FOI
SWEDISH DEFENCE
RESEARCH AGENCY

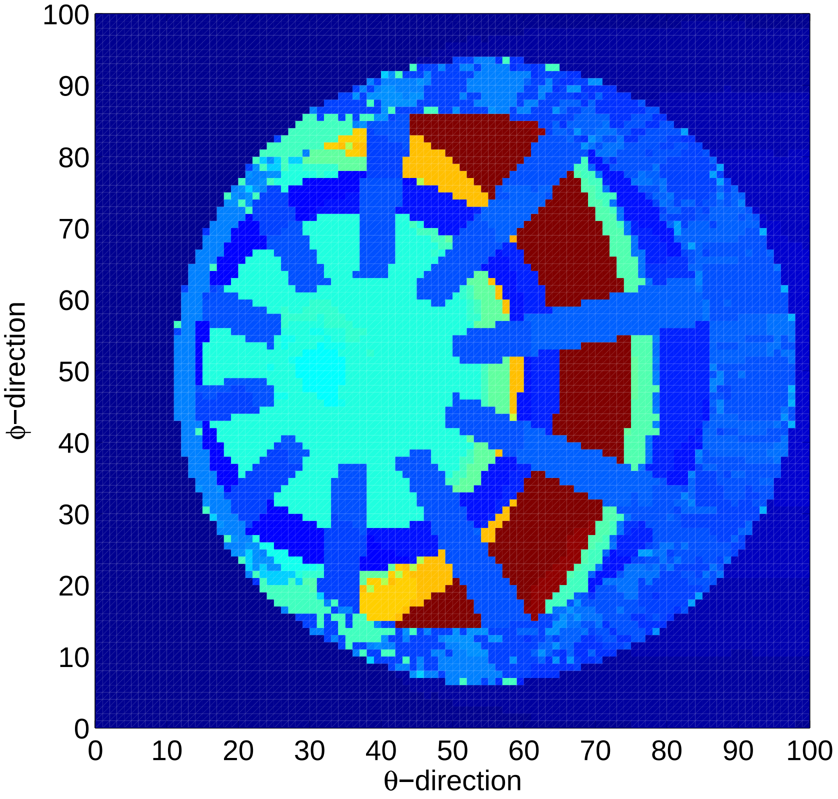
FOI-R–0828–SE
January 2003
1650-1942

Methodology report

Marlene Andersson

SIGGE V1.0

Theory, Implementation and Verification



Division of Aeronautics, FFA
SE-172 90 STOCKHOLM

FOI – Swedish Defence Research Agency
Division of Aeronautics, FFA
SE-172 90 STOCKHOLM

FOI-R–0828–SE
January 2003
1650-1942

Methodology report

Marlene Andersson

SIGGE V1.0

Theory, Implementation and Verification

Division of Aeronautics, FFA
SE-172 90 STOCKHOLM

Abstract

To be able to predict the IR-signature of an aircraft, the heat radiated, in form of IR-radiation, has to be calculated. A program package, SIGGE, has been developed for this purpose. The already existing code has been improved. The code can now handle radiation from hot gases and from walls with complex geometries, including specular reflections.

This report describes some key equations used and a detailed description of SIGGE is presented. The code has been verified against other calculations with satisfactory result.

Contents

List of Symbols

A	area	[m^2]
b	black body	
c_0	speed of light in vacuum	2.9979$\times$10^8 m/s
h	Planck’s constant	6.6261$\times$10^{-34} Js
I	intensity	[W/sr]
I$_\eta$	spectral intensity	[W/sr/cm^{-1}]
i	cell number	[-]
k	Boltzmann constant	1.3807$\times$10^{-23} J/K
k	“specie”	
L	radiance	[W/sr/m^2]
L$_\eta$	spectral radiance	[W/sr/cm^{-1}/m^2]
n_θ, n_ϕ	number of rays	[-]
p	pressure	[Pa]
s	length	[m]
t	length	[m]
T	temperature	[K]
TP	target position	
VP	view position	
α	angle	[rad]
ε	emissivity	[-]
η	wave number	[cm^{-1}]
θ	angle	[rad]
$\overline{\kappa}_\eta$	mean absorption coefficient	[cm^{-1}]
$\overline{\kappa}_{k,\eta}$	mean absorption coefficient of species	[cm^{-1}]
$\overline{\kappa}_{\rho\eta}$	mass absorption coefficient	[cm^{-1} m^3/kg]
$\overline{\kappa}_{p\eta}$	pressure absorption coefficient	[cm^{-1}/Pa]
ρ	density	[kg/m^3]
ρ_{ss}	reflectivity	[-]
τ	transmissivity	[-]
τ_η	spectral transmissivity	[-]
ϕ	angle	[rad]

1 Introduction

When aircraft have a low radar signature, their probalility to penetrate hostile air space increases, which at the same time increases the risk factor of discovery by infra red (IR) sensors. Thus, it is of great importance to minimize the IR-radiation from the aircraft. The intensity of the radiation is dependent on the temperature and the structure of the surface. In a gas, the IR-radiation heat transfer depends on the temperature and the composition of the gas and a hot plume will contribute to the IR-signature.

The Department of Computational Aerodynamics, Aeronautics Division (FFA), at the Swedish Defence Research Agency (FOI), is developing a code, SIGGE, for calculating IR-signatures for aircraft. A β-version of SIGGE already exists [1], which has been validated for hot gases against experimental data [2]. The present report complete earlier reports [1, 2, 3] with some improvements, a description of the code and verification.

The code SIGGE uses a line-of-sight method with an efficient ray-tracing algorithm. Very complex geometries can be handled without any problems. In SIGGE V1.0, the capability of handling specular reflection on walls has been implemented and the band models used have been improved. In the earlier β-version of SIGGE, only the spectral IR-intensity was calculated, but in this version the spectral radiance is calculated as well. The ability to use mole fraction as an alternative to mass fraction [2] for distribution of species has also been implemented.

The new code, SIGGE V1.0, has been verified against other calculations. IR-radiation from a simplified RM12 engine has been studied. Contributions from walls have been verified both with and without reflections.

This report describes the theory used to do calculations with reflections, section 3, and how this was implemented in the code, section 4. The report also describes some programs used for pre- and post-processing, section 5. The news have been verified and the result is presented in section 6. In section 7, conclusions and outlook are found.

2 Basic concepts of the program

The program SIGGE is module based. Each module can operate alone or together with the other modules. A proposed structure of SIGGE is presented in Figure 1. In SIGGE V1.0, the modules RAYFINDER and RADCALC are integrated in the package, see section 4, while the “Data treatment” is delivered separately as postprocessing tools, see section 5.

