Contemporary Mathematics
Volume 157, 1994

Coupling Boltzmann and Euler Equations without Overlapping

J.F. BOURGAT, P. LE TALLEC, B. PERTHAME, Y. QIU

Abstract

We present herein a method for coupling the Euler and the Boltzmann equations when these equations are used on two different, possibly nonoverlapping subdomains. This strategy is based on a kinetic writing of the Euler equations and on a pointwise matching of the associated kinetic half-fluxes. A time-marching algorithm is then used to solve the coupled problem. This approach is tested numerically on different two-dimensional configurations involving rarefied flows at low Knudsen numbers.

1 Introduction

The most accurate model for predicting rarefied flows around obstacles flying at high speed and high altitude is based on the Boltzmann equations. Unfortunately, when the mean free path gets small, the numerical solution of these equations becomes impossible : indeed, the grid which is used must have a discretization step smaller than the mean free path. The classical solution consists then in replacing the Boltzmann equations by the limit obtained when the mean free path goes to zero, that is either by the Euler equations (at order 1) or by the Navier Stokes equations (at order 2).

But, when the mean free path is roughly one thousand times smaller than the length of the obstacle, these limiting models are not complety

*1991 Mathematics Subject Classification. 65N20, 65F10, 65W05

†Supported by the Hermes Research Program

‡This paper is in final form and no version of it will be submitted for publication elsewhere

0271-4132/94 $1.00 + $.25 per page

valid, especially next to the obstacle. In such situations, one may use the Euler (or Navier-Stokes) equations far from the body but one must use the Boltzmann equations in the vicinity of the body. The objective of this work is to develop a numerical method for the simultaneous solution of the Euler and the Boltzmann equations when these equations are used together but on two different domains. For this purpose, we first introduce adequate compatibility conditions to be imposed at the interfaces, and we then propose an algorithm for the numerical solution of the resulting coupled problem.

The compatibility conditions are based on a kinetic interpretation of the Euler equations [3]. With this interpretation, the unique unknown in both domains is the kinetic distribution $f(x, v, t)$ of particles which are at time t in point x with velocity v. This distribution satisfies an equation which looks like a linear transport equation. Therefore, on each domain, we only need to impose as boundary conditions the kinetic distribution of the incoming particles. For the Euler domain, this distribution is taken to be equal to the distribution of particles leaving the Boltzmann domain at the same point. Similarly, for the Boltzamnn domain, the distribution of incoming particles is taken to be equal to the distribution of particles leaving the Euler domain at this point. These two conditions are the two compatibility conditions to be imposed at the interface between the Euler and the Boltzmann domain. With these conditions, we match the half-flux going out from one domain to the half-flux entering the other domain.

The coupling algorithm is the time marching algorithm introduced and studied in [2] and [6]. It successively solves the Boltzmann equations on the Boltzmann domain with imposed boundary conditions deduced from the previous Euler solution, and the Euler equations on the Euler domain with imposed boundary conditions deduced from the Boltzmann solution.

The present paper begins by the derivation of a kinetic writing of the Euler equations, from which we deduce compatibility conditions to impose at the interface between Euler and Boltzmann (§2). The coupling algorithm is then described (§3), together with the numerical solvers to be used locally on each domain (§4,5). The resulting numerical method is then validated on several two-dimensional examples at low Knudsen numbers (§6). This presentation concludes by a brief analysis of the numerical results and by a description of the developments needed in order to realize a three-dimensional adaptive coupling between the Navier-Stokes and the Boltzmann equations.

2 Kinetic Euler Equations and Coupling Conditions

2.1 A simple kinetic writing of the Euler equations

Following Perthame [3], we first introduce a positive distribution function χ defined on IR^N and satisfying

$$\begin{cases} \chi(-\omega) = \chi(\omega), & \forall \omega \in IR^N, \\ \int_{IR^N} \chi(\omega) d\omega = 1, \\ \int_{IR^N} \omega_i^2 \chi(\omega) d\omega = 1, & \forall 1 \leq i \leq N. \end{cases} \tag{1}$$

To a given density field $\rho(x,t)$, velocity field $u(x,t)$ and temperature field $T(x,t)$, we then associate the distribution function

$$f_{\rho,u,T}(x,v,t) = \frac{\rho(x,t)}{(RT(x,t))^{N/2}} \chi\left(\frac{v-u(x,t)}{\sqrt{RT(x,t)}}\right). \tag{2}$$

Here, f is a kinetic distribution of particles in translation with velocity v.

In standard kinetic theory, the distribution function χ is taken as the Maxwellian

$$\chi(\omega) = \frac{1}{(2\pi)^{N/2}} exp\left(\frac{-\omega^2}{2}\right).$$

In this case, the associated distribution f is an equilibrium solution of the Boltzmann equations. In our numerical tests, we have also used the step function

$$\begin{aligned} \chi(\omega) &= \left(\frac{1}{\sqrt{3}}\right)^N \quad \text{if} \quad \sup|\omega_i| \leq \sqrt{3}, \\ &= 0 \quad \text{if not,} \end{aligned}$$

which leads to very simple calculations.

In any case, if we choose a distribution function χ satisfying (1), and if we introduce the notation $\lambda = (N+2-N\gamma)/2(\gamma-1)$ with $\gamma = \dfrac{C_P}{C_V}$ the ratio of specific heats, we have

$$\rho(x,t) = \int_{IR^N} f_{\rho,u,T}(x,v,t) dv, \tag{3}$$

$$\rho u(x,t) = \int_{IR^N} v f_{\rho,u,T}(x,v,t)dv, \tag{4}$$

$$E(x,t) = \int_{IR^N} \left(\frac{|v|^2}{2} + \lambda T \right) f_{\rho,u,T}(x,v,t)dv. \tag{5}$$

There is in fact a strong connection between the distribution function f and the corresponding moments ρ, ρU and E, as indicated in the next theorem due to Perthame [3]:

Theorem 2.1 *: Let f be solution of the linear kinetic problem*

$$\partial_t f + v\partial_x f = 0, \quad t \geq 0, \quad x, v \in IR^N, \tag{6}$$

with initial data

$$f(x,v,0) = f_{\rho_0,u_0,T_0}. \tag{7}$$

Then the moments ρ, ρu and E defined by (3)-(5) are first order approximation (in Δt) of the solution of the Euler equations for any time $t \leq \Delta t$.

This result has three useful consequences.

