DE GRUYTER
OPEN

Communications in Applied
and Industrial Mathematics
ISSN 2038-0909

Research Article

Commun. Appl. Ind. Math. 8 (1), 2017, 67–80
DOI: 10.1515/caim-2017-0004

Integral equations for free-molecule flow in MEMS: recent advancements

Patrick Fedeli[1*], Attilio Frangi[1]

[1]Department of Civil and Environmental Engineering, Politecnico di Milano, Milan, Italy

[*]Email address for correspondence: patrick.fedeli@polimi.it

Communicated by Nicola Bellomo
Received on 12 20, 2016. Accepted on 03 21, 2017.

Abstract

We address a Boundary Integral Equation (BIE) approach for the analysis of gas dissipation in near-vacuum for Micro Electro Mechanical Systems (MEMS). Inspired by an analogy with the radiosity equation in computer graphics, we discuss an efficient way to compute the visible domain of integration. Moreover, we tackle the issue of near singular integrals by developing a set of analytical formulas for planar polyhedral domains. Finally a validation with experimental results taken from the literature is presented.

Keywords: `Boundary Integral Equations, Rarefied Gas Dynamics, MEMS`
AMS subject classification: `65R20`

1. Introduction

Microsystems are experiencing an important increase of commercial success but still pose several scientific challenges. MEMS typically consist of a collection of fixed parts and moving shuttles separated by variable gaps of few microns. One important issue in the design of these devices is represented by the accurate evaluation of mechanical dissipation. Indeed the quality factor Q, a key element in quantifying the performance of devices in industrial environments, is proportional to the ratio between the energy stored in the system and the dissipation.

Several MEMS applications like gyroscopes, resonators or magnetometers, contain parts in perpetual motion close to their mechanical resonance, so that it often makes sense to assume that the movement is proportional to the mechanical eigenmode. This permits to obtain a 1D reduced order model:

$$M\ddot{q}(t) + B\dot{q}(t) + Kq(t) = F(t), \tag{1}$$

where M is the generalised mass, B the damping coefficient and K the generalised stiffness. In terms of these parameters, the quality factor can be simply computed as

$$Q = \frac{2\pi f M}{B}. \tag{2}$$

While techniques for obtaining K and M are trivial, or anyway well established, the numerical estimation of B is a complicated task.

In order to reduce dissipation, these resonating MEMS are packaged in near-vacuum with a getter. Anyway, among the different sources, gas damping often provides a meaningful contribution. The length scale and the working pressure are such that the collisions between molecules can be neglected; this regime is known as free-molecule flow [1–6].

The deterministic model implemented, proposed in [7,8], is an application of the collisionless Boltzmann equation which rests on the following assumptions: i) the mean free-molecule path is much larger than the typical dimension d of the flow (i.e. Kn$\gg$1); ii) molecules and solid surfaces interact according the diffuse-reflection model which basically states that molecules are re-emitted by the walls according to the wall equilibrium Maxwellian; iii) perturbations are small so that quadratic terms in the expansion of variables can be neglected. While details of the derivation can be found in [8], only the final equations are presented herein. Let $J(\mathbf{x},t)$ denote the first order perturbation of the flux of incoming molecules at point $\mathbf{x}$ of the MEMS surface. This scalar unknown is governed by the following integral equation:

$$\begin{aligned} J(\mathbf{x},t) = &\sqrt{\pi}\tilde{g}_n(\mathbf{x},t) - \\ &\frac{2}{\pi}\int_{S^+} J(\mathbf{y},t)\big(\mathbf{r}\cdot\mathbf{n}(\mathbf{x})\big)\big(\mathbf{r}\cdot\mathbf{n}(\mathbf{y})\big)\frac{1}{r^4}T_3(i\tilde{\omega}r)\mathrm{d}S + \\ &\frac{4}{\pi}\int_{S^+} \big(\mathbf{r}\cdot\tilde{\mathbf{g}}(\mathbf{y},t)\big)\big(\mathbf{r}\cdot\mathbf{n}(\mathbf{x})\big)\big(\mathbf{r}\cdot\mathbf{n}(\mathbf{y})\big)\frac{1}{r^5}T_4(i\tilde{\omega}r)\mathrm{d}S, \end{aligned} \tag{3}$$