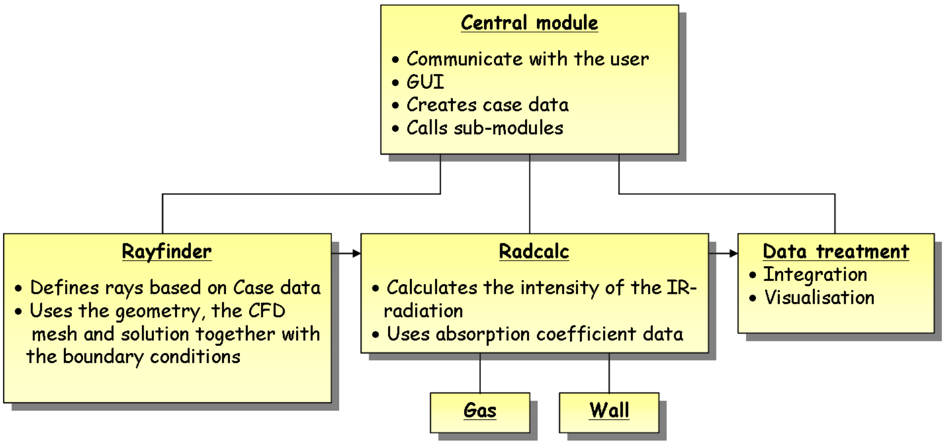

Central module
• Communicate with the user
• GUI
• Creates case data
• Calls sub-modules
Rayfinder
• Defines rays based on Case data
• Uses the geometry, the CFD mesh and solution together with the boundary conditions
Radcalc
• Calculates the intensity of the IR-radiation
• Uses absorption coefficient data
Data treatment
• Integration
• Visualisation
Gas
Wall

Figure 1. The proposed structure of the module based code SIGGE.

A CFD-solution[1] with a mesh of the area of interest is required as an input for SIGGE. The CFD-solution should contain values of variables such as temperature, pressure and distribution of species. Rays will be drawn through the mesh from different view positions, defined by the user, see Figure 2.

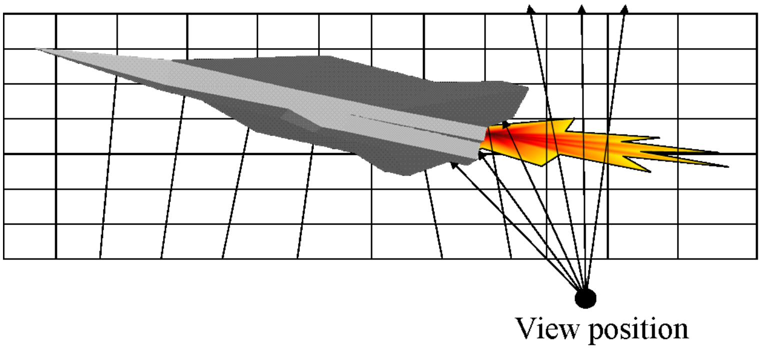


Figure 2. Rays are drawn from a view position toward an object in space through a mesh.

At the view position, an imaginary sensor is defined. The sensor consists of a flat plane with a number of pixels, giving an IR-image for each specified wave number, see Figure 3. The orientation of the sensor in space is defined by two orthogonal vectors, $\hat{\theta}$ and $\hat{\phi}$. Each pixel has a field-of-view of $\Delta\theta \times \Delta\phi$, which gives a total field-of-view of $n_\theta\Delta\theta \times n_\phi\Delta\phi$, where n_θ and n_ϕ are the number of pixels in the $\hat{\theta}$- and $\hat{\phi}$-directions, respectively. The number of rays, $n_\theta \times n_\phi$, and the angular resolution, $\Delta\theta \times \Delta\phi$, are defined by the user.

The user will also define a target position, which is the position in space where the centre of the sensor is aiming at, giving an aiming direction. This direction should be orthogonal to the sensor plane.

The module RAYFINDER will use this information to draw rays from the view position through the mesh. Each ray will be separated from their neighbours

[1] CFD - Computational Fluid Dynamics

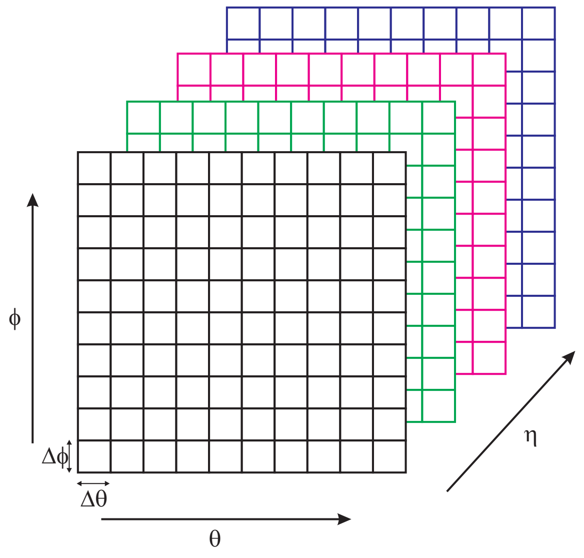


Figure 3. Imaginary sensor defined by the user. The sensor will collect one image for each wave number, η, specified. Each pixel has the field-of-view: $\Delta\theta \times \Delta\phi$.

with angles of $\Delta\theta$ and $\Delta\phi$. Thus, each ray will correspond to a pixel in the imaginary sensor. In Figure 4, a view position and a target position are defined in space, making, in this case, the aiming direction of the sensor run along the z-axis. The imaginary sensor will, therefore, coincide with the x-y plane and the $\hat{\theta}$-direction can be chosen to run along the x-axis and, consistently, the $\hat{\phi}$-direction will run along the y-axis.

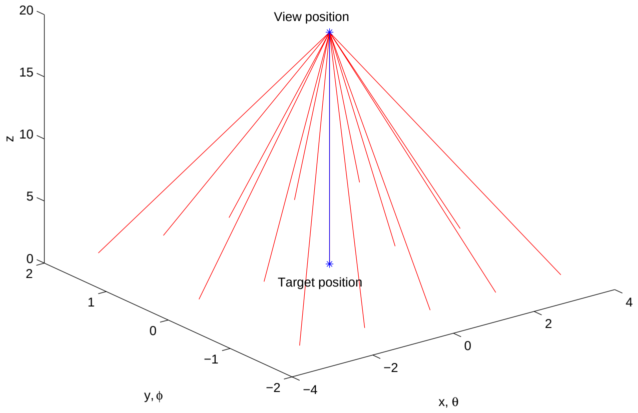


Figure 4. Rays are drawn from a view position, where the centre-ray is drawn toward the target position. Each ray is separated from the neighbouring rays with angles of $\Delta\theta$ and $\Delta\phi$.

If the mesh is axisymmetric, it does not have to cover the whole space. The module RAYFINDER will mirror the mesh around the axisymmetrical axis and associate the values from the CFD-solution with cells in a virtual mesh. The mesh as to cover either half the space or a fourth of the space, given one or two symmetry planes, respectively.