First, it reduces the numerical solution of the Euler equations to the numerical solution of a linear kinetic problem for which easy numerical algorithms can be introduced.

Secondly, since ρ, ρu and E are the moments of the distribution function f, their associated fluxes $F.n$ in the Euler equations will be the integral of the corresponding kinetic flux, that is

$$\begin{aligned} F.n &= F^+ + F^- \\ &= \int_{v.n \geq 0} (v.n) \begin{pmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \lambda T \end{pmatrix} f_{\rho,u,T}(v)dv \\ &\quad + \int_{v.n \leq 0} (v.n) \begin{pmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \lambda T \end{pmatrix} f_{\rho,u,T}(v)dv. \end{aligned} \tag{8}$$

Above n denotes the unit normal vector to the surface across which we compute the flux and F is the flux function associated to the Euler equations, that is

$$F(\rho, \rho u, E) = \begin{bmatrix} \rho u \\ \rho u \otimes u + pId \\ (E+p)u \end{bmatrix}, \tag{9}$$

with

$$p = \rho RT = (\gamma - 1)\left[E - \frac{1}{2}\rho|u|^2\right].$$

Finally, the above kinetic interpretation indicates the type of boundary condition which is relevant for the Euler equations. Indeed, the natural boundary condition for the kinetic problem (6) is

$$f(x, v, \varepsilon, t) = f_{imp}(x, v, \varepsilon, t), \forall x \in \partial\Omega_E, \forall t, \forall \varepsilon > 0, \forall v \text{ with } v.n \leq 0.$$

Here, ε is a distribution of internal energy associated to the term λT. If we write this condition in terms of the kinetic flux (8), we then obtain the following boundary condition for the Euler equation

$$\begin{aligned} F.n_{|\partial\Omega_E} &= \int_{v.n\geq 0} (v.n) \begin{pmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \lambda T \end{pmatrix} f_{\rho,u,T}(v)dv \\ &+ \int_{\varepsilon}\int_{v.n\leq 0} (v.n) \begin{pmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \varepsilon \end{pmatrix} f_{imp}(x, v, \varepsilon, t)dvd\varepsilon. \end{aligned} \tag{10}$$

2.2 Geometric description

We consider the flow of a rarefied gas in a global domain Ω around a solid body of boundary Γ_W (Figure 1). We split this domain into two (possibly overlapped) subdomains Ω_E and Ω_B. The domain Ω_B will be used for the Boltzmann equations. This is a local domain which contains the obstacle, with internal boundary Γ_W and external boundary $\Gamma_B \subset \bar{\Omega}_E$. The domain Ω_E will be used for the Euler equations. This is a large domain, with an internal boundary $\Gamma_{Eint} \subset \bar{\Omega}_B$ which surrounds the body and an external boundary Γ_{Eext} which is the external boundary of the computational domain Ω.

In the present approach, the subdomains Ω_E and Ω_B are fixed arbitrarily at the beginning of the calculation but in the future, the definition of Ω_E and Ω_B will be automatically adapted to the physical characteristics of the solution.

2.3 The Boltzmann problem

On Ω_B, as stated earlier, we solve the Boltzmann equations. When we have internal degrees of freedom, these are

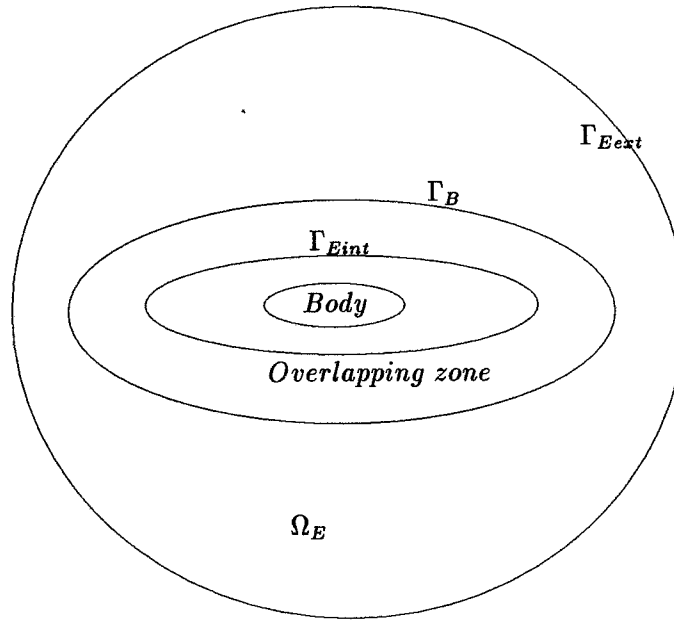

Figure 1: *Description of the Domain Splitting*

$$\partial_t f(x,v,\varepsilon,t) + v\partial_x f(x,v,\varepsilon,t) = Q(f,f)(x,v,\varepsilon,t) \tag{11}$$

with f the kinetic distribution of particles at point x with velocity v and internal energy ε, and $Q(f,f)$ the quadratic collision term

$$Q(f,f)(x,v,\varepsilon,t) = \int_{v_1\in IR^3}\int_{\varepsilon_1>0}\int_{\omega\in S^2}(f_1'f' - ff_1)q(v-v_1,\omega)dv_1d\varepsilon_1d\omega. \tag{12}$$

Above, $q(v-v_1,\omega)$ is the collision cross section, and we have used the notation

$$\begin{aligned} f_1 &= f(x,v_1,\varepsilon_1,t), \\ f' &= f(x,v',\varepsilon',t), \\ f_1' &= f(x,v_1',\varepsilon_1',t), \end{aligned}$$

with (v',ε') and (v_1',ε_1') the velocity and internal energy of the particles (v,ε) and (v_1,ε_1) after a collision of parameter ω. In our calculation, we have used a hard sphere model for the cross section ($q = d^2|(v-v_1).\omega|$)

and a Larsen-Borgnakke model for predicting the velocities and internal energies after collision [7].

