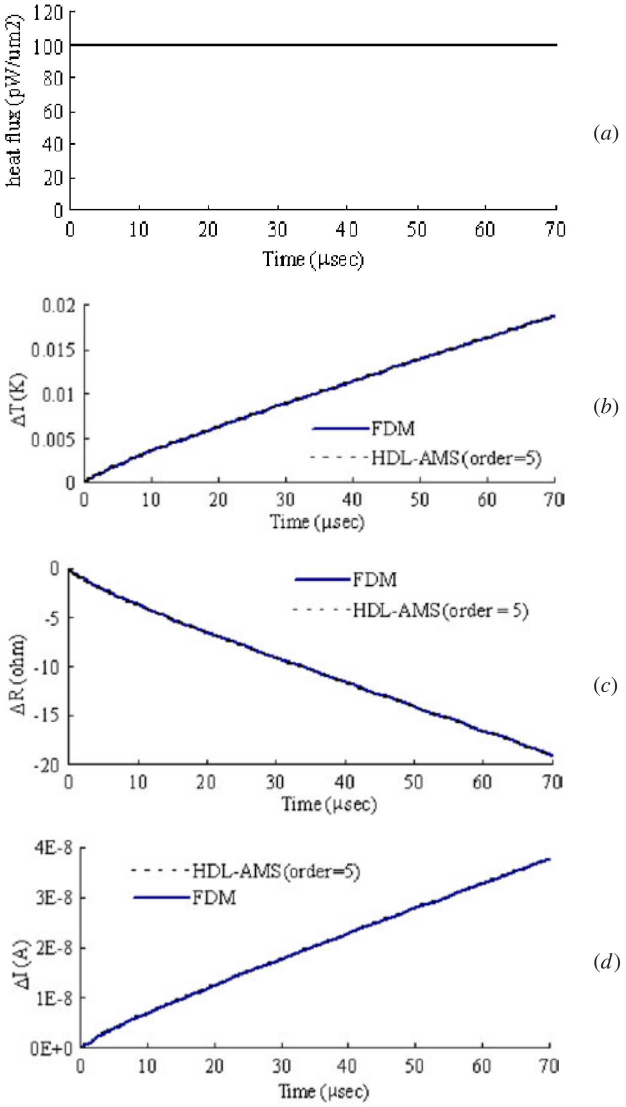

Figure 16. Simulated VHDL-AMS results for a constant input heat flux impinging on the top surface of the imager cell: (*a*) heat flux versus time. (*b*) The simulated temperature variation simulated by the solver. (*c*) Resistance versus time. The initial resistance is 50 kΩ. (*d*) The current variation versus time.

shapes of resistor layouts, can be easily derived based on any VLSI design textbooks [31]. The HDL-AMS template of a second-order macromodel for the heat-flux-sensitive resistor (equation (16)) is listed in appendix B. As shown in figure 15, since the input terminals of the monolithic integrators are held at ground potential, the bias voltage of each resistor is equal to $V_{\mathrm{substrate}}$. Assuming that there are 256 pixels in each row and that the typical frame rate of the imager is 60 Hz, each pixel in a row is biased at $V_{\mathrm{substrate}}$ for 65 μs ($= 1/(60 \times 256)$). The electric current flowing through each resistor is integrated by a monolithic integrator. The integrators' outputs are transferred to storage registers before being multiplexed out serially. The capacitance used in the simulation is 10 pF. Figures 16(*a*)–(*d*) show the simulated results for a pixel of the infrared imager under a constant heat flux for 65 μs. Figure 16(*a*) is the input flux. The average temperature increase of the nodes on the VO$_x$ layer is shown in figure 16(*b*). Figure 16(*c*) is the corresponding resistance change due to the temperature increase caused by the input flux. Since the TCR of the VO$_x$ is negative, the resistance curve decreases as the temperature of the absorber increases. Figure 16(*d*) is the output current change due to the resistance change. Figure 16 also indicates that the HDL-based simulated results by the SMASH [32] and the results by the FDM solver are almost indistinguishable (less than 1% discrepancy).