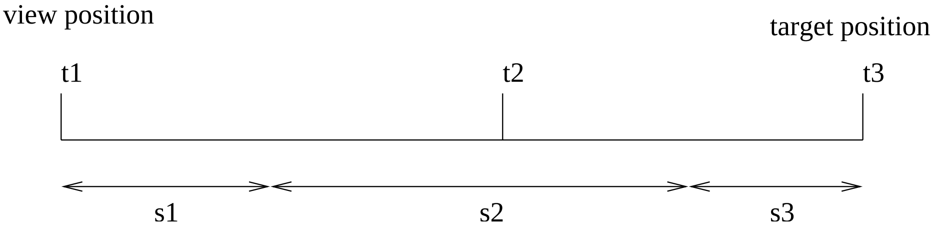
In the module RADCALC, the spectral intensity, I_η, and the spectral radiance, L_η, are calculated for each ray, see section 3.

3 Theory

3.1 General equations

The mesh around the object of interest is represented in a coordinate system. The rays drawn from the view position are represented in the same coordinate system. The coordinates of each ray are transformed to a one-dimensional system, given by t_1, t_2 . . . . In Figure 5, an example, with only three cells involved, is sketched.

Figure 5. One example of the cell intervals. t_1 - t_3 are one-dimensional coordinates and s_1 - s_3 are the lengths of the cells.



The view position is placed in $t_1 = 0$. From the one-dimensional coordinates, the length of each cell is defined as:

$$s_1 = \frac{t_2 - t_1}{2} \tag{1}$$

$$s_2 = \frac{t_3 - t_1}{2} \tag{2}$$

$$s_3 = \frac{t_3 - t_2}{2} \tag{3}$$

The intensity for each ray can be calculated by adding the contribution from each cell, Eq. (4), and a possible contribution from a wall, Eq. (5).

$$\mathrm{I}_\eta^{gas} = \sum_{i=1}^{\#\ cell} \left(\mathrm{L}_{b\eta}^{i} \mathrm{A}_i \left(\tau_{\eta,i-1} - \tau_{\eta,i} \right) \right) \tag{4}$$

$$\mathrm{I}_\eta^{wall} = \varepsilon(\theta) \mathrm{L}_{b\eta}^{wall} \mathrm{A}_{wall} \tau_{\eta,\#\ cells} \tag{5}$$

In Eq. (4) and Eq. (5), η is the wave number, $\tau_{\eta,i}$ is the transmissivity from the i^{th} cell to the view position , ε is the angle dependent emissivity of the wall and A_i is an area segment defined as:

$$\mathrm{A}_i = t_i^2 \Delta\theta \Delta\phi \tag{6}$$

where t_i is the distance to the view position discussed above and $\Delta\theta \times \Delta\phi$ is the angular resolution discussed in section 2. $\mathrm{L}_{b\eta}^{i}$ is the spectral black body radiance of the i^{th} cell, given by:

$$\mathrm{L}_{b\eta}^{i} = \frac{2\pi h c_0^2 \eta^3}{\pi(e^{hc_0\eta/kT} - 1)} = \frac{C_1 \eta^3}{e^{C_2\eta/T} - 1} \tag{7}$$

where

$$C_1 = 2hc_0^2 = 1.191 \cdot 10^{-16} W m^2 \tag{8}$$

$$C_2 = hc_0/k = 1.4388 \cdot 10^{-2} K m \tag{9}$$

These equations are derived in Ref. [3], where the difference between intensity and radiance is carefully discussed, with the conclusion that:

$$\mathrm{I} = \mathrm{L} \times \mathrm{A} \tag{10}$$

where A is the area segment defined in Eq. (6).

3.1.1 Transmissivity

When estimating the transmissivity, the Statistical Narrowband (SNB) model for a uniform gas is used. The line distribution within a band is assumed to be exponential. This includes broadening due to collisions with other molecules and for one cell with length s, the transmissivity is written as [4]:

$$-\ln \tau_\eta^k = \frac{G}{\sqrt{1 + \frac{G}{2a_\eta}}} \tag{11}$$

$$G = \overline{\kappa}_{k,\eta} p_k s \frac{\mathrm{T}_0}{\mathrm{T}} \tag{12}$$

where $\overline{\kappa}_\eta$ is the spectral mean-absorption coefficient for the current “specie” k, p_k is the partial pressure for “specie” k, T_0 is the reference temperature equal to 273.15 K and T is the temperature in the current cell. The absorption coefficients vary for different species and are dependent on the wave number, the temperature and the distribution of species. The absorption coefficients are stored in a data base, Appendix E. The outcome of the calculation is dependent on the quality and correctness of the data base and the CFD-solution. The broadening parameter a is given by:

$$a = \frac{\gamma_k}{d(\mathrm{T})} \tag{13}$$

where

$$d(\mathrm{T}) = e^{-0.00106\mathrm{T}+1.21} \tag{14}$$

and

$$\gamma_k = \left[\sum_l (\gamma_{k,l})_{T_0} \, p_l \left(\frac{\mathrm{T}_0}{\mathrm{T}} \right)^{n_{k,l}} \right] + (\gamma_{k,k})_{T_0} \, p_k \left(\frac{\mathrm{T}_0}{\mathrm{T}} \right)^{n_{k,k}} \tag{15}$$

Only broading due to collisions with N_2 and self-broadening is included. The broadening coefficients used are summerized in Table 1.

Table 1. Broadening coefficients used in Eq. (15).

Specie	H_2O	CO_2
$\gamma_{k,k}$	0.44	0.01
$\gamma_{k,l}$	0.09	0.07
$\mathrm{n}_{k,k}$, $\mathrm{n}_{k,l}$	0.5	0.5

The total transmissivity for each “specie” k, used in Eq. (4) and Eq. (5), is given by multiplying the contribution from the current cells with the contributions from all cells between the current cell, i, and the view position:

$$\tau^k_{\eta,i} = \prod_{j=1}^{i} \tau^k_{\eta,j} \tag{16}$$

For H_2O, the transmissivity is corrected for "strong lines" [5, 6]. If $x \geq 0.1$, the total transmissivity, from the i^{th} cell to the view position, is calculated as:

$$-\ln \tau^k_{\eta,i} = \frac{B}{\sqrt{1 + \frac{\pi}{2}x}} \tag{17}$$

where B and x are given by:

$$B = \sum_{j=1}^{i} G_j \tag{18}$$

$$x = \frac{B^2}{2\pi \sum_{j=1}^{i} a_j G_j} \tag{19}$$

where a_j and G_j are the same a and G discussed above.

