

A cut cell method for the 3D simulation of Crookes radiometer

Guillaume Dechristé and Luc Mieussens

*Univ. Bordeaux, IMB, UMR 5251, F-33400 Talence, France.
CNRS, IMB, UMR 5251, F-33400 Talence, France.*

Abstract. Devices involved in engineering applications, such as vacuum pumps or MEMS, may be made of several moving parts. This raises the issue of the simulation of rarefied gas flow around moving boundaries. We propose a simple process, known as cut cell method, to treat the motion of a solid body in the framework of the deterministic solving of a kinetic equation. Up to our knowledge, this is the first time that this approach has been used for this kind of simulations. The method is illustrated by the 2D and 3D simulations of a Crookes radiometer.

Keywords: BGK equation, cut cell method, moving boundary

PACS: 47.45.-n, 47.11.-j

INTRODUCTION

Current engineering developments make possible the conception of more and more complex nano-micro devices. The simulation of rarefied gas micro flows in these devices becomes a challenging issue when they are made of several moving parts. Their size may induce specific phenomena, like the radiometric effect [1]. This classical phenomenon can also be observed in the famous Crookes radiometer [2]. It is characterised by a force exerted on a thin obstacle immersed in a rarefied gas, and caused by temperature difference between the sides of the obstacle. Previous studies already investigate the properties of the force in different configurations [3, 4, 5, 6]. In this work we are interested in the transient motion of plate resulting from this force because it is a perfect example of rarefied gas flow acting on solid moving structures.

Tacking into account boundary motion within this framework is mainly treated by the use of Lagrangian [7] and semi-Lagrangian [8] scheme for one dimensional problems. For higher dimensional problems, immersed boundary methods [9, 10] and moving mesh approach [11] have been proposed. However, the former class of method is not conservative and the latter one could need costly remeshing algorithms in case of complex shaped solid objects. Another way for the computation of incompressible viscous flows with moving boundaries is the cut cell method [12]. This method consists in building an unstructured mesh, based on a primary Cartesian grid by cutting the cells that are crossed by the boundary. The main advantages of the approach is that it avoids remeshing strategy. Moreover the numerical scheme is fully conservative. In this paper, the method is used to design a deterministic solver for rarefied gas flows and is illustrated by the simulation of the Crookes radiometer. To decrease the computational resources required by the 3D simulation, an adaptive mesh refinement (AMR) technique is implemented to refine the Cartesian grid in the region close to the boundary.

DESCRIPTION OF THE FLOW

The Crookes radiometer is made of a spherical glass bulb containing four rectangular flat vanes. Each left side of these vanes is black, while their right sides are white. Once put in the light, the black sides become hotter than the white sides. Because of this temperature difference, the torque exerted from the gas to the vanes increases and the vanes are rotating. In this paper, the length of the diagonal of the vanes is denoted L . Their thickness is $L/10$ and the radius of the main sphere is $2L$. Let its center be the origin of the coordinate system : then, the center of the four vanes are respectively set to $(0.75L, 0, 0)$, $(0, 0.75L, 0)$, $(-0.75L, 0, 0)$ and $(0, -0.75L, 0)$. The temperature of the white side of the vanes is assumed to be the same as that of the sphere, and is denoted by T_0 . The black side of a vane is twice as hot as its white side. Both 2D and 3D simulations are investigated. The 2D problem consists in the slice of the 3D problem at z -coordinate equal to zero, see figure 1. Note that the transient motion of a similar two dimensional problem has

already been studied with a moving mesh approach [6]. In this paper, the 2D simulations are proceeded as a basis for 3D computations.