At the wall Γ_w, we impose as kinetic boundary condition a linear combination of a specular reflexion and of a complete thermal accomodation. In other words, each particle colliding with the wall is reemitted with a velocity and internal energy distribution of the form

$$v' = (1-\alpha)v'_s + \alpha v'_a, \tag{13}$$

$$\varepsilon' = (1-\alpha)\varepsilon'_s + \alpha\varepsilon'_a. \tag{14}$$

Here $\alpha \in [0,1]$ is the coefficient of accomodation, v'_s and ε'_s are the velocity and internal energy of the particle after specular reflexion, and v'_a and ε'_a are the velocity and internal energy of the particle reemitted with a Maxwell distribution at temperature T_W. In a local orthonormal basis (t,n,z), with n normal to the wall, we have

$$v'_{st} = v_t, \quad v'_{sn} = -v_n, \quad v'_{sz} = v_z, \quad \varepsilon'_s = \varepsilon, \tag{15}$$

$$\begin{pmatrix} v'_{at} \\ v'_{az} \end{pmatrix} = \sqrt{-T_W \log a_2} \begin{pmatrix} \cos 2\pi a_1 \\ \sin 2\pi a_1 \end{pmatrix}, \tag{16}$$

$$v'_{an} = \sqrt{-T_W \log a_3}, \tag{17}$$

$$\varepsilon'_a = \frac{Nd}{4} \cdot \sqrt{-T_W \log a_4}, \tag{18}$$

with $(a_i)_{i=1,4}$ random numbers uniformly distributed between 0 and 1 and $\dfrac{Nd}{4}$ the number of internal degrees of freedom of a given particle.

As a final boundary condition, we must specify the kinetic distribution of the particles entering the domain Ω_B through the external boundary Γ_B. We suppose here that the gas leaving the Euler domain and entering the Boltzmann domain is in thermal equilibrium, which means that we impose the boundary condition

$$\begin{gathered} f_B(x,v,\varepsilon,t) = \frac{\rho}{(2\pi RT)^{3/2} N_d RT} \exp\left(\frac{-(v-u)^2}{2RT}\right) \exp\left(\frac{-\varepsilon}{N_d RT}\right), \\ \forall x \in \Gamma_B, \quad \forall v \text{ with } v.n_B \leq 0, \quad \forall \varepsilon > 0. \end{gathered} \tag{19}$$

Here, ρ, u and T are the local values of the Euler solution on Γ_B.

2.4 The Euler problem

On Ω_E, we solve the Euler problem

$$\partial_t \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix} + \operatorname{div}[F(\rho, \rho u, E)] = 0, \tag{20}$$

with F the flux function introduced in (9). To this, we add the kinetic boundary conditions introduced in (10), that is

$$\begin{aligned} F.n|_{\partial\Omega_E} &= \int_{v.n\geq 0} (v.n) \begin{pmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \lambda T \end{pmatrix} f_{\rho,u,T}(v)dv \\ &+ \int_\varepsilon \int_{v.n\leq 0} (v.n) \begin{pmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \varepsilon \end{pmatrix} f_{\text{imp}}(v,\varepsilon)d\varepsilon dv. \end{aligned} \tag{21}$$

In our coupled problem, we take as distribution of incoming particles the functions

$$f_{\text{imp}}|_{\Gamma_{E\ \text{ext}}} = f_{\rho_\infty,u_\infty,T_\infty} \text{ with } \varepsilon = \lambda T_\infty, \tag{22}$$

$$f_{\text{imp}}|_{\Gamma_{E\ \text{int}}} = f_B(x,v,\varepsilon,t). \tag{23}$$

In other words, the kinetic distribution imposed at infinity is the one associated to the uniform flow $(\rho_\infty, u_\infty, T_\infty)$ and the distribution at $\Gamma_{E\ \text{int}}$ is equal to the Boltzmann distribution introduced in §2.3.

In our numerical tests, the distribution function χ used to generate the distribution $f_{\rho,u,T}$ was the step function of §2.1. With this choice, the boundary condition (21) takes a simple algebraic form, which is not the case when one uses a Maxwell function in the definition of χ.

2.5 The final coupled problem

Our final problem consists now in finding steady state solutions of the Boltzmann equations (11)-(12) written on Ω_B with boundary conditions (13)-(19) and of the Euler equations (20) written on Ω_E with boundary conditions (21)-(23). These two systems are coupled through the boundary conditions (19) and (23). We will now introduce an algorithm which solves these systems in an uncoupled iterative procedure.

3 The Coupling Algorithm

The proposed algorithm is the time marching algorithm introduced in [2] or [6]. It proceeds as follows:

For $n \geq 0$, and knowing estimates $(\rho^n, \rho u^n, E^n)$ and f_B^n of the Euler and Boltzmann solutions on Ω_E and Ω_B, new estimates f_B^{n+1} and $(\rho^{n+1}, \rho u^{n+1}, E^{n+1})$ are computed by:

i) solving the Boltzmann equations (11)-(12) on Ω_B with initial condition $f(.,0) = f_B^n$ and boundary conditions (13)-(19). In (19), the incoming distribution f_B is calculated using the known values $(\rho, u, T) = (\rho^n, u^n, T^n)$. At the end of this step, we obtain an improved estimate f_B^{n+1} of the Boltzmann solution on Ω_E and an updated value

$$(F^{n+1})^- = \int_\varepsilon \int_{v.n \leq 0} (v.n) \begin{bmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \varepsilon \end{bmatrix} f_B^{n+1}(x, v, \varepsilon) dv d\varepsilon \qquad (24)$$

of the half-flux entering Ω_E through $\Gamma_{E\,\mathrm{int}}$;

ii) solving the Euler equations (20) on Ω_E with boundary conditions (21)-(23), where in (21) the value of the half-flux entering Ω_E thought $\Gamma_{E\mathrm{int}}$ is replaced by the value $(F^{n+1})^-$ computed in (24). At the end of this step, we obtain the desired new estimates $(\rho^{n+1}, \rho u^{n+1}, E^{n+1})$.

In practice, the above algorithm is stopped as soon as the trace of $(\rho^n, \rho u^n, E^n)$ on Γ_B reaches a stationary value. This means then that the boundary conditions for Boltzmann will not vary from one step to another and that therefore we have reached a steady solution of the Boltzmann equations (and hence of the Euler equations since the value $(F^{n+1})^-$ used in the boundary conditions (21)-(23) will then stay fixed). In our computations, the stopping criterion was

$$\mathrm{Res} = \left(\sum_{x_i \in \Gamma_B} |\rho_i^{n+1} - \rho_i^n| \right) / \left(\sum_{x_i \in \Gamma_B} \rho_i^0 \right) \leq 10^{-3}.$$

This value of 10^{-3} is the maximal accuracy which can be obtained in the Boltzmann solution because of the noise which is generated by the Monte Carlo procedure.