5. Conclusion

A model order reduction technique for generating heat-transfer macromodels for MEMS devices is presented in this work. This technique reduces the matrices created by the FEM/FDM discretization into low-order macromodels which are compatible for system-level analysis. Transient simulated results for two MEMS thermal devices are presented. The comparison between the full-meshed models (the original FEM/FDM solvers) and the macromodels are also provided. In terms of computational cost, the macromodels are about three to four orders-of-magnitude faster than the full-meshed models for less than 1% discrepancy. The experimental results of thermal actuators are also in good agreement with the macromodel results. The relationship between the required orders of macromodels for different operation frequencies is also studied. For an infrared imager absorber under a heat flux of 1 MHz, the required macromodel order is about 20, which is much less than the order of its full-meshed model. The macromodels are converted as an 'entity' of a circuit simulator using HDL-AMS. The system-level simulations, which include the HDL-based heat-transfer macromodel and other circuit components, are also demonstrated.

Acknowledgments

This work is partially supported by National Science Council of Taiwan, R.O.C., contract number NSC 90-2218-E-002-031. The authors would like to thank Dr Chih-Chun Wang at MIT for his assistance in the HDL-based modeling. Technical comments and suggestions on frequency and mesh convergence study from Professor Stephen D Senturia of MIT are also greatly appreciated.

Appendix A. The analytical forms of the entries of the $\underline{\underline{K}}$ matrix in equation (10)

$$k_{11} = \frac{4}{EtW_{\mathrm{h}}^3}\left(L_{\mathrm{h}}^3 + L_{\mathrm{f}}^3\right) + \frac{4}{EtW_{\mathrm{c}}^3}\left(3gL_{\mathrm{h}}^2 + L_{\mathrm{h}}^3 - L_{\mathrm{f}}^3\right)$$

$$k_{12} = -\frac{6}{EtW_{\mathrm{h}}^3}\left(gL_{\mathrm{f}}^2\right) - \frac{6}{EtW_{\mathrm{c}}^3}\left(g^2L_{\mathrm{h}} + gL_{\mathrm{h}}^2 - gL_{\mathrm{f}}^2\right)$$

$$k_{21} = -\frac{6}{EtW_{\mathrm{h}}^3}\left(gL_{\mathrm{f}}^2\right) - \frac{6}{EtW_{\mathrm{c}}^3}\left(g^2L_{\mathrm{h}} + gL_{\mathrm{h}}^2 - gL_{\mathrm{f}}^2\right)$$

$$k_{13} = -\frac{6}{EtW_{\mathrm{h}}^3}\left(L_{\mathrm{h}}^2 + L_{\mathrm{f}}^2\right) - \frac{6}{EtW_{\mathrm{c}}^3}\left(L_{\mathrm{h}}^2 + 2gL_{\mathrm{h}} - L_{\mathrm{f}}^2\right)$$

$$k_{31} = -\frac{6}{EtW_{\mathrm{h}}^3}\left(L_{\mathrm{h}}^2 + L_{\mathrm{f}}^2\right) - \frac{6}{EtW_{\mathrm{c}}^3}\left(L_{\mathrm{h}}^2 + 2gL_{\mathrm{h}} - L_{\mathrm{f}}^2\right)$$

$$k_{22} = \frac{4}{EtW_{\mathrm{c}}^3}(g^3 + 3g^2L_{\mathrm{c}}) - \frac{12}{EtW_{\mathrm{h}}^3}(g^2L_{\mathrm{f}})$$

$$k_{23} = \frac{6}{EtW_{\mathrm{c}}^3}(g^2 + 2gL_{\mathrm{c}}) + \frac{12}{EtW_{\mathrm{h}}^3}(gL_{\mathrm{f}})$$

$$k_{32} = \frac{6}{EtW_{\mathrm{c}}^3}(g^2 + 2gL_{\mathrm{c}}) + \frac{12}{EtW_{\mathrm{h}}^3}(gL_{\mathrm{f}})$$

$$k_{33} = \frac{12}{EtW_{\mathrm{c}}^3}(L_{\mathrm{h}} + L_{\mathrm{f}}) + \frac{12}{EtW_{\mathrm{c}}^3}(L_{\mathrm{c}} + g).$$