where $r = \|\mathbf{y}-\mathbf{x}\|$; $\mathbf{n}$ is the outward normal to the surface; S^+ denotes the portion of surface visible from $\mathbf{x}$; $\tilde{\omega} = \omega/\sqrt{2\mathcal{R}T_0}$ and $\tilde{\mathbf{g}}(\mathbf{y},t) = \mathbf{g}(\mathbf{y},t)/\sqrt{2\mathcal{R}T_0}$ are the normalised angular and linear velocity of surfaces ($\mathcal{R}$ is the universal gas constant divided by the molar mass and T_0 is the package temperature), $\tilde{g}_n = \tilde{\mathbf{g}}\cdot\mathbf{n}$ being its projection along $\mathbf{n}$; $T_n(i\tilde{\omega}r)$ is the transcendental function [9]

$$T_n(i\tilde{\omega}r) = \int_0^\infty u^n exp\left(-u^2 - \frac{i\tilde{\omega}r}{u}\right)\mathrm{d}u. \tag{4}$$

Moreover, if frequency f of perturbations is small with respect to thermal velocity, i.e. $d\times\tilde{\omega} \ll 1$, the integral terms can be further simplified using the identities $T_3(0) = 1/2$ and $T_4(0) = 3\sqrt{\pi}/8$. This corresponds to the quasi-static assumption which holds if the average collision period of a molecule is small with respect to the duration of one oscillation.

A posteriori a second integral equation provides the distribution of perturbation forces $\mathbf{t}$ on the structure:

$$
\begin{aligned}
(5)\quad -\frac{\mathbf{t}(\mathbf{x},t)}{\rho_0 2\mathcal{R}T_0} =& \frac{1}{4}J(\mathbf{x},t)\mathbf{n}(\mathbf{x}) + \frac{1}{\sqrt{\pi}}\tilde{g}_n(\mathbf{x},t)\mathbf{n}(\mathbf{x}) + \frac{1}{2\sqrt{\pi}}\tilde{\mathbf{g}}_t(\mathbf{x},t) - \\
& \frac{1}{\pi^{3/2}}\int_{S^+}\mathbf{r}\Big(\mathbf{r}\cdot\mathbf{n}(\mathbf{x})\Big)\Big(\mathbf{r}\cdot\mathbf{n}(\mathbf{y})\Big)\frac{1}{r^5}J(\mathbf{y},t)T_4(i\tilde{\omega}r)\mathrm{d}S + \\
& \frac{2}{\pi^{3/2}}\int_{S^+}\mathbf{r}\Big(\mathbf{r}\cdot\tilde{\mathbf{g}}(\mathbf{y},t)\Big)\Big(\mathbf{r}\cdot\mathbf{n}(\mathbf{x})\Big)\Big(\mathbf{r}\cdot\mathbf{n}(\mathbf{y})\Big)\frac{1}{r^6}T_5(i\tilde{\omega}r)\mathrm{d}S,
\end{aligned}
$$

where $\tilde{\mathbf{g}}_t = \tilde{\mathbf{g}} - \tilde{g}_n\mathbf{n}$. As for Equation (3), also Equation (5) simplifies in the quasi-static case since $T_4(0) = 3\sqrt{\pi}/8$ and $T_5(0) = 1$.

In Equations (3),(5) the velocity $\mathbf{g}$ of the deformable MEMS is assumed to be proportional to a given vector shape function: $\mathbf{g}(\mathbf{y},t) = \boldsymbol{\psi}(\mathbf{y})q(t)$. Hence one has that $\mathbf{t}(\mathbf{x},t) = \mathbf{f}(\mathbf{x})\dot{q}(t)$, where $\mathbf{f}(\mathbf{x})$ is directly provided by Equation (5). The equivalent damping term in the 1D reduced order model finally becomes:

$$
\int_S \mathbf{t}(\mathbf{x},t)\boldsymbol{\psi}(\mathbf{x})\mathrm{d}S = \Big(\int_S \mathbf{f}(\mathbf{x})\boldsymbol{\psi}(\mathbf{x})\mathrm{d}S\Big)\dot{q} = B\dot{q},
$$

where the integral is extended over the whole surface of the MEMS.

The constant B can be conveniently expressed as

$$
(6)\qquad B = \tilde{B}\rho_0\sqrt{2\mathcal{R}T_0} = \tilde{B}p_0\sqrt{2/(\mathcal{R}T_0)},
$$

where $\tilde{B}$ is a coefficient with the dimensions of a surface depending only on the problem geometry.