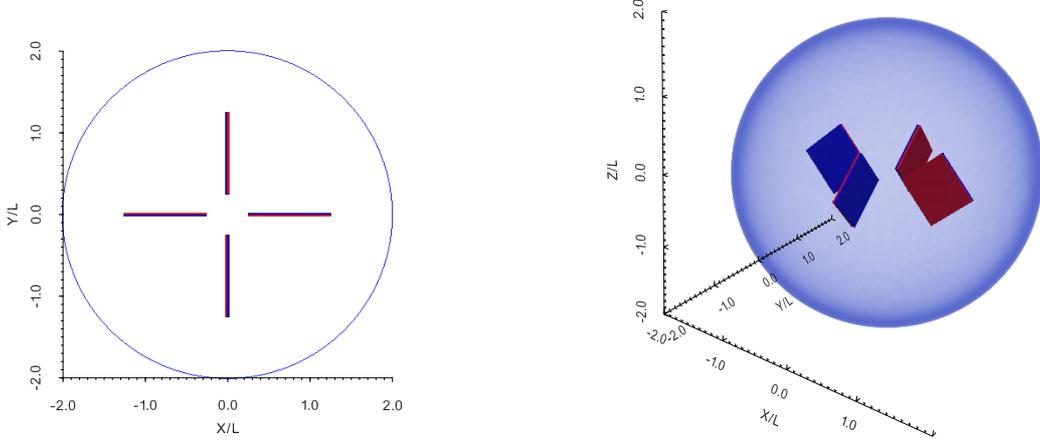


FIGURE 1. 2D and 3D shapes of the Crookes radiometer.

GOVERNING EQUATIONS

The gas surrounding the four vanes is governed by the BGK equation [13] :

$$\frac{\partial f}{\partial t} + \vec{v} \cdot \nabla f = \frac{1}{\tau} (\mathcal{M}[\rho, \vec{u}, T] - f), \quad \mathcal{M}[\rho, \vec{u}, T] = \frac{\rho}{(2\pi RT)^{3/2}} \exp\left(-\frac{\|\vec{v} - \vec{u}\|^2}{2RT}\right), \quad (1a)$$

$$\rho = \int_{\mathbb{R}^3} f d\vec{v}, \quad \rho \vec{u} = \int_{\mathbb{R}^3} \vec{v} f d\vec{v}, \quad 3\rho RT = \int_{\mathbb{R}^3} \|\vec{v} - \vec{u}\|^2 f d\vec{v}, \quad (1b)$$

where f and \vec{v} are the distribution function and the microscopic velocity, while ρ , \vec{u} and T are the density, macroscopic velocity and temperature of the gas. The variable R is ratio between the Boltzmann constant and the mass of a gas particle and the relaxation time τ relies on the Knudsen number $\text{Kn} = \lambda/L$, where the mean free path is computed by the relation $\lambda = (2/\sqrt{\pi})(\rho/\rho_0)\sqrt{2RT_0}\tau$, with ρ_0 setting for the initial density. Equation (1a) is more convenient for our purpose when integrated over a volume V which may be moving with an arbitrarily velocity. Denoting the surface of V by S and the outward normal on S by \vec{n} , this equation reads :

$$\frac{\partial}{\partial t} \int_V f dV + \int_S (\vec{v} - \vec{w}) \cdot \vec{n} f dS = \int_V \frac{1}{\tau} (\mathcal{M}[\rho, \vec{u}, T] - f) dV. \quad (2)$$

In this expression, \vec{w} stands for the velocity of the surface S . This velocity comes from the integration of the time derivative with the Reynolds transport theorem.

The boundary conditions are diffuse reflections and are computed in a standard way :

$$f(\vec{x} \in \partial\Omega, \vec{v} \in \mathcal{V}_{\text{in}}) = \mathcal{M}[\rho_w, \vec{u}_w, T_w], \quad \rho_w = - \left(\int_{\mathcal{V}_{\text{out}}} (\vec{v} - \vec{u}_w) \cdot \vec{n} f d\vec{v} \right) / \left(\int_{\mathcal{V}_{\text{in}}} (\vec{v} - \vec{u}_w) \cdot \vec{n} \mathcal{M}[1, \vec{u}_w, T_w] d\vec{v} \right), \quad (3)$$