The total transmissivity for all species is given by multiply the transmissivity of each of the species:

$$\tau_{\eta,i} = \prod_{k=1}^{\text{\# of species}} \tau^k_{\eta,i} \tag{20}$$

3.2 Reflection

3.2.1 Method

The specular spectral reflectivity, ρ_{ss}, is defined as [7]:

$$\rho_{ss} = \frac{\text{reflected part of incoming radiation}}{\text{total incoming radiation}} \tag{21}$$

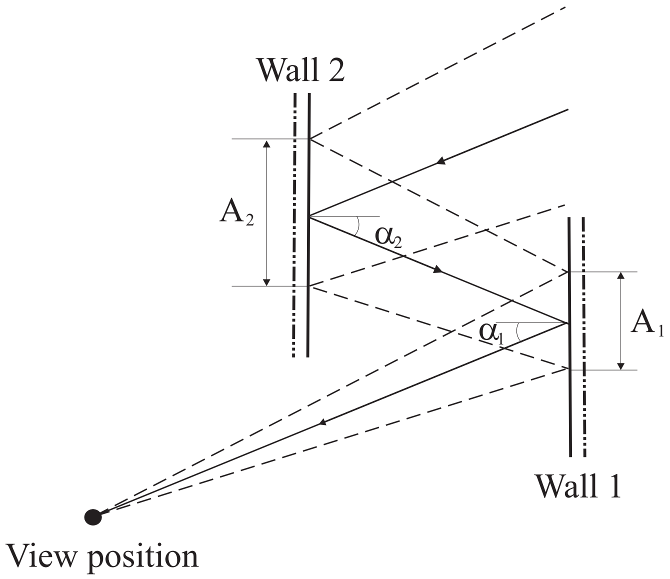
Using the line-of-sight method, the reflected ray path can be described as sketched in Figure 6. The walls in Figure 6 can be non-parallel and, thus, it is possible that α_1 is not equal to α_2.

In this example, incoming radiation is reflected on wall 2 and, thus, reduced by a factor given as the reflectivity, $\rho^{(2)}_{ss}$, of wall 2, defined in Eq. (21). The radiation reaching wall 1 is given by:

$$\begin{aligned} \mathrm{I}^{(1)}_\eta &= \rho^{(2)}_{ss}(\alpha_2) \cdot \mathrm{L}^{in}_\eta \mathrm{A}_2 + \varepsilon^{(2)}_\eta(\alpha_2) \cdot \mathrm{L}^{wall}_{b\eta} \mathrm{A}_{wall} \tau_{\eta,\#\ cells} + \\ &\quad + \sum_{i=1}^{\#\ cells} \left(\mathrm{L}^i_{b\eta} \mathrm{A}_i \left(\tau_{\eta,i-1} - \tau_{\eta,i} \right) \right) = \\ &= \rho^{(2)}_{ss}(\alpha_2) \cdot \mathrm{I}^{in}_\eta + \mathrm{I}^{wall(2)}_\eta + \mathrm{I}^{gas(2)}_\eta \end{aligned} \tag{22}$$

where $\varepsilon^{(2)}_\eta(\alpha_2)$ is the emissivity of wall 2, $\tau_{\eta,i}$ is the total transmissivity between wall 1 and wall 2. The intensities $\mathrm{I}^{gas(2)}_\eta$ and $\mathrm{I}^{wall(2)}_\eta$ are calculated in the same way as described in section 3.1.

Figure 6. Sketch of reflection in one wall.



At wall 1, the radiation is reflected again and reduced with the reflectivity of wall 1, $\rho_{ss}^{(1)}$. Thus, the radiation reaching the view position is given by:

$$\mathrm{I}_{\eta}^{(VP)} = \rho_{ss}^{(1)}(\alpha_1) \cdot \mathrm{I}_{\eta}^{(1)} + \mathrm{I}_{\eta}^{wall(1)} + \mathrm{I}_{\eta}^{gas(1)} \tag{23}$$

where $\mathrm{I}_{\eta}^{gas(1)}$ and $\mathrm{I}_{\eta}^{wall(1)}$ are the contributions from the gas between wall 1 and the view position and the contribution from wall 1, respectively. If no reflections were considered, the resulting intensity will be given by $\mathrm{I}_{\eta}^{gas(1)}$ + $\mathrm{I}_{\eta}^{wall(1)}$.

3.2.2 Calculating the reflection

As seen in Eq. (22) and (23), each reflected ray is treated separately, i.e. the intensity and the radiance are calculated along the ray as an original ray starting at the view position, VP. Thus, the extension of the reflected ray, through the mesh, has to be calculated. As for the original ray, a view position, VP_{new} and a target position, TP_{new}, are needed, see section 2. The original view position, VP, the orignal target position, TP, the hit coordinate, HC, and the normal of the wall are given by SIGGE. The hit coordonate, HC, is the point in space where the line connecting VP and TP cuts the first wall. The new view position and the new target position can be calculated from the orignal target position, the hit coordinate and the normal of the wall. In Table 2, the variables given and needed are collected and in Figure 7, the variables are defined.

The new view position is the same as the hit coordinate at the wall:

$$\mathrm{VP}_{new} = \mathrm{HC} \tag{24}$$

The new target position is given by:

$$\mathrm{TP}_{new} = \mathrm{VP} + \bar{\mathrm{a}}_1 \tag{25}$$

The vector $\bar{\mathrm{a}}_1$ is defined as:

$$\bar{\mathrm{a}}_1 = 2 \cdot (\bar{\mathrm{v}}_1 + \bar{\mathrm{n}}_2) \tag{26}$$

Table 2. List of variables

Name	Abbreviation	Description
View position	VP	Position in space where imaginary sensor is situated. Given by SIGGE
Hit coordinate	HC	Position in space where ray hit the wall. Given by SIGGE
Normal to wall	$\bar{\mathrm{n}}_1$	Normal to wall given by SIGGE.
Normal to wall	$\bar{\mathrm{n}}_2$	Normal with length from HC to middle of $\bar{\mathrm{a}}_1$, see Fig
New view position	VP_{new}	View position for reflected ray.
New target position	TP_{new}	Target position for reflected ray.
	$\bar{\mathrm{v}}_1$	Vector between VP and HC. $\bar{\mathrm{v}}_1$ = HC - VP
	$\bar{\mathrm{a}}_1$	Vector between VP and TP_{new}. TP_{new} = VP + $\bar{\mathrm{a}}_1$

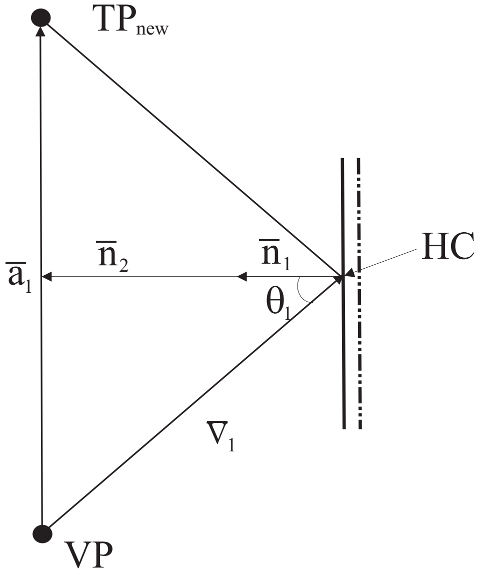


Figure 7. Definitions of variables for calculating new target position.

where $\bar{\mathrm{v}}_1$ and $\bar{\mathrm{n}}_2$ are given by:

$$\bar{\mathrm{v}}_1 = \mathrm{HC} - \mathrm{VP} \qquad \bar{\mathrm{n}}_2 = \frac{-\bar{\mathrm{v}}_1 \cdot \bar{\mathrm{n}}_1}{|\bar{\mathrm{n}}_1|^2}\bar{\mathrm{n}}_1 \qquad (27)$$

where HC, VP and $\bar{\mathrm{n}}_1$ are given by SIGGE.