It is worth stressing that Equation (3) is very similar to the radiosity equation of Computer Graphics, which is a tool for the generation and manipulation of images on computer screens. One of the key elements is the presence of the visibility operator limiting the integration to the visible portion of surfaces. Basically, given a set of 3D objects and a viewing point, the objective is to determine which lines or surfaces of the objects are visible [10–12]. Inspired by this analogy, in the present implementation we propose an efficient way to compute the domain of integration adapting some typical techniques of Computer Graphics, where the viewing point corresponds to the source object. Moreover, when integrating over visible

"close" portions of the surface, issues associated to the singular nature of integral kernels require particular care. In this case we develop analytical formulas valid for triangles and piecewise constant representation of the unknown field. We discuss these issues in Section 2, while in Section 3 we present two benchmarks with experimental results taken from the literature.

2. Numerical implementation

Excluding specific cases where an analytical expression for $\mathbf{t}$ is available, Equations (3),(5) must be solved numerically.

Since due to technological constraints the vast majority of MEMS is composed by piecewise planar surfaces, the structure is initially represented as the collection of non-overlapping large planar "parent" quadrangles, as illustrated in Figure 1-a. Each quadrangle is then meshed with "children" triangles (Figure 1-b), typically using the free software GMSH [13]. Conformity between the different quadrangles is not required. This two-level geometrical representation considerably simplifies testing the visibility condition as discussed next. In the present implementation J is modeled as

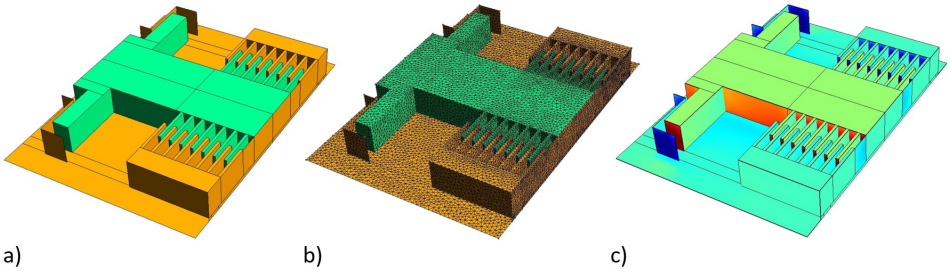

Figure 1. Example of comb-finger device, a typical MEMS structure: a) quadrangular discretization, b) triangular mesh with 62526 elements and c) force per unit surface in the direction of the shuttle velocity.

piecewise constant over each triangle; Equation (3) is then collocated at the center of each triangle and the system of equations is solved by an iterative GMRS (Generalized minimal residual method) solver. The final output of the simulation is the constant $\tilde{B}$ (see Equation (6)), obtained through a post-processing of the force $\mathbf{t}$ (Figure 1-c).

Recent improvements in the procedure for testing the visibility condition, the introduction of analytical formulas to compute the near-singular integrals and a large-scale OpenMP parallel implementation allow to simulate almost realistic MEMS structures on standard hardware. Figure 2 collects a convergence study performed on the test geometry depicted in Fig-

ure 1, which represents a portion of a realistic comb-finger MEMS [18]. The relative error is computed with respect to the "reference" numerical solution obtained on a fine mesh with 250000 elements.

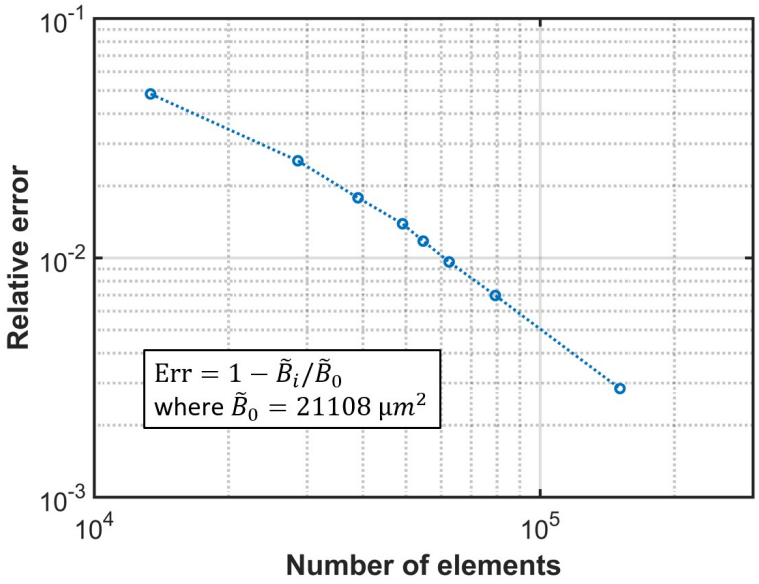

Figure 2. Convergence of the solution for different levels of mesh refinement.