where $\partial\Omega$ is the boundary of the vanes. The normal vector \vec{n} to the boundary at coordinate \vec{x} is pointed to the wall and the incoming and outgoing velocities are defined by $\mathcal{V}_{\text{in}} = \{\vec{v} \in \mathbb{R}^3 | (\vec{v} - \vec{u}_w) \cdot \vec{n} < 0\}$ and $\mathcal{V}_{\text{out}} = \mathbb{R}^3 / \mathcal{V}_{\text{in}}$. The temperature T_w of the wall is either T_0 or $2T_0$ depending on which side of the vanes is located \vec{x} . Note that the boundary condition strongly depends on the velocity \vec{u}_w of the point \vec{x} . It is related to its the angular velocity $\dot{\theta}$ of the vane, that is $\vec{u}_w = r\dot{\theta}(\sin\theta, \cos\theta)$, where r is the distance between \vec{x} and its orthogonal projection on the rotational axis.

To close the system of equations, the angular velocity is computed from the conservation of the angular momentum, which reads :

$$J_{\Delta} \frac{d\hat{\theta}}{dt} = \left(\int_{\partial\Omega} r \bar{\bar{\sigma}} \vec{n} \right) \cdot (0, 0, 1) \quad (4)$$

$$\bar{\bar{\sigma}} = \int_{\mathcal{Y}_{\text{out}}} (\vec{v} - \vec{u}_w) \otimes (\vec{v} - \vec{u}_w) f \, d\vec{v} + \int_{\mathcal{Y}_{\text{in}}} (\vec{v} - \vec{u}_w) \otimes (\vec{v} - \vec{u}_w) \mathcal{M}[\rho_w, \vec{u}_w, T_w] \, d\vec{v},$$

where the right hand side of the first equation is the torque and J_{Δ} is the moment of inertia of the vanes. It is set to $\rho_0 L^{d+2} \int_{\Omega} (\frac{r}{L})^2 \, d\vec{x}$, where d is the spatial dimension of the flow (two or three). This means that the density of the vanes is the same as the initial density of the gas. We empathize that this choice is not necessarily relevant from a physical point of view, but this feature only plays a role during the transient flow. The moment of inertia for the 3D computations differs from that for the 2D ones because the shape of the vanes depends on the dimension of the flow : the integral $\int_{\Omega} (\frac{r}{L})^2 \, d\vec{x}$ is equal to 97/750 for the 3D simulations, and to 97/397 for the 2D ones. In (4), $\bar{\bar{\sigma}}$ denotes the stress tensor at the solid surface.

CUT CELL METHOD

First, the velocity space is discretized with a discrete velocity method on a Cartesian grid. Let N_v be the number of discrete velocities in each direction. Then for all $k \in [0, N_v^3 - 1]$, there is a single triplet $(k_1, k_2, k_3) \in [0, N_v - 1]^3$ such that $k = k_3 \times N_v^2 + k_2 \times N_v + k_1$, and the k^{th} discrete velocity is denoted $\vec{v}_k = (v_{k_1}, v_{k_2}, v_{k_3}) = \vec{v}_0 + (k_1, k_2, k_3)\Delta v$, where Δv is the velocity step.

Then, the cut cell method is applied for the space discretization. This method is based on a finite volume approach on a fixed Cartesian grid of cell C_i . Because mesh does not fit with the physical boundary, a special treatment has to be applied on cells that are cut by the boundary. Now, consider a cuboid cell C_i which is sliced by the boundary. It is assumed that this cell is small enough so that its intersection with the boundary can be approximated by a single plane. A virtual cell $\bar{C}_i(t)$ is associated to the cuboid cell and correspond to the part of C_i that contains gas. This virtual cell is therefore a polyhedra that has at most 7 faces, and the surface of each face is are denoted by S_p . The index p goes from 1 to 7 : between 1 and 6, it corresponds respectively to the right, back, top, left, front and bottom faces that fit with the cuboid cell; the index $p = 7$ refers to the face of the virtual cell that fit with the boundary. For a better understanding, see figure 2. We empathize that since the boundary is moving, the virtual cell depends on time.