4 The Boltzmann Solver

The numerical solution of the Boltzmann equations uses a time marching technique, which splits the collision step and the transport step. The original technique was developed at the University of Kaiserslautern and was then upgraded at INRIA in order to take into account inelastic collisions and heterogeneous boundary conditions.

In this approach, the distribution f is represented by a discrete set of particles with positions (x_i), velocities (v_i), and energy (ε_i). The collision process is then simulated by a Monte Carlo technique. More precisely, starting with a given distribution of particles $(x_i, v_i, \varepsilon_i)$, we update this distribution at each time step by the following sequence of operations ([2]):

1) On each part of Γ_B, we generate additional particles with a random distribution in space and a Maxwellian distribution in velocity (as predicted by the available Euler solution).

2) Each particle is advanced in space by the formula

$$x_i = x_i + v_i \Delta t.$$

3) Each particle which collides with the body during step 2 has its velocity and energy changed according to (13)-(18).

4) Each particle which has left Ω_B is erased.

5) The particles belonging to a same rectangular cell may collide with each other. Within each cell, pairs of particles are built randomly. The decision of collision for a given pair is made randomly depending on their probability of collision. If collision occurs, the velocities and energies of the colliding particles are updated according to the Larsen-Borgnakke model, with collision parameters to be picked randomly.

This algorithm is stopped after a given number of time steps or if the average number of particles reaches a steady value. When the algorithm has converged, we still need to compute the flux $(F^{n+1})^-$ in (24). At the particle level, this is very easy. Indeed, by definition, the flux $(F^{n+1})^-$ on a given segment Γ_{seg} of $\Gamma_{E\ \text{int}}$ is the total flux of mass, momentum and energy carried by the particles which enter Ω_E through Γ_{seg} during the time interval $[t, t + \Delta t]$, divided by the length Δs of Γ_{seg} and the length Δt of the time interval. If I denotes the set of particles which enter Ω_E through Γ_{seg} during K consecutive time steps, this definition of $(F^{n+1})^-$ reduces to

$$(F^{n+1})^-|_{\Gamma_{\text{seg}}} = \frac{1}{K \Delta t \Delta s} \sum_{i \in I} \begin{bmatrix} m_i \\ m_i v_i \\ m_i \left(\frac{|v_i|^2}{2} + \varepsilon_i \right) \end{bmatrix}. \tag{25}$$

5 The Euler Solver

The numerical solution of the Euler equations (20) can be achieved by any of the solvers which have been proposed in the litterature. For example, we have run recent numerical tests with a SUPG solver developed by Dassault Aviation and based on a least square Galerkin formulation

of the Euler equations (using entropy variables). The tests presented herein are based on a Finite Volume discretization of the Euler equations and on a kinetic splitting of the fluxes at the cell interfaces.

In this approach, the domain Ω_E is partitioned into cells (C_i), each cell being associated to a node of an unstructured finite element triangulation of Ω_E (Figure 2). On each cell C_i, the equation (20) takes the simple iterative form

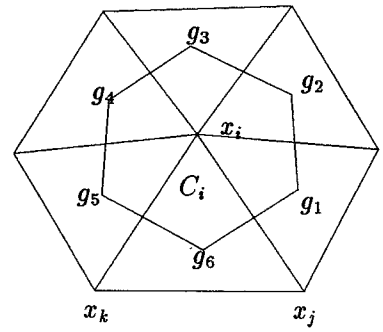

Figure 2: *Description of a Finite Volume cell*

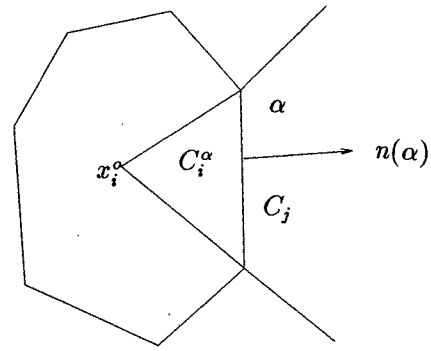

Figure 3: *Description of the cell boundary*

$$U_i^{n+1} - U_i^n + \sum_{\alpha \in N(i)} \Delta t \frac{\ell_\alpha}{\mathrm{Vol}(C_i)} F_\alpha . n_\alpha = 0. \tag{26}$$

Above $\alpha \in N(i)$ denotes the vertex α of the cell C_i. of length ℓ_α

and normal vector n_α, and $F_\alpha.n(\alpha)$ is the flux of conservative variables passing through this vertex.

On each vertex, we split this flux $F_\alpha.n(\alpha)$ into

$$F_\alpha.n(\alpha) = F^+(\mathcal{U}_i^n).n(\alpha) + F^-(\mathcal{U}_j^n).n(\alpha), \tag{27}$$

with j the index of the cell which is outside α, under the notation

$$F^+(\mathcal{U}).n(\alpha) = \int_{v.n(\alpha)\geq 0} (v.n) \begin{bmatrix} 1 \\ v \\ \frac{|v|^2}{2} + \lambda T \end{bmatrix} f_{\rho,u,T}(v)dv, \tag{28}$$

and with a similar definition for $F^-(\mathcal{U}).n(\alpha)$. Moreover, the variable $\mathcal{U}_i$ denotes the averaged quantity

$$\mathcal{U}_i = \frac{1}{\mathrm{Vol}(C_i)} \int_{C_i} \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} dx.$$

In (28), the values of ρ, u, T which are used are deduced from the above values of $\mathcal{U}_k$ by interpolation. This interpolation procedure is described in [4] and guarantees a second order accuracy in space.

On (26), the treatment of the boundary condition (21) is straightforward. For each vertex α on the boundary Γ_E, we replace (27) by (21) with f_{imp} given by (22) or (23).

The Euler algorithm (26) is stopped after a given number of time steps or as soon as we have

$$\frac{\|\rho^{n+1} - \rho^n\|_1}{\|\rho^0\|_1} \leq 10^{-16}.$$

6 Numerical tests

The numerical tests deal with the two-dimensional flow of a perfect diatomic gas ($\gamma = 1.4$) flowing at hypersonic speed around an ellipse of length $7.2m$ and thickness $2.18m$. The domains Ω_E and Ω_B are rectangles as indicated on Figure 4. In the cases without overlapping, the boundaries $\Gamma_{E\ \mathrm{int}}$ and Γ_B are identical.