2.1. *Testing the visibility condition*

Given a source point $\mathbf{x}$, the integral in Equations (3) and (5) must be limited to the portion S^+ of the surface which is visible from $\mathbf{x}$. A fast and reliable procedure for testing the visibility condition clearly represents one of the major issues in the numerical implementation of the method. A four-level algorithm has been devised. Each quadrangle (and its children) is endowed with an outward normal vector $\mathbf{n}$ defining the positive side exposed to collisions with molecules. Clearly, visibility is tested for the positive sides of the quadrangles (triangles) and this is left implicit in what follows.

Two quadrangles can be either *invisible*, or *partially visible*, or *totally visible* (*visible* for simplicity). One can conceive a symmetric visibility matrix for quadrangles. Its entries are: set to zero if two quadrangles are invisible; set to one if they are visible; left unspecified otherwise (partial visibility). Since the number of quadrangles is much smaller than the number of triangles, establishing the visibility between quadrangles has a limited cost, and visibility (or invisibility) transfers immediately to the children. On the contrary, if two parents are only partially visible, the condition has

to be tested for every child triangle at a much higher cost. An application of tree-structures to quadrangles, like in fast integral approaches, could greatly optimize this phase and is currently under investigation.

Level 1. This level rapidly identifies quadrangles which are clearly invisible. Considering a given source quadrangle, its plane divides the space in a positive and a negative half-spaces (according to its positive side) as illustrated in Figure 3-a. If a target quadrangle has the four vertices in the negative half-space, then the target is invisible from the source and both entries of the visibility matrix are filled with a zero. This procedure is often called "hidden surface removal".

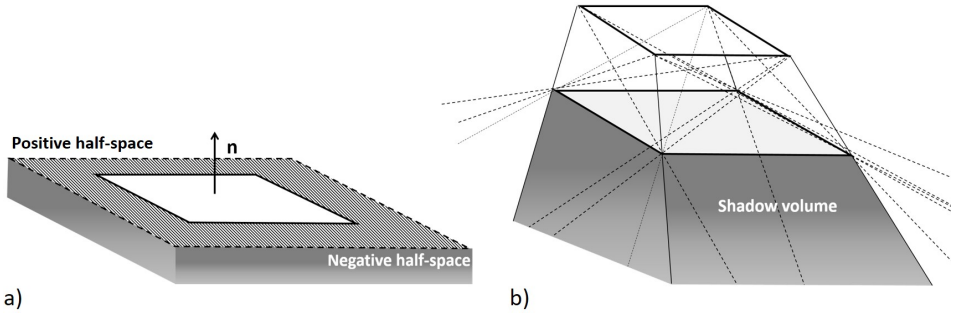

Figure 3. Algorithms for invisibility: a) Hidden Surface Removal, b) Shadow Volume.

Level 2. If two quadrangles have not been identified as invisible in Level 1, they are analysed in Level 2 using a shadow volume technique [14], often used in computer graphics to add shadows to a rendered scene. Let us consider a source quadrangle and a target quadrangle. The aim of this phase is to test if any other "obstacle" quadrangle can generate a shadow (Figure 3-b) making the target invisible from the source. The shadow volume is constructed projecting a ray from each vertex of the source through each vertex of the obstacle. These projections will together form a set of four "pyramids", one for each source vertex, and their intersection is the volume of total-shadow. If a target surface has all the vertices inside the total-shadow volume, then it is invisible to the source.

Level 3. If two quadrangles have not been identified as invisible in the first two phases, they are addressed in Level 3 to verify whether they are totally visible using an algorithm based on 3D Delaunay triangulation (or

tetrahedralization) [15]. Triangulation of a given set P of n points in 3-dimensional space is a decomposition into tetrahedra of the convex hull of P such that: i) vertices of tetrahedra belong to P; ii) intersection of two tetrahedra is either or a vertex or an edge or a face.

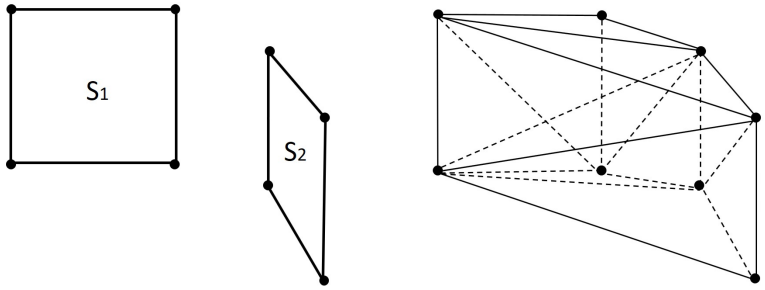

Figure 4. 3D Delaunay triangulation starting from source and target quadrangles.