4 Detailed description of the program

The program SIGGE requires a number of input files and will generate two output files, see section 4.1 and section 4.2. All files used by SIGGE have a common data format, the FFA-format, which is described in Ref. [1, 8]. The basic layout of the modules RAYFINDER and RADCALC are found in section 4.3 and section 4.4. Many of the subroutines have been modified in this version of SIGGE and, thus, all subroutines are described in section 4.5 and section 4.6.

4.1 Input files needed

General input file In this file, variables needed to run the program are defined together with the names of the other input files and the output files. This file is refered to as the *input file* in this report and is described in detail in Appendix B. The file normally has the extension *.ainp*.

Boundary condition file Contains the boundary conditions, used both in the CFD-solution and for generating rays in SIGGE. The extension is *.aboc*.

Mesh file Contains the mesh and has the extension *.bmsh*.

CFD-solution file Contains the CFD-solution with the extension *.bout*.

The *boundary condition file*, the *mesh file* and the *CFD-solution file* have a format which is required by the CFD-solver EDGE [9, 10, 11]. EDGE is an edge-based Euler solver for unstructured grids developed at the Department of Computational Aerodynamics, Aeronautics Division (FFA), FOI.

Ray definition file In this file, the user defines how the rays through the mesh should be drawn. In this report, this file is referred to as the *ray definition file*. The file is described in Appendix C and has the extension *.aray*.

Wall definition file In this file, the user defines properties, such as temperature, emissivity and reflectivity, of different walls. This file is referred to as the *wall definition file*. The file is described in Appendix D and has the extension *.awall*.

Absorption data base file Contains the mean absorption coefficients needed to calculate the IR-intensity. The extension is *.btab* and the file is described in Appendix E.

4.2 Output files generated

Ray data file File generated by the subroutine RAYFINDER, which contains all information needed to calculate the IR-intensity. This is also an input file for the module RADCALC, when discussing this module, this file is referred to as the *indata file*. The extension is *.bdat*

Output file If only RAYFINDER has been run, this file is the same as the *ray data file* described above. If RADCALC has been run, the file has been complemented with the IR-intensities and IR-radiances.

The information and structure of the output files are found in Figure 8 and Figure 9.

Figure 8. The basic structure of the output files. Only the names of the different sub-data sets are displayed. The number of sub-data sets, *sub_ray*, are given by the number of rays defined by the user. The sub-data sets *free_stream_data* and *flow_data* contain more sub-data sets.