Appendix B. VHDL-AMS entity of the macromodel for an infrared imager cell

```
>>> VHDL
library IEEE;
use IEEE.electrical_systems.all;
use IEEE.math_real.all;
entity MOR is
generic(
   alpha : real :=-0.02;
   R_node : real :=1681.0;
   R0 : real :=50000.0;
   b1 : real :=3.2959e5;
   b2 : real := 1.0568e5;
   a11 : real :=-0.0859e5;
   a12 : real :=-1.7218e5;
   a21 : real :=-0.0276e5;
   a22 : real :=-2.2491e5;
   c1 : real :=22.2723;
   c2 : real :=7.1415);
   port(terminal P, N : electrical);
end MOR;
architecture behav of MOR is
   QUANTITY VPTON across IPTON through P to N;
   QUANTITY y : real;
   QUANTITY s1 : real;
   QUANTITY s2 : real;
   QUANTITY R : real;
```

```
begin
   break s1 => 0.0, s2 => 0.0;
   s1'dot == a11*s1+a12*s2+b1;
   s2'dot == a21*s1+a22*s2+b2;
   y == ((c1*s1+c2*s2)/R_node);
   R == R0*(1.00-alpha*y);
   VPTON == R*IPTON;
end behav;
```

References

[1] Comtois J H and Bright V M 1996 Surface micromachined polysilicon thermal actuator arrays and applications *Tech. Dig. of 1996 Solid-State Sensor and Actuator Workshop (Hilton Head Island, SC, June 1996)* pp 174–7

[2] Cole B E, Higashi R E and Wood R A 1998 Monolithic two-dimensional arrays of micromachined microstructures for infrared applications *Proc. IEEE* **86** 1679–86

[3] Amon C H, Murthy J Y, Yao S C, Narumanchi S, Wu C-F and Hsieh C-C 2001 MEMs-enabled thermal management of high-heat-flux devices, EDIFICE: embedded droplet impingement for integrated cooling of electronics *Exp. Therm. Fluid Sci.* **25** 231–42

[4] Northrup M A, Ching M T, White R M and Watson R T 1993 DNA amplification with a microfabricated reaction chamber *Proc. 7th Int. Conf. on Solid-State Sensors and Actuators (Transducers '93) (Yokohama, Japan, June 1993)* pp 924–6

[5] Daniel J H, Iqbal S, Millington R B, Moore D F, Lowe C R, Leslie D L, Lee M A and Pearce M J 1998 Silicon microchambers for DNA amplification *Sensors Actuators* A **71** 81–8

[6] Senturia S D 2001 *Microsystem Design* (Boston: Kluwer)

[7] Batty W, Christoffersen C E, David S, Panks A J, Johnson R G, Snowden C M and Steer M B 2001 Fully physical time-dependent compact thermal modelling of complex non linear 3-dimensional systems for device and circuit level electro-thermal CAD *Semiconductor Thermal Measurement and Management, 2001. Seventeenth Annual IEEE* pp 71–84

[8] Yang Y-J, Gretillat M and Senturia S D 1997 Effect of air damping on the dynamics of nonuniform deformations of microstructures *Proc. 9th Int. Conf. on Solid-State Sensors and Actuators (Transducers'97) (Chicago, June 1997)* pp 1093–6

[9] Rudyni E B and Korvink J G 2002 *Review: Automatic Model Reduction for Transient Simulation of MEMS-based devices,* Sensors update v 11 pp 3–33

[10] Bechtold T, Salimbahrami B, Rudyni E B, Lohmann B and Korvink J G 2003 Krylov-subspace-based order reduction methods applied to generate compact thermo-electric models for MEMS *Technical Proc. 2003 Nanotechnology Conf. and Trade Show (MSM 2003) (San Francisco, 23–27 Feb. 2003)* vol 2