Given a source quadrangle S_1 and a target quadrangle S_2, the set P is populated with their 8 vertices and the Delaunay triangulation is built, as depicted in Figure 4. If all the other quadrangles don't intersect the convex hull, then the source and target are fully visible.

Level 4. When two children belong to partially visible parents, a point-to-point test visibility based on an image-space approach has to be applied. Drastically simplifying reality, two triangles are said to be visible if the segment (a sort of ray of view) that connects the centers of mass of two element does not encounter obstacles. This assumption is totally acceptable if a sufficiently fine mesh is adopted (as reported in Figure 2).

2.2. *Analytic evaluation of integrals for polygonal elements*

The integral terms in Equations (3) and (5) are computed according to the following strategy. If the source point and target triangle are well-separated, all the integrals are regular and are evaluated using standard Gauss-Hammer quadrature rules with a variable number of points according to the distance between the target and the source. On the contrary, when the source and the target are close, all the integrals are evaluated by means of the analytical formulas developed herein. For a given element E with center of mass $\mathbf{y}$, a generalized pyramid is defined by connecting the source point $\mathbf{x}$ and the nodes of E. Moreover each edge e of the element and the source point $\mathbf{x}$ define a triangle of normal $\mathbf{N}$ and angle $\Delta\alpha$ at vertex $\mathbf{x}$ (see Figure 5).

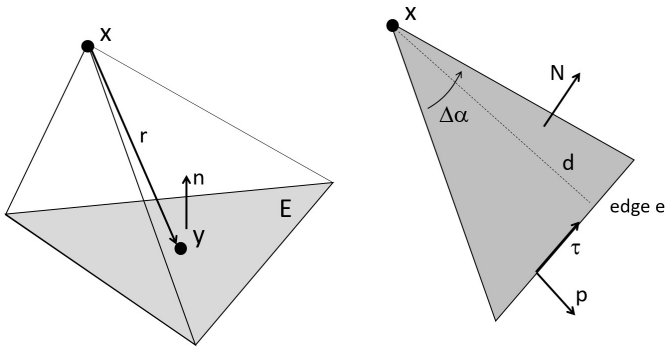

Figure 5. Generalised pyramid and notation adopted for the analytical integration procedure.

The first integral term of Equation (3) for element E becomes:

$$\int_E \big(\mathbf{r}\cdot\mathbf{n}(\mathbf{x})\big)\big(\mathbf{r}\cdot\mathbf{n}(\mathbf{y})\big)\frac{1}{r^4}\mathrm{d}S_y = n_i(\mathbf{x})\int_E \frac{r_i r_k}{r^4} n_k \mathrm{d}S_y \tag{7}$$

and

$$\int_E \frac{r_i r_k}{r^4} n_k \mathrm{d}S_y = \frac{1}{2}\sum_e \Delta\alpha N_i, \tag{8}$$

which is known as Lambert formula.

The second integral term of Equation (3), which is equal to the first one in Equation (5), for a constant $\tilde{\mathbf{g}}$ becomes:

$$\int_E \big(\mathbf{r}\cdot\tilde{\mathbf{g}}\big)\big(\mathbf{r}\cdot\mathbf{n}(\mathbf{x})\big)\big(\mathbf{r}\cdot\mathbf{n}(\mathbf{y})\big)\frac{1}{r^5}\mathrm{d}S_y = \tilde{g}_i n_j(\mathbf{x})\int_E \frac{r_i r_j r_k}{r^5} n_k \mathrm{d}S_y \tag{9}$$

and

$$3\int_E \frac{r_i r_j r_k}{r^5} n_k \mathrm{d}S_y = \sum_e dN_j \int_e \frac{r_i}{r^3}\,\mathrm{d}s - \delta_{ij}\Omega, \tag{10}$$

where Ω is the solid angle at the vertex $\mathbf{x}$ of the generalised pyramid. The integral term in the rhs of Equation (10) can be easily evaluated analytically. Indeed, with reference to Figure 5, if s is an abscissa running along edge e, with $s = 0$ at the minimum distance from $\mathbf{x}$, then $\mathbf{r} = d\mathbf{p} + s\boldsymbol{\tau}$ and $r = \sqrt{s^2 + d^2}$.