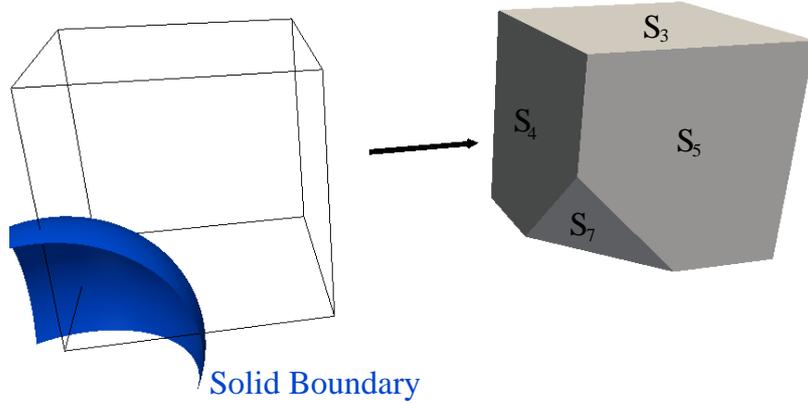


FIGURE 2. The cuboid cell C_i is plotted on the left with straight lines. It is cut by the blue boundary. The virtual cell \bar{C}_i associated to C_i is represented on the right with the index of several faces.

Finally, the time step is denoted Δt and $t^n = n \times \Delta t$. We introduce the average value of the distribution function $f_{i,k}^n$ and the macroscopic quantities (1b) are estimated by replacing the integral with respect to the velocity with a summation over the wall range of discrete velocities :

$$f_{i,k}^n = \frac{1}{V_i^n} \int_{\bar{C}_i(t^n)} f(t^n, \vec{x}, \vec{v}_k) \, dV, \quad \rho_i^n = \sum_{k=0}^{N_v^3-1} f_{i,k}^n \Delta v^3, \quad \rho_i^n \vec{u}_i^n = \sum_{k=0}^{N_v^3-1} \vec{v}_k f_{i,k}^n \Delta v^3, \quad 3\rho_i^n RT_i^n = \sum_{k=0}^{N_v^3-1} \|\vec{v}_k - \vec{u}_i^n\|^2 f_{i,k}^n \Delta v^3.$$

In the definition of the average value of the distribution function, V_i^n denotes the volume of $\bar{C}_i(t^n)$. The integrated BGK equation (2) is solved for every discrete velocity, and the time derivative is approximated with a standard backward Euler scheme. The set of N_v^3 equations hence reads :

$$\frac{1}{\Delta t}(V_i^{n+1} f_{i,k}^{n+1} - V_i^n f_{i,k}^n) + \sum_{p=1}^7 \mathcal{F}_{i,k}^n(S_p) = \frac{1}{\tau_i^n} (M_k[\rho_i^n, \vec{u}_i^n, T_i^n] - f_{i,k}^n), \quad (5)$$

where $\mathcal{F}_{i,k}^n(S_p) = \int_{S_p} (\vec{v} - \vec{w}) \cdot \vec{n} f_k^n(\vec{x}) dS$ is the numerical flux across the face p , and \vec{w} the velocity of this face. The discrete Maxwellian M_k is computed following the procedure described in [14]. To compute the fluxes, note that $\vec{w} \cdot \vec{n}$ is zero for the six faces of the virtual cell that fit with the initial cuboid cell ($p \leq 6$) and that \vec{w} is close to the velocity \vec{u}_w of the boundary for the last face ($p = 7$). Therefore, a standard upwind scheme leads to

$$\mathcal{F}_{i,k}^n(S_p) := \begin{cases} |S_p| \times (\max(0, v_{k_p}) f_{i,k}^n + \min(0, v_{k_p}) f_{i_{p,k}}^n) & \text{for } p \leq 3, \\ |S_p| \times (\max(0, v_{k_{p-3}}) f_{i_{p,k}}^n + \min(0, v_{k_{p-3}}) f_{i,k}^n) & \text{for } p \leq 6, \\ |S_p| \times (\max(0, (\vec{v}_k - \vec{u}_w^n) \cdot \vec{n}) f_{i,k}^n + \min(0, (\vec{v}_k - \vec{u}_w^n) \cdot \vec{n}) M_k[\rho_w^n, \vec{u}_w^n, T_w^n]) & \text{for } p = 7. \end{cases} \quad (6)$$