[11] Bechtold T, Rudnyi E B and Korvink J G 2003 Automatic generation of compact electro-thermal models for semiconductor devices *IEICE Trans. Electron.* **86** 459–65

[12] Yu C-C and Yang Y-J 2003 Compact heat transfer model generation for MEMS devices *Technical Proc. 2003 Nanotechnology Conf. and Trade Show (MSM 2003) (San Francisco, 23–27 Feb. 2003)* vol 2

[13] Yu C-C and Yang Y-J 2003 Extraction of heat transfer macromodels for MEMS devices *Proc. 12th Int. Conf. on Solid-State Sensors and Actuators (Transducers '03) (Boston, USA, June, 2003)* pp 1852–5

[14] Yang Y-J, Kamon M, Rabinovich V L, Ghaddar C, Deshpande M, Greiner K and Gilbert J R 2001 Modeling gas damping and spring phenomena in MEMS with frequency dependent macro-models *14th Int. Conf. on Micro Electro-mechanical Systems Workshop (MEMS 2001) (Interlaken, Switzerland, Jan. 2001) (Proc. IEEE)* pp 365–8

[15] Wang F and White J 1998 Automatic model order reduction of a microdevice using the Arnoldi approach *Proc. 1998 ASME Int. Mechanical Engineering Congress and Exposition: IMECE' 98 (Anaheim, CA, Nov. 1998)* (DSC vol 66) pp 527–30

[16] Odabasioglu A, Celik M and Pileggi L T 1998 PRIMA: passive reduced-order interconnect macromodeling algorithm *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **17** 645–54

[17] Mills A F 1992 *Heat Transfer* (Boston: Richard D Irwin Inc.)

[18] Holman J P 1989 *Heat Transfer* (Singapore: McGraw-Hill)

[19] Mankame N D and Ananthasuresh G K 2001 Comprehensive thermal modeling and characterization of an electro-thermal-compliant microactuator *J. Micromech. Microeng.* **11** 452–62

[20] Reddy J N and Gartling D K 1994 *The Finite Element Method in Heat Transfer and Fluid Dynamics* (Boca Raton, FL: CRC Press)

[21] Yang Y-J and Senturia S D 1996 Numerical simulation of compressible squeezed-film damping *Tech. Dig. of 1996 Solid-State Sensor and Actuator Workshop (Hilton Head, SC, June 1996)* pp 76–9

[22] Press W H, Flannery B P, Teukolsky S A and Vetterling W T 1993 *Numerical Recipes in C: The Art of Scientific Computing* 2nd edn (Cambridge: Cambridge University Press)

[23] Lott C D, McLain T W, Harb J N and Howell L L 2002 Modeling the thermal behavior of a surface-micromachined linear-displacement thermomechanical microactuator *Sensors Actuators* **101** 239–50

[24] CoventorWare2001 2001 *Reference Manual* (Cary, NC: Coventor, Inc.)

[25] *MUMPS Design Handbook* 2002 (Oakland, CA: MEMSCAP, Inc.)

[26] Bathe K-J 1996 *Finite Element Procedures* (London: Prentice-Hall)

[27] Huang Q A and Lee N K S 1999 Analysis and design of polysilicon thermal flexure actuator *J. Micromech. Microeng.* **9** 64–70

[28] Butler T J, Bright V M and Cowan W D 1999 Average power control and positioning of polysilicon thermal actuators *Sensors Actuators* A **72** 88–97

[29] Marco S, Carmona M and Samiter J 1998 Extraction of dynamic HDL-A models of thermally based microsystems from physical simulations *Modeling and Simulation of Microsystems, MSM'98 (Santa Clara, CA, USA)* pp 159–62

[30] Carmona M, Marco S, Sieiro J, Ruiz O, Go'mez-Cama J and Samitier J 1999 Modeling of microsystems with analog hardware description languages *Sensors Actuators* A **76** 32–42

[31] Weste N H E and Eshraghian K 1993 *Principles of CMOS VLSI Design, A System Perspective* 2nd edn (New York: Addison-Wesley)

[32] Tutorial for SMASH 5, Dolphin Integration, 2002