The second integral term of Equation (5) gives for a constant $\tilde{\mathbf{g}}$:

$$\int_E \mathbf{r}\Big(\mathbf{r}\cdot\tilde{\mathbf{g}}\Big)\Big(\mathbf{r}\cdot\mathbf{n}(\mathbf{x})\Big)\Big(\mathbf{r}\cdot\mathbf{n}(\mathbf{y})\Big)\frac{1}{r^6}\mathrm{d}S = \tilde{g}_i n_j(\mathbf{x})\mathbf{e}_m \int_E \frac{r_i r_j r_k r_m}{r^6} n_k \mathrm{d}S_y \tag{11}$$

and, using Lambert formula:

(12)
$$8\int_E \frac{r_i r_j r_k r_m}{r^6} n_k \mathrm{d}S_y = \sum_e \left[\Delta\alpha(N_j\delta_{im} + N_i\delta_{jm}) + 2N_m d\int_e \frac{r_i r_j}{r^4}\mathrm{d}s\right].$$

Finally, the integral term in the rhs of Equation (12) can be integrated as before introducing the abscissa s running along the edges.

2.3. *Parallel implementation*

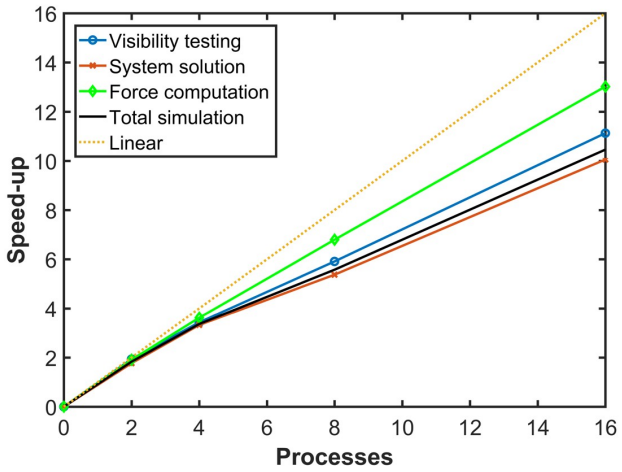

Figure 6. Speed-up of the three parallelized routines and for the total simulation obtained with the super-computer Galileo of Cineca.

A large-scale OpenMP parallel implementation [16] has been developed for three fundamental routines: point-to-point visibility test in Level 4 described in Section 2.1; iterative GMRES solution of the linear system obtained collocating Equation (3); force computation from Equation (5).

The speed-up, defined as the ratio between the wall-clock time for serial execution and the wall-clock time for parallel one, is used as an indicator of performance. In Figure 6 the speed-up obtained for the three parallel routines during the simulation of the example in Figure 1 is reported, showing a good scalability of the problem. Since the visibility test and the solution of the linear system are the most demanding parts of the code, the overall speed-up is similar to that of these routines.

3. Experimental validation

Validations of the proposed tool have been performed using several experimental data for MEMS described in the literature. Limiting our atten-

tion to the quasi static case, as discussed in Section 1, among the different options we consider two MEMS working in the free-molecule flow: a tilting resonator and a classical Tang resonator.

3.1. *Rotational resonator*

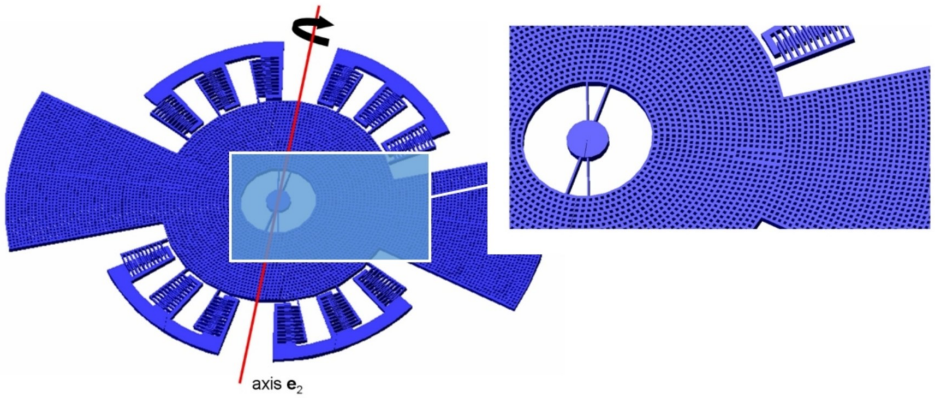

Figure 7. Tilting resonator from [7]: layout.