In these expressions, i_p stand for the index of the neighbouring cells of C_i , $|S_p|$ is the area of S_p and $M_k[\rho_w, \vec{u}_w, T_w]$ is the boundary condition, where ρ_w is computed with (3). Note that for faces 1 to 6, the normal the the face is either $(\pm 1, 0, 0)$, $(0, \pm 1, 0)$ or $(0, 0, \pm 1)$, hence the normal velocity is $\pm v_{k_1}$, $\pm v_{k_2}$, $\pm v_{k_3}$. This is taken into account by the compact notation v_{k_p} for $p \leq 3$ and $v_{k_{p-3}}$ for $4 \leq p \leq 6$. The velocity of the wall is also computed with a backward Euler scheme, and eq. (4) yields

$$J_\Delta \frac{1}{\Delta t} (\dot{\theta}^{n+1} - \dot{\theta}^n) = \left(\sum_i |S_7| \bar{\vec{\sigma}}_i^n \vec{n}_i \right) \cdot (0, 0, 1)$$

$$\bar{\vec{\sigma}}_i^n = \sum_{\vec{v}_k \in \mathcal{V}_{\text{out}}} (\vec{v}_k - \vec{u}_w^n) \otimes (\vec{v}_k - \vec{u}_w^n) f_{i,k}^n \Delta v^3 + \sum_{\vec{v}_k \in \mathcal{V}_{\text{in}}} (\vec{v}_k - \vec{u}_w^n) \otimes (\vec{v}_k - \vec{u}_w^n) M_k[\rho_w^n, \vec{u}_w^n, T_w^n] \Delta v^3.$$

Note that the global process may induced the generation of very small virtual cells. This leads to very restrictive CFL condition. In order to avoid this problem, virtual cells $\bar{C}_i(t^n)$ which volumes are lower than the half of a cuboid cell are merged with a neighbouring virtual cell $\bar{C}_{i'}$. The control volume for the scheme (5) is in this case defined by the union $\bar{C}_i(t) \cup \bar{C}_{i'}$. This merging procedure also enables the treatment of appearing/disappearing virtual cell. For example, consider a cell C_i that is cut by the boundary at t^n and completely immersed in the solid at the next time step. Despite $\bar{C}_i(t^{n+1})$ is empty, the control volume for this time step is not empty since $\bar{C}_i(t)$ is merged. Therefore, the quantity contained in $\bar{C}_i(t^n)$ is automatically transferred to the merged cell.

The cut cell method can be applied to 2D computations. In that case, virtual cells are no more polyhedra but polygons. Their volumes are now area and their 7 faces are replaced by 5 edges. For 3D simulations as well as for 2D ones, the intersection points between the boundary and the edges of a cuboid cell are computed thanks to a level-set function stored in the vertex of the cell. From this intersection points, volume, surfaces and normal are computed with Green formula. To end this section, we summarize the different steps of the computation :

- Initialisation :
 - Time : $n = 0$ and $t^0 = 0$
 - Vane parameters : $\theta^0 = 0$ and $\dot{\theta}^0 = 0$
 - Distribution function : $f_{i,k}^0 = M_k[\rho_0, \vec{u}_0, T_0]$
- Do while $t^n < t_{\text{max}}$:
 - Virtual cell parameters : V_i^n , S_p and $\vec{u}_w^n = r \dot{\theta}^n (\sin \theta^n, \cos \theta^n)$
 - Computation of V_i^{n+1} after motion of the vanes : $\theta^{n+1} = \theta^n + \Delta t \dot{\theta}^n$,
 - Scheme in three steps :
 - * Boundary condition : $M_k[\rho_w^n, \vec{u}_w^n, T_w^n]$
 - * Flux $\mathcal{F}_{i,k}^n(S_p)$: eq. (6)
 - * Distribution function $f_{i,k}^{n+1}$: eq. (5)
 - Update of the angular velocity $\dot{\theta}^{n+1}$: eq. (4)
 - $t^{n+1} = t^n + \Delta t$ and $n = n + 1$