At infinity, the characteristics of the flow are

$$u_\infty = 5672m/s, \quad T_\infty = 194^oK,$$

corresponding to a Mach number $M_\infty = 20$. The temperature of the body is $T_W = 1000^oK$. Such data are supposed to represent the flow around a reentry vehicle at an altitude of $70km$ or more.

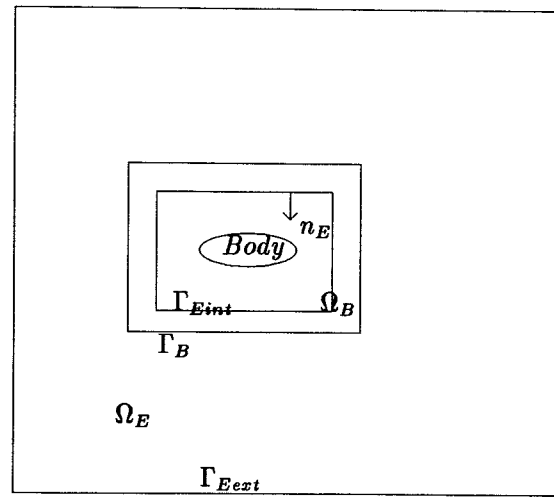

Figure 4: *Description of the Computational Domain*

In all our tests, we have used the coupling algorithm of §3. This algorithm has always converged in less than 15 iterations, and the ($\log_{10}$ of the) residual was rapidly decreasing during the iterations as indicated on Figure 5. At each iteration, the Boltzmann solution was taken to be the solution obtained after 300 steps of the time marching algorithm described in §4 (with a CFL number equal to 1) and the Euler solution was the solution obtained after 500 time steps of algorithm (26) (with a CFL number equal to 1.2).

As a reference solution, we have first computed the solution of the Euler equations on the whole domain Ω, at zero angle of attack, using an unstructured adapted mesh of 11 432 triangles. This solution was computed in $12h$ of CPU time on an Apollo DN 10 000 superworkstation.

The second calculation corresponds to the same flow, but computed by the coupled model with overlapping domains. The mean free path was $\lambda = 0,1m$ in the Boltzmann region ($Kn = 0.014$), and specular reflexion was imposed at the wall. The dimensions of the different domains were

$$\Omega \quad = 30m \times 30m,$$

$$\Omega_B \quad = 11.2m \times 4.2m,$$

$$\Gamma_{E\ \mathrm{int}} \quad = 10.2m \times 3.6m.$$

For the discretization, we have used 5 400 triangles on Ω_E and

7 700 rectangular cells for Ω_B, with an average of 25 particles per cell. The computing time was $22h$ on the same machine (for 10 iterations), that is approximately twice the time required for the Euler solution. The numerical results are represented on Figure 6 (density contours), Figure 7 (temperature contours) and Figure 8 (comparison of the density profiles of the reference Euler solution and of the coupled solution). We observe a very good superposition of all the different solutions (global Euler, local coupled Boltzmann and coupled Euler).

The third calculation is the same as the second one but without overlapping and with a very small Knudsen number ($\lambda = 0.01m$, $Kn = 0.0014$). We have again represented density contours (Figure 9), temperature contours (Figure 10) and density profiles (Figures 11 and 12). In this case, we still have a perfect superposition of the shocks, but the shock in the Boltzmann region is thinner, due to the very small Knudsen number. For such Knudsen numbers, it is impossible to solve the Boltzmann equations on the whole domain Ω.

The last calculation is still done without overlapping, but considers the more realistic case of a flow at a 30^o angle of attack, with a mean free path of $0.1m$ ($Kn = 0.014$) in the Boltzmann domain and full accomodation at the wall. The different domains are represented on Figure 13, and we have used 5 700 triangles in the Euler domain and 3 000 rectangular cells in the Boltzmann region. The coupled solution was obtained after 15 iterations and $30h$ of CPU time. The density contours are represented on Figure 13, the temperature contours on Figure 14, density profiles on Figure 15, and finally Figure 16 represents the heat flux distribution on the body. The density profile corresponds to a location where the shock crosses the interface but this has no effect on the quality of the matching. The values of the heat fluxes are sensitive to the discretization step in Ω_B and insensitive to the position of Γ_B. This means that we must choose in this case a small Boltzmann region with a very large number of cells.

7 Conclusion

The proposed technique of coupling Euler and Boltzmann by half-fluxes is very natural in its principle and in its implementation. Our numerical tests indicate that this coupling leads to very consistent results at low Knudsen numbers. We also observe that the coupling algorithm seems to converge independently of the boundary conditions at the wall, independently of the discretization step and independently of the position of the interface and of the amount of overlapping.

This coupled approach allows to study hypersonic flows at low Knud-

sen numbers, this at a reasonable cost (because it does not use a kinetic model everywhere in the domain) and with a good physical accuracy (because it uses a kinetic model in the boundary layer and in the afterbody region).

This approach must now be developed in three directions:

- automatic definition of the Boltzmann and Euler regions, which will require the introduction of mathematical criteria assessing the validity of the Euler model at a given point,
- replacement of the Euler equations by the Navier-Stokes equations, in order to be able to treat viscosity effects in the domain Ω_E,
- implementation in three-dimensional configurations.

Observe that these three-dimensional calculations will require large computer resources. The amount of memory and of CPU time which will be needed is at least three times more than what is required by a standard three-dimensional Euler calculation. The proposed coupling strategy does not reduce the computing time of the standard models of Computational Fluid Dynamics but it extends their domain of application to more rarefied regimes.

References

[1] J.F. Bourgat : Notice d'utilisation du Logiciel BOL2D, Version 2, Mai 92, INRIA.

[2] J.F. Bourgat, P. Le Tallec, D. Tidriri, Y. Qiu : Numerical coupling of nonconservative or kinetic models with the conservative compressible Navier-Stokes equations. Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations,Norfolk , May 1991, R. Glowinski, G. Meurant, B. Voigt, O. Widlund eds, SIAM 1992,

[3] B. Perthame : Second-order Boltzmann schemes for compressible Euler equations in one and two space dimensions, SIAM J. Numer. Anal., Vol. 29, n^o 1, February 1992, pp. 1-19.

[4] B. Perthame, Y. Qiu : A new variant of Van Leer's Method for multidimensional systems of conservation laws. Rapport INRIA n^o 1562, Nov. 1991.