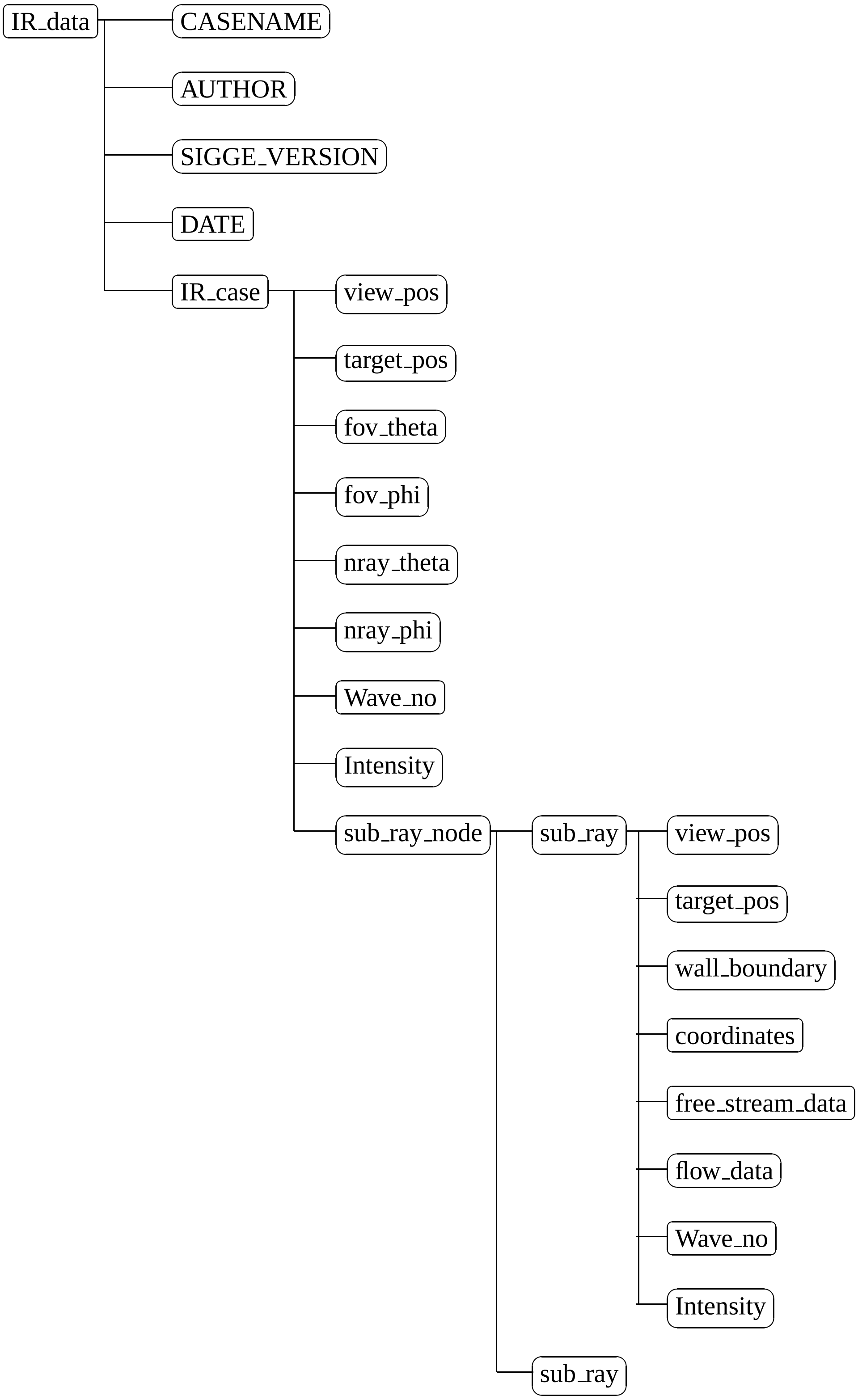
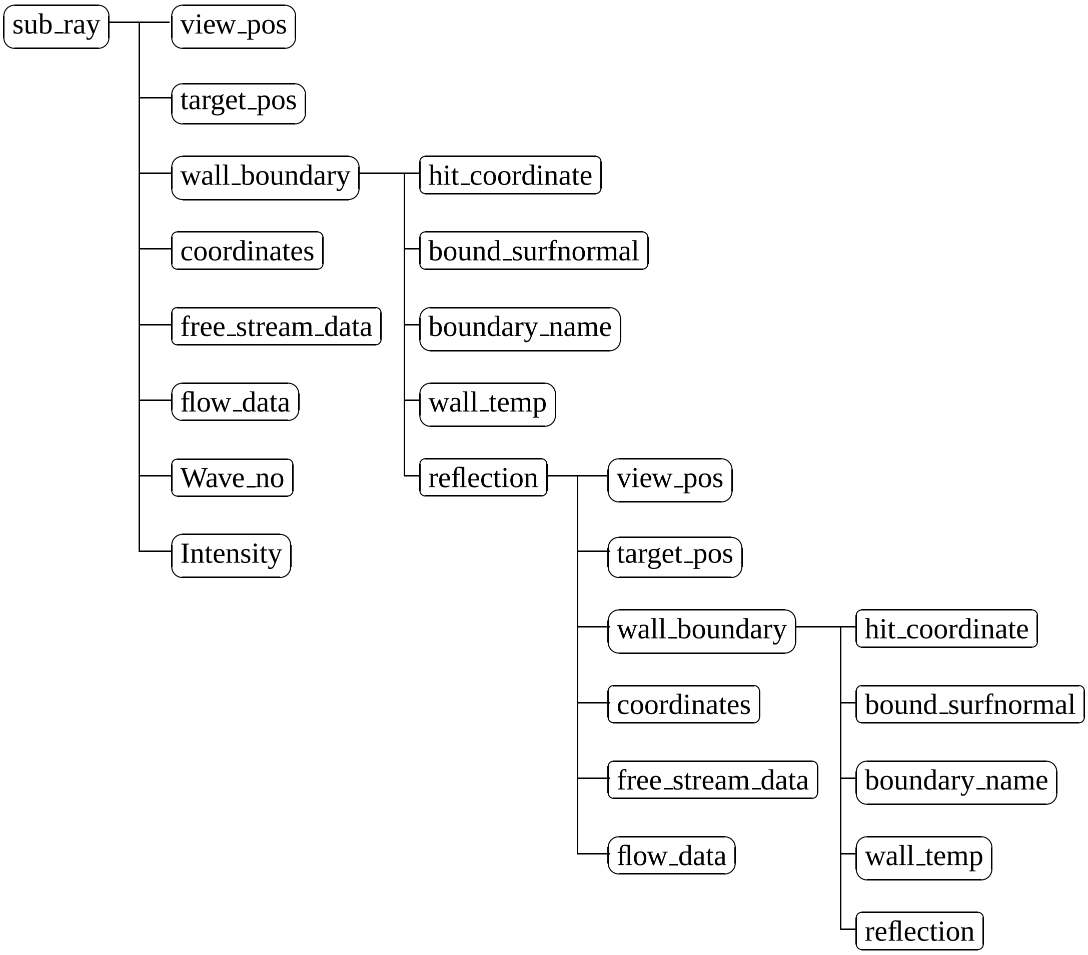


Figure 9. The structure for the data set *sub_ray* including two reflections



4.3 Main program of RAYFINDER

The main structure of RAYFINDER is presented in Figure 10. The rays are tracked through the mesh by stepping from cell to cell along the ray. This implies that the implementation is efficient even for grids with a large number of cells. When testing for ray intersections with the cell faces, a line-triangle intersection algorithm, presented in Ref. [12], is used. The way of handling reflections on walls is described in section 3.2.2.