NUMERICAL RESULTS

An efficient parallel code has been implemented in order to make expensive computations. Both space and velocity meshes are treated with parallel algorithms. To give an idea, 2D simulations are launch on less than 64 processors while up to 200 of them are used for the 3D computations.

First consider 2D simulations. The reference solution is obtained on a very thin mesh, that is 500^2 cells in space and 50^3 velocity points. The angular velocity $\hat{\theta}(t)$ of the vanes is plotted in figure 3 for three different Knudsen numbers. As time goes on, this velocity tends to a finite stationary angular velocity. And as expected, at equilibrium, the higher is the Knudsen number, the faster move the vanes. When we look closer to the transient motion, it may be surprising to see that after a rapid increase, the velocity suddenly decreases before increasing again. This might be explain by the temperature discontinuity at initial time along the hot side of the vanes. The same phenomena is observed in [6].

If 2D computations are quite feasible with relatively fine discretization, the same requirement cannot be fulfilled in 3D because of the computational cost (time and memory). In order to make 3D computations, we are looking at the coarser 2D mesh that recovers the reference solution with sufficient accuracy . Then, by adding the same number of cell in the third dimension than in the two others, the 3D simulation should be converged. It is found that with 200^2 cells in space and 21^3 velocity points, the difference between the stationary velocity and the reference one is less than 2%. For coarser mesh, the length of cell edges is too long as compared to the thickness of the vanes to get accurate results. Adding 200 cells in the third dimension would lead to too prohibitively expensive 3D computations. To tackle this problem, AMR procedure based on quadtree and octree algorithm is implemented. The refinement criterion is only based on the distance between the cell and the nearest vane. This means than the mesh will be thinner in the vicinity of the vanes than far from it. The refinement seems sufficient because the main part of the flow is located near the four vanes. The cut cell method is well adapted to the quadtree/octree structure because every cells are rectangular/cuboid. Therefore, the scheme presented before is still valid. For a primary mesh made of 50^2 cells and two levels of refinement (figure 3) the relative error of the final angular velocity is less than 5%. The equivalent 3D mesh contains approximatively 2.4×10^5 which much less than the initial 3D cartesian grid of $200^3 = 8 \times 10^6$ cells. The gain factor is more than 30 which is really significant and 3D computations are now possible. Note that even if it is too costly to check the convergence, it is hoped that the error is of the same order of magnitude than in 2D.