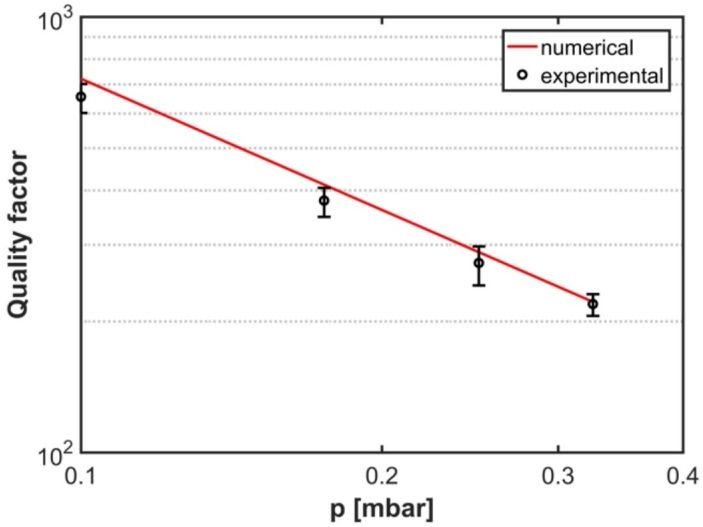

Figure 8. Tilting resonator from [7]: comparison between experimental and numerical quality factors.

The first structure (Figure 7) is the tilting perforated mass addressed in [7] with a small-scale implementation of the integral approach. The mass is attached to the central circular anchor by means of four deformable beams

and is suspended over a substrate. Electrodes placed on the substrate induce a tilting movement around the axis drawn in the picture. In this case only the contribution from holes in the mass has been considered, and the tilting motion is described by setting $\boldsymbol{\psi} = x\mathbf{e}_z$, where x denotes the distance from the rotation axis and z is the out-of-plane direction.

The integral approach provides a very good matching with the experimental data in the wide pressure range where free-molecule regime (Figure 8) is expected to develop.

3.2. *Tang resonator*

The Tang resonator of Figure 9 has been addressed in [17] using a simplified BGK model of the Boltzmann equation. The resonator consists of: two fixed arrays of fingers (stators) attached to the substrate on the upper and lower sides; a shuttle in the middle oscillating in the direction of the comb-fingers; springs connecting the shuttle to fixed blocks in order to constrain any motion along the other directions. Due to the quite unusual layout of springs, the overall flow is a combination of Couette type (in the comb-fingers and between the shuttle and the substrate) and Poiseuille type of flow (between the pairs of flexible springs). Moreover, springs are highly deformable and hence cannot be treated as rigid structures. Also in this case the agreement between the numerical prediction and the experimental data is remarkable (Figure 10).

4. Conclusions

We have discussed some numerical issues associated to a large-scale implementation of a Boundary Integral Equation approach proposed by the authors for the deterministic analysis of microstructures working in the free molecule flow.

We have focused first on techniques for assessing the visibility conditions based on a two level geometrical representation of the structure surface, consisting of large parent quadrangles and children triangles. An application of tree-structure techniques to quadrangles, like in fast integral approaches, could greatly optimize this phase and is currently under investigation. Next we have detailed the analytical integration formulas that we have developed to address near singular integrals. The application of multipole-expansions to the kernels of the BIE could further improve the overall performance.

Finally, we have benchmarked the proposed approach against experimental data taken from the literature confirming the expected accuracy of the formulation.

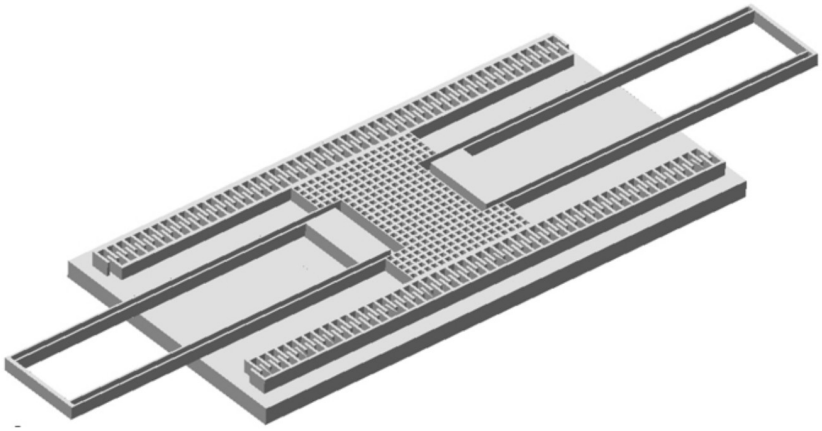

Figure 9. Tang resonator from [17]: layout.