[5] E. Turkel, B. Van Leer: Flux vector splitting and Runge-Kutta methods for the Euler equations, ICASE Report 84-27, June 1984.

[6] D. Tidriri : Couplage d'approximations et de modèles de type différents dans le calcul d'écoulements externes, Thèse, Paris 9, Mai 1992.

[7] C. Borgnakke, P.S. Larsen: Statistical collision model for Monte Carlo simulation of polyatomic gas mixture, J. Comp. Phys., Vol. 18, pp.405-420, 1975.

INRIA, Domaine de Voluceau, B.P. 105, 78153 Le Chesnay Cedex, France.
e.mail address : le_tallec@menusin.inria.fr

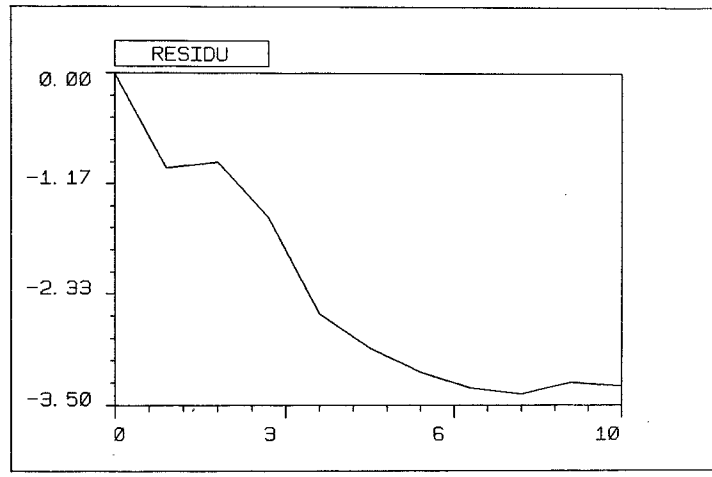

Figure 5: Convergence curve

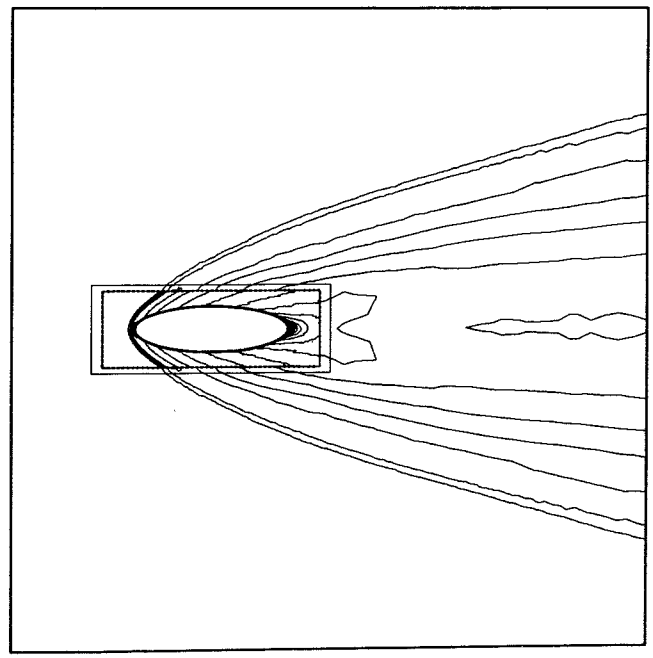

Figure 6: Overlapping: log of density

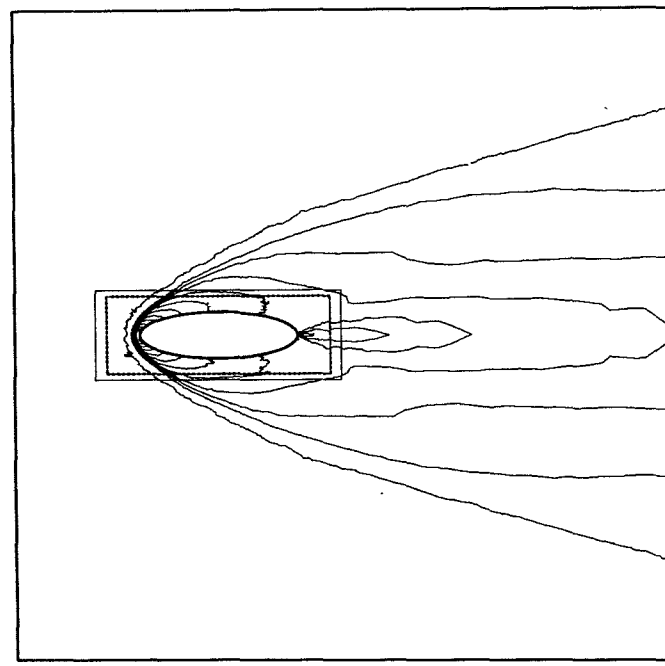

Figure 7: Overlapping: temperature

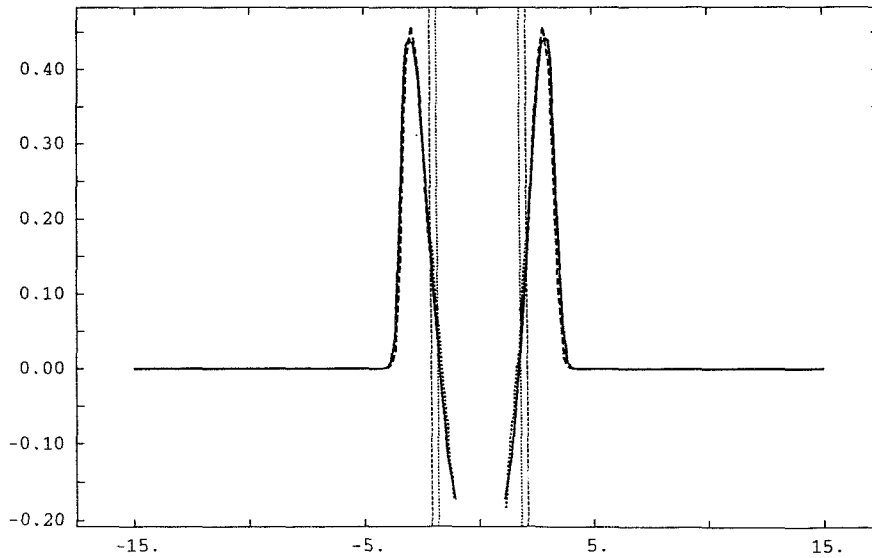

Figure 8: Overlapping: log of density profile at midbody

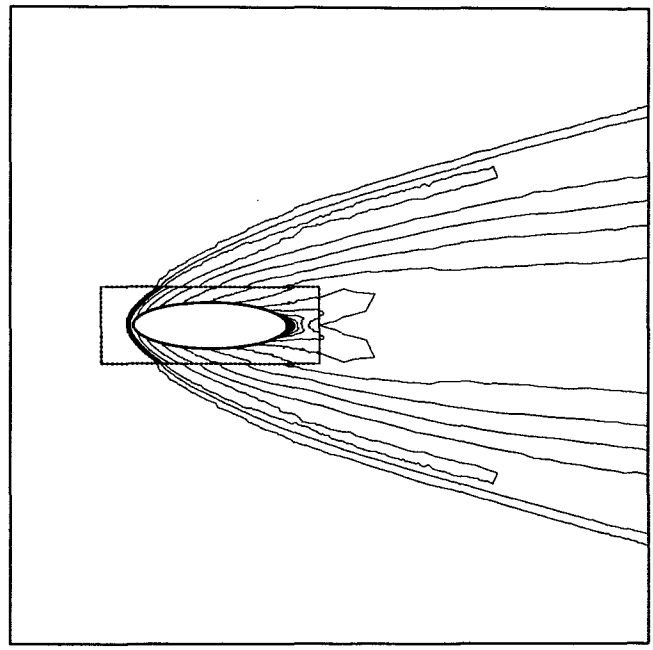

Figure 9: No overlapping: log of density

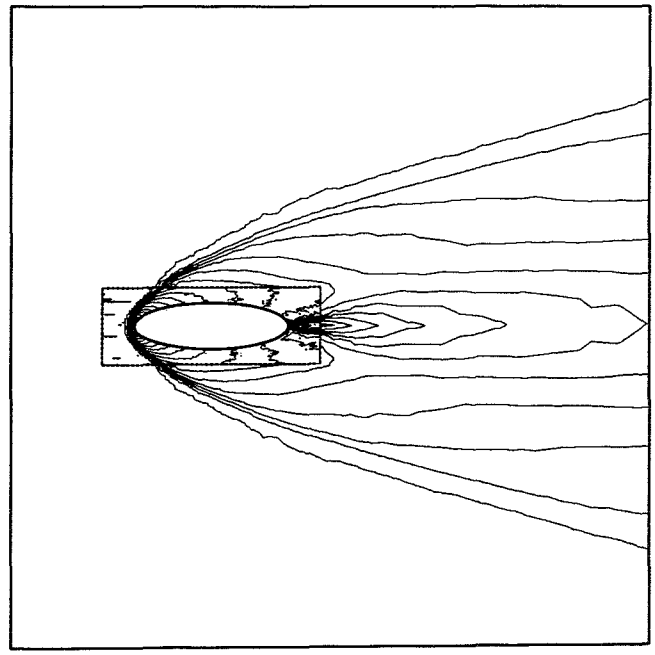

Figure 10: No overlapping: temperature

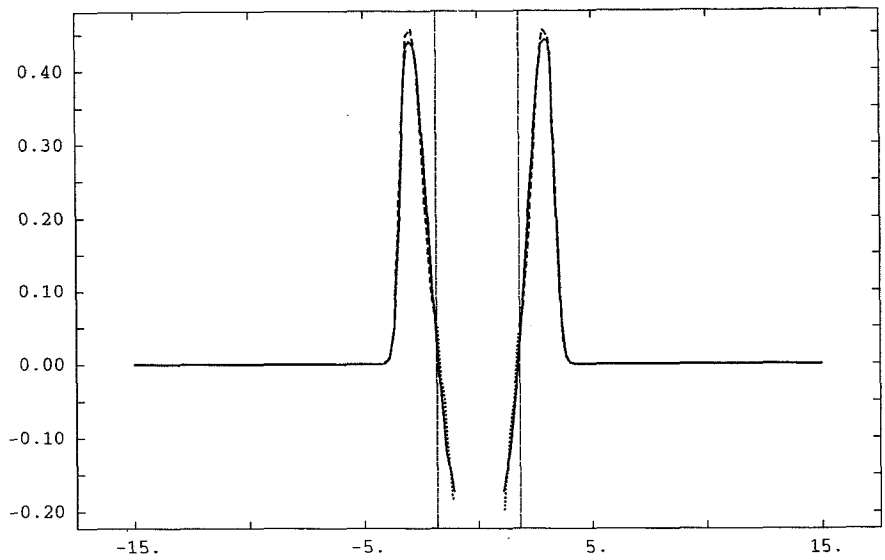