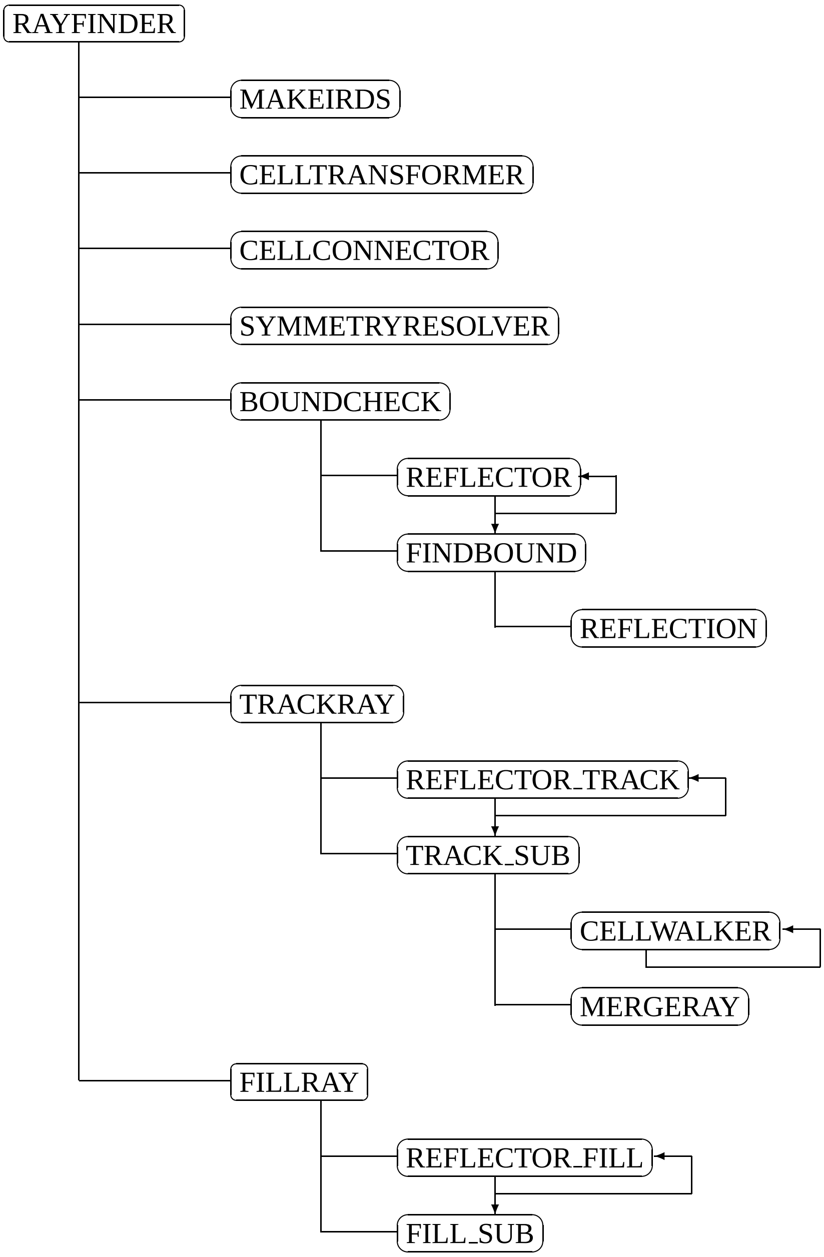
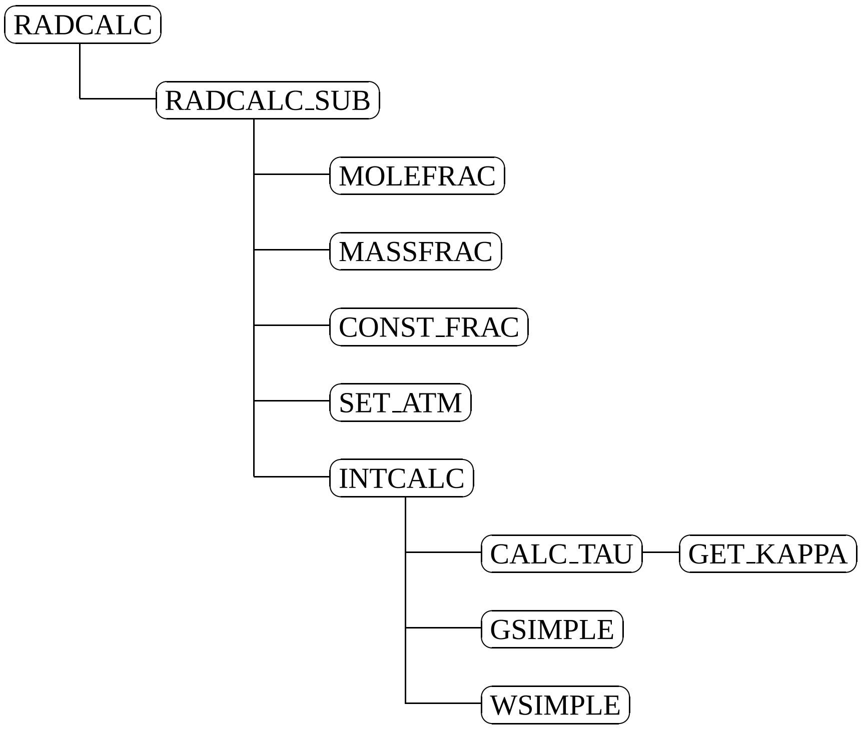


Figure 10. Program structure of RAYFINDER. The subroutines REFLECTOR, REFLECTOR_TRACK and REFLECTOR_FILL are recursive subroutines. The subroutines FINDBOUND, TRACK_SUB and FILL_SUB are called from the subroutines BOUNDCHECK, TRACKRAY, FILLRAY, respectively, and the recursive subroutines, indicated by the arrows.

4.4 Main program of RADCALC

The program structure of RADCALC is presented in Figure 11. For each ray generated by RAYFINDER and for each wave number specified in the *input file*, the intensity and the radiance are calculated as described in section 3 and in Ref. [3]. In the present version of SIGGE reflections on walls are included and described in section 3.2.1. The resulting spectral intensity has the unit [W/sr/cm^{-1}] and the spectral radiance has the unit [W/sr/cm^{-1}/m^2].

Figure 11. Program structure of RADCALC.



4.5 Subroutines in RAYFINDER

4.5.1 makeirds

The subroutine *makeirds* creates sub rays according to the number of rays requested in the field-of-view. Each sub ray is assigned an individual target position, giving individual aiming directions.

4.5.2 celltransformer

The subroutine *celltransformer* transforms all cells in the CFD-mesh to tetrahedra. The following element types are supported: Tetrahedron, hexahedron, prismatic element and pyramid.

4.5.3 cellconnector

The subroutine *cellconnector* stores information for each node, containing element numbers of the elements that the node is a member of. This information is later used in the subroutine *cellwalker*.

4.5.4 symmetryresolver

The subroutine *symmetryresolver* searches all CFD-boundary conditions for symmetry planes. If symmetry planes are found new coordinates are created representing the absent regions. One or two symmetry planes are supported, in the case of two symmetry planes perpendicular planes are required.

4.5.5 boundcheck

The subroutine *boundcheck* searches all CFD-boundaries for ray intersections. The intersections found is used in *trackray* as start and end points when the rays are tracked through the mesh. Additional data such as the direction of the surface normal is calculated and stored for each intersection. The recursive subroutine **reflector** is used to find the reflections in the walls by calling itself until the maximum number of reflections is reached or until no wall is hit. The subroutine **findbound** finds the intersections described above and the subroutine **reflection** calculates new view- and target positions, see section 3.2.2.

4.5.6 trackray

The subroutine *trackray* handles the tracking of the rays through the mesh. The intersections found in *boundcheck* are used and for each ray attempts are made to find a way through the cells in the mesh that connects the intersections. The recursive subroutine **reflector_track** is used to find the reflections in the walls by calling itself until the maximum number of reflections is reached or until no wall is hit. The subroutine **trackray_sub** tracks the ray through the mesh.

4.5.7 cellwalker

The subroutine *cellwalker* is a recursive routine that finds the way through the cells in the mesh given a start point and an end point. Starting in the cell at the start point the end point is searched by step wise walking to one of the neighbouring cells and checking for ray intersections with the new cell.

4.5.8 mergeray

The subroutine *mergeray* cleans up after *cellwalker* and creates the final representation of the rays.

4.5.9 fillray

The subroutine *fillray* interpolates the CFD-solution to the points on the ray, linear interpolation is used. A node based representation of the CFD-solution is required. The recursive subroutine **reflector_fill** is used to find the reflections in the walls by calling itself until the maximum number of reflections is reached or until no wall is hit. The subroutine **fillray_sub** interpolates the CFD-solution to the points on the ray.

4.6 Subroutines in RADCALC

4.6.1 radcalc_sub

This subroutine will step down in the structure of the *indata file*, see Figure 9, to find the initial radiation to be able to start the calculations as described in section 3.2.1. At each wall and at the view position the reflected radiance, the wall radiance and radiance from the gas is added. The radiance and the intensity is calculated , for each ray, in the subroutine *intcalc*. For each reflection ray the subroutine *set_atm* and one of the subroutines *molefrac*, *massfrac* or *const_frac* will be called, see also Appendix A.