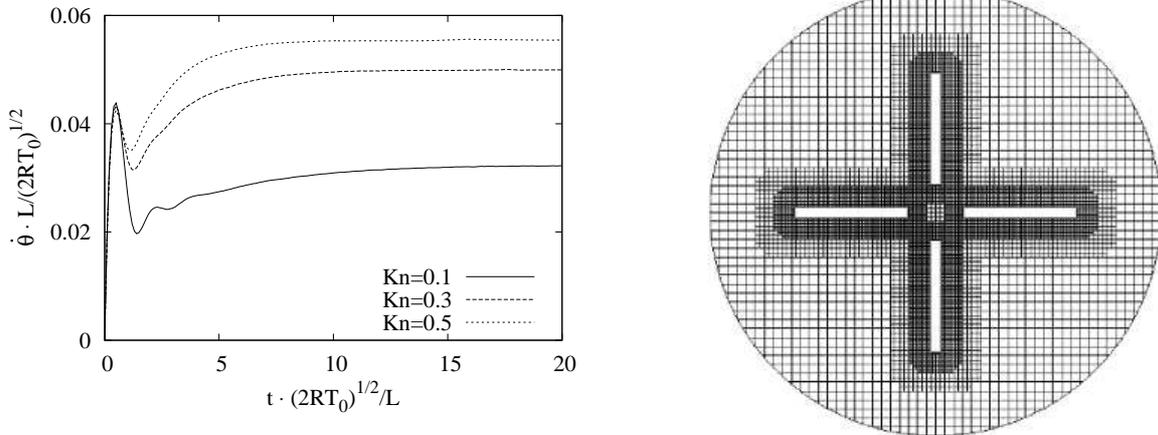


FIGURE 3. Results for the 2D computations. On the left is plotted the angular velocity of the vanes for three different Knudsen number. On the right, the computational mesh generated by the AMR procedure is represented.

For the 3D computations, the angular velocity $\hat{\theta}(t)$ of the vanes is plotted in figure 4 for the same three Knudsen numbers as before. The transient motion is similar to the 2D simulations, but the magnitude of the velocity is higher than the previous simulations. They are summarized in figure 4 for three different Knudsen numbers. Finally, two pictures of the flow are illustrated in figure 5 at distinct times.

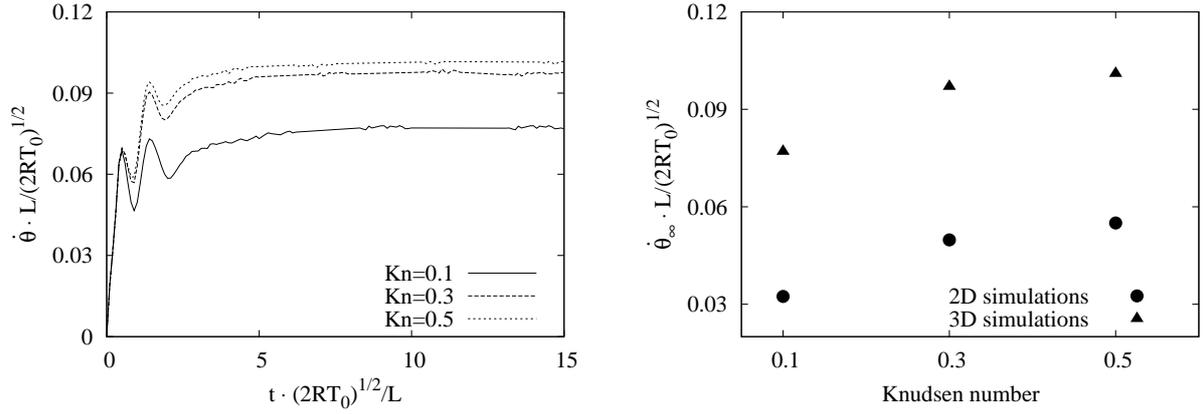


FIGURE 4. On the left is plotted the angular velocity of the vanes in function of time for the 3D computations. On the right, stationary angular velocity obtained with 2D simulations and the one obtained with 3D simulations are compared.