Figure 11: No overlapping: log of density profiles at midbody

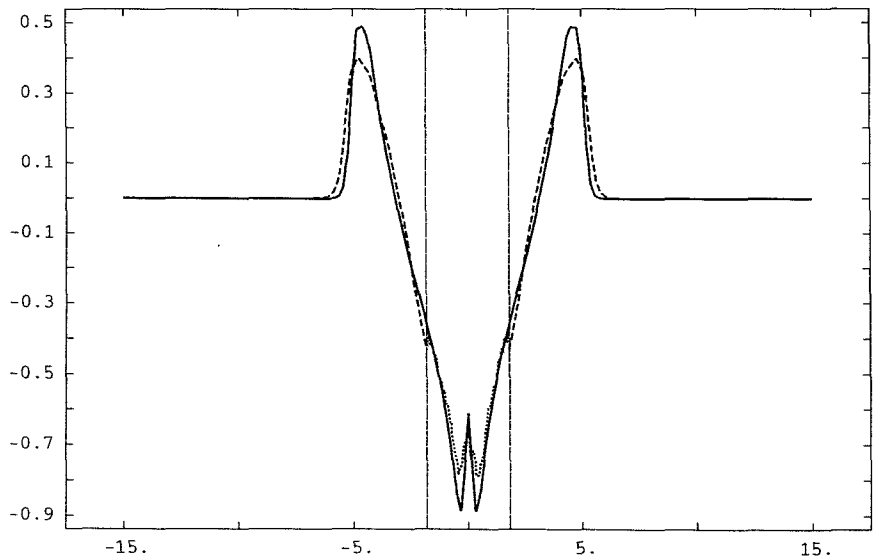

Figure 12: No overlapping: log of density profiles in the wake

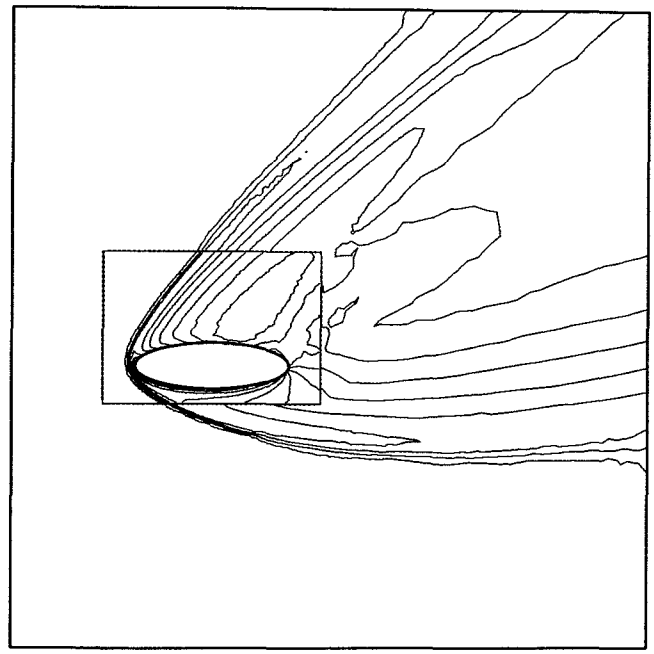

Figure 13: No overlapping and angle of attack: log of density

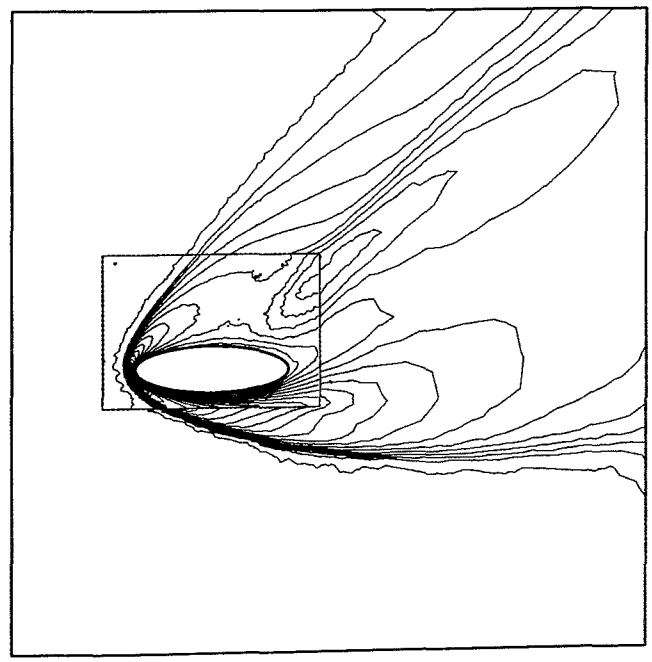

Figure 14: No overlapping and angle of attack: temperature

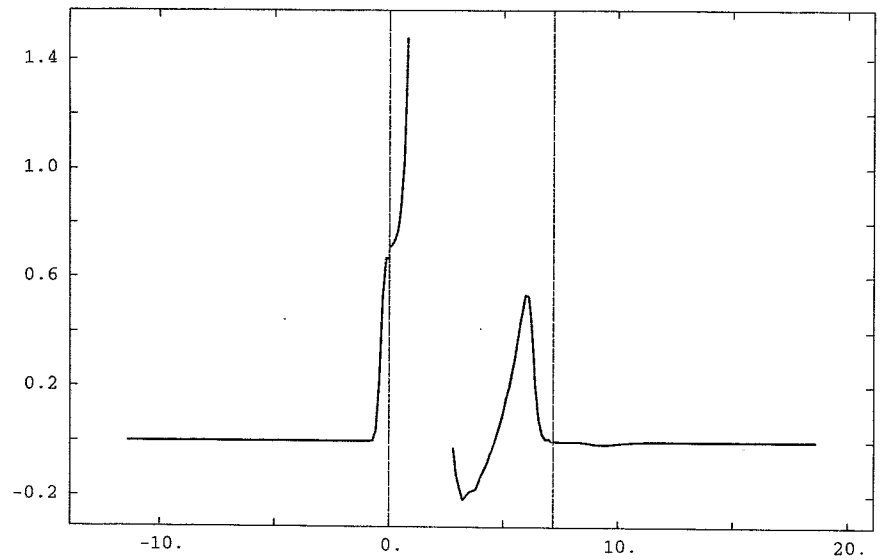

Figure 15: Density profile with an angle attack

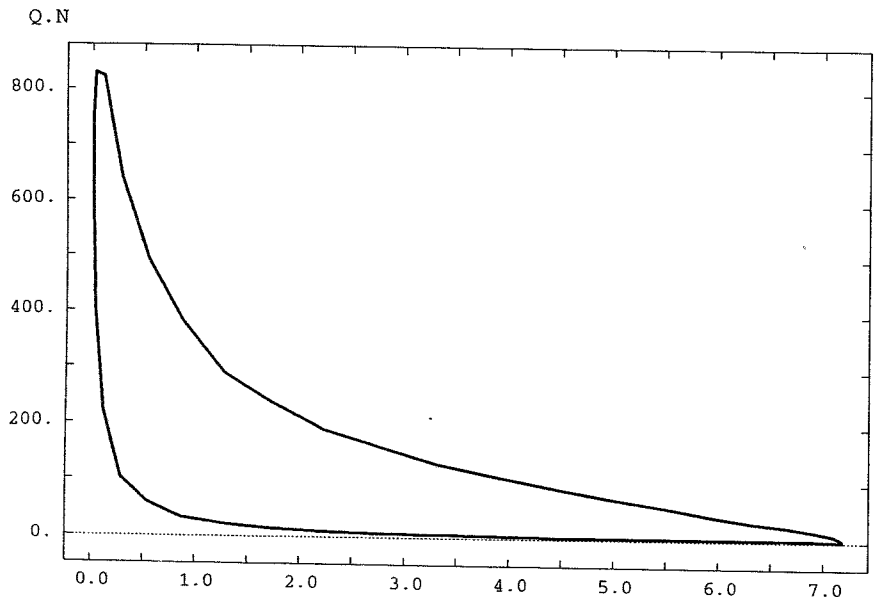

Figure 16: Heat flux distribution