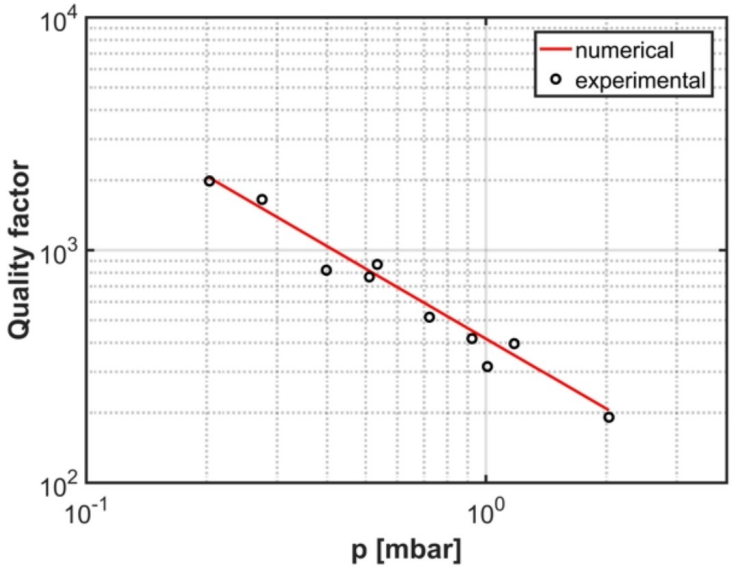

Figure 10. Tang resonator from [17]: comparison between experimental and numerical quality factors.

Acknowledgements

Funding from ST-Microelectronics is gratefully acknowledged.

REFERENCES

1. C. Cercignani, *The Boltzmann equation and its applications*, Springer, 1988
2. S. Chapman and T. Cowling, *The mathematical theory of non-uniform gases*, Cambridge Univeristy Press, 1960
3. G.A. Bird, *Molecular gas dynamics and the direct simulation of gas flows*, Clarendon Press, 1994
4. M. Gad-el-Hak, The fluid mechanics of microdevices - the Freeman scholar lecture, *J. Fluids Eng.*, vol. 121, pp. 5–33, 1999
5. G:E. Karniadakis and A. Beskok, *Micro flows, fundamentals and simulation*, Springer, 2002
6. M.M.R. Williams, A review of the rarefied gas dynamics theory associated with some classical problems in flow and heat transfer, *Z. Angew. Math. Phys.*, vol. 52, pp. 500–516, 2001
7. A. Frangi, A. Ghisi, L. Coronato, On a deterministic approach for the evaluation of gas damping in inertial MEMS in the free-molecule regime, *Sensor & Actuators A*, vol. 49, pp. 21–28, 2009
8. A. Frangi, BEM technique for free-molecule flows in high frequency MEMS resonators, *Engineering Analysis with Boundary Elements*, vol. 33, pp. 493–498, 2009
9. M. Abramowitz and I. Stegun, *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, Dover, 1964
10. J. Hughes, A. Van Dam, M. Mcguire, D. Sklar, J. Foley, S. Feiner and K. Akeley, *Computer Graphics: principles and practice (3rd edition)*, Addison-Wesley, 2013
11. J. Bittner and P. Wonka, Visibility in computer graphics, *Environment and Planning B: Planning and Design*, vol. 30, pp. 729–755, 2003
12. S. Kumar, D. Manocha, W. Garrett and M. Lin, Hierarchical back-face computation, *Computer and Graphics*, vol. 23, pp. 681–692, 1999
13. C. Geuzaine, J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, *International Journal for Numerical Methods in Engineering*, vol. 79, pp. 1309–1331, 2009
14. P. Bergeron, A general version of Crows shadow volumes, *IEEE Computer Graphics and Applications*, vol. 6, no. 9, pp. 17–28, 1986
15. S. Idelsohn, N. Calvo and E. Oñate, Polyhedrization of an arbitrary 3D point set, *Comput.Methods Appl. Mech. Engrg.*, vol. 192, pp. 2649–2667, 2003
16. P. Pacheco, *An introduction to parallel programming*, Morgan Kaufamann, 2011

17. C. Cercignani, A. Frangi, S. Lorenzani, B. Vigna, BEM approaches and simplified kinetic models for the analysis of damping in deformable MEMS, *Engineering Analysis with Boundary Elements*, vol. 31, pp. 451–457, 2007

18. A. Frangi, P. Fedeli, G. Laghi, G. Langfelder and G. Gattere, Near vacuum gas damping in MEMS: numerical modeling and experimental validation, *IEEE JMEMS*, vol. 25, no. 5, pp. 890–899, 2016