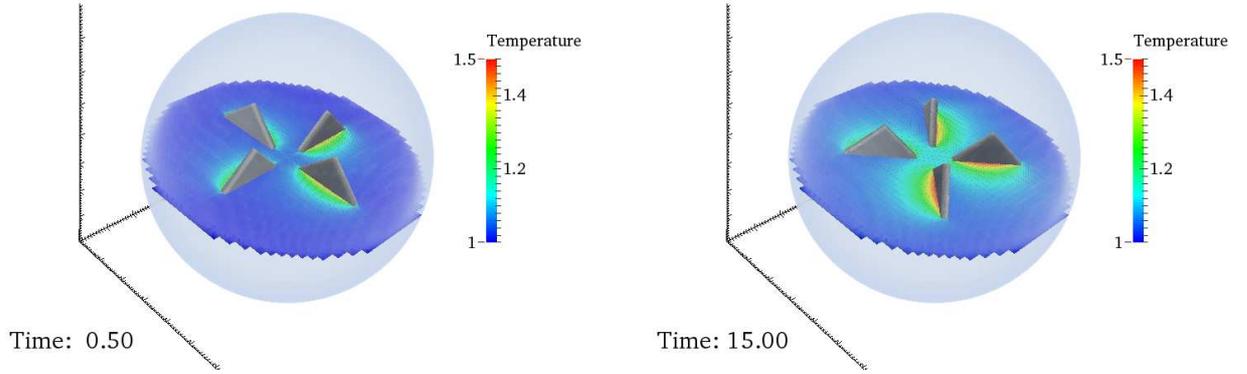


FIGURE 5. 3D radiometer : temperature field for two distinct times.

CONCLUSION

We have presented a numerical method for solving rarefied gas flow around moving obstacles. This method is an immersed boundary technique on Cartesian grid based on the cut cell approach and a finite volume scheme. Like every immersed boundary technique, it is easy to use with moving obstacles, but the cut cell and finite volume approach lead to a scheme which is fully conservative. Our method can be adapted to an AMR structure to gain efficiency in 3D computations, and is designed to work efficiently on parallel computers. This method works both for 2D and 3D flows, and has been used to simulate a 3D Crookes radiometer: up to our knowledge, such simulation was never presented before in the literature.

ACKNOWLEDGMENTS

This study has been carried out in the frame of 'the Investments for the future' Programme IdEx Bordeaux - CPU (ANR-10-IDEX-03-02).

Experiments presented in this paper were carried out using the PLAFRIM experimental testbed, being developed under the Inria PlaFRIM development action with support from LABRI and IMB and other entities: Conseil Régional d'Aquitaine, FeDER, Université de Bordeaux and CNRS (see <https://plafirim.bordeaux.inria.fr/>).

Computer time for this study was provided by the computing facilities MCIA (Mésocentre de Calcul Intensif Aquitain) of the Université de Bordeaux and of the Université de Pau et des Pays de l'Adour

REFERENCES

1. A. Passien, A. Wig, F. Meriaudeau, T. Ferell, and T. Thunda, *J. Appl. Phys.* **92**, 6326 (2002).
2. W. Crookes, *Philos. Trans. R Soc. London* **164**, 501–527 (1874).
3. S. Taguchi, and K. Aoki, *J. Fluid Mech.* **694**, 191–224 (2012).
4. S. Taguchi, and K. Aoki, “A Simple Model for Flows around Moving Vanes in Crookes Radiometer,” in *28th International Symposium on Rarefied Gas Dynamics 2012*, 2012, pp. 786–793.
5. Y. A. Anikin, *Comput. Math. Math. Phys.* **51**, 1923–1932 (2011).
6. S. Chen, K. Xu, and C. Lee, *Phys. Fluids* **24**, 111701 (2012).
7. T. Tsuji, and K. Aoki, *J. Comput. Phys.* **250**, 574–600 (2013).
8. G. Russo, and F. Filbet, *Kinetic and related models 2* **1**, 231–250 (2009).
9. C. Perkardan, S. Chigullapalli, L. Sun, and A. Alexeenko, “Immersed Boundary Method for Boltzmann Model Kinetic Equations,” in *28th International Symposium on Rarefied Gas Dynamics 2012*, 2012, pp. 358–365.
10. F. Filbet, and C. Yang, *J. Comput. Phys.* **245**, 43–61 (2013).
11. C. Jin, K. Xu, and S. Chen, *Adv. Appl. Math. Mech.* **Vol. 2, No.6**, 746–762 (2010).
12. D. M. Ingram, D. M. Causon, and C. G. Mingham, *Mathematics and Computers in Simulation* **61**, 561–572 (2003).
13. P. L. Bhatnagar, E. P. Gross, and M. Krook, *Phys. Rev.* **94**, 511–525 (1954).
14. L. Mieussens, *J. Comput. Phys.* **162**, 429–466 (2000).