4.6.2 molefrac

This subroutine calculates the partial density and pressure, for the different species, from the mole fraction given in the *indata file*.

4.6.3 massfrac

This subroutine calculates the partial density and pressure, for the different species, from the mass fraction given in the *indata file* and mole fraction given in the *input file*.

4.6.4 const_frac

This subroutine defines the partial density and pressure for different species if neither then mole fraction or the mass fraction is not defined in the *indata file*. Constant values given in the *input file* are used.

4.6.5 set_atm

In this routine the distribution of species in the atmosphere is set by using the values defined in the *input file*. For temperature, density and pressure either the free-stream-values or user-defined values are used.

Since the models used in the present version of SIGGE are not vaild for the atmosphere, the distribution of species should always be set to zero.

4.6.6 intcalc

This subroutines calculates the spectral intensity and the spectral radiance for each ray. The transmissivity needed is calculated in the subroutine *calc_tau*, which is used to calculated the intensity and radiance from gases and walls. The intensity and radiance from a gas are calculated in **gsimple** and from a wall in **wsimple**, as described in section 3.1.

4.6.7 calc_tau

For each cell and wave number, the subroutine *get_kappa* is called, which will get absorption coefficients for the different species from the *absorption data base file*, section 4.6.8 and Ref. [3]. The transmissivity is calculated as described in section 3.1.1.

4.6.8 get_kappa

This subroutine will read the absorption coefficients from the *absorption data base file*. Normally is mean absorptions coefficients given in the data base. The absorption coefficients are either given as pressure absorption coefficients, $\overline{\kappa}^k_{p\eta}$, or mass absorption coefficients, $\overline{\kappa}^k_{\rho\eta}$. These absorptions coefficients have to be multiplied with either the partial pressure or the partial density, Appendix A:

$$\overline{\kappa}^k_{\eta} = \overline{\kappa}^k_{p\eta} \times p_k \tag{28}$$

$$\overline{\kappa}^k_{\eta} = \overline{\kappa}^k_{\rho\eta} \times \rho_k \tag{29}$$

The units for $\overline{\kappa}^k_{p\eta}$ should be (atm $\times$ cm)$^{-1}$ and for $\overline{\kappa}^k_{\rho\eta}$ it should be m$^3\times$(kg $\times$ cm)$^{-1}$. It has to be given in the *input file* whether the data base contains pressure absorption coefficients or mass absorption coefficients.

The program will check if the temperature and wave number of interest is in the interval of the data base, if not the closest one will be used.

The program will use bilinear interpolation to find the absorption coefficients in the data base.

The partial pressure of interest or the partial density of interest will be found and multiplied with either the pressure absorption coefficients or the mass absorption coefficients as described above. Absorption coefficients for the different species specified will be returned to the subroutine *calc_tau*.

5 Pre- and post-processing programs

A couple of pre- and post-processing programs have been developed for SIGGE.

MARD Stands for Mean Absorption Coefficient Read Data. Calculates the mean absorption coefficients from data extracted by HITRAN and HITEMP [13], using the line strengths and the Lorentz broadening [14]. The output files from JHAWKS[2] can be used directely as inputfiles to MARD. The format of the output file is presented in Appendix E.

ffav2f Converts a CFD-solution with mesh on a format deliverd by VAC[3] from the program VOLSOL [15] to FFA-format.

ffae2f Converts a CFD-solution on a simple ascii format to FFA-format. A mesh, on FFA-format (unstructed), must exist.

postSIGGE Converts the FFA-format output file to ascii format to be read by, for example, MATLAB. Some alternatives can be choosen:

- Total spectra with intensity and radiance. All pixels/rays are added.
- Total spectra for different parts of the geometry.
- Spectra for every individual ray.
- IR-images.

[2] JAVA HITRAN ATMOSPHERIC WORKSATION; interface for HITRAN and HITEMP

[3] Volvo Aero Corporation, Trollhättan, Sweden

6 Verification

The verification was made for a jet engine exhaust pipe in two steps. In the first case, V1, a simpler geometry was used and radiation from gases and walls without reflections was considerd. In the second case, V2, the geometry was a little more complicated and reflections were included. The geometries and the CFD-solution was provided by VAC, who also provided IR-calculations to verify against. The VAC calculations were made using an 2D-IR-code, developed at VAC [16]. The two codes use the same band model, described in section 3.1 and section 3.1.1, but the implementations and the data base with absorption coefficients are different.

6.1 Description of verification cases

Both verification cases use the same base geometry, Figure 12, which is a simplification of the RM12 engine. In both cases, the view position is located 5 km away, giving parallell rays. The rays have an angle of 5° to the symmetry axis.

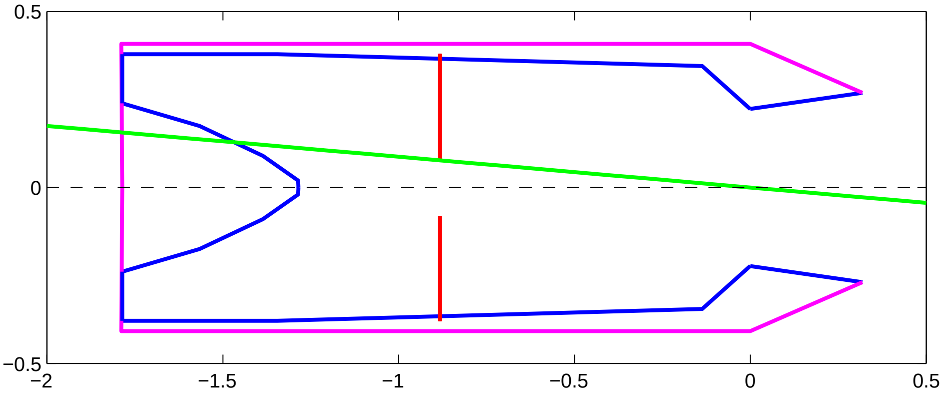


Figure 12. Sketch of the geometry for case V1 and V2. In verification case V1 onlythe the inner structure (blue) and the outer structure (purple) was defined. In verification case V2 the flame holders (red) were included. In the figure the central ray which has an angle of 5° to the symmetry axis is drawn in green.

In verification case V2, 10 radial flame holders and an anular flame holder were included in the geometry.

The same CFD-solution was used in both cases. The geometry was meshed using the mesh program GAMBIT 2.0.4 [4]. The temperature, the emissivity and the reflectivity of the walls were set in the *wall definition file* and values for different parts are summerized in Table 3. The reflectivity was set to 1 - the emissivity.

A field-of-view of 0.07°×0.07° was used, which covered the whole opening of the engine. The calculation was made in the wave number intervall 500 cm^{-1} - 5500 cm^{-1}, with a resolution of 25 cm^{-1}. 25×25 rays were used.

The species considered were CO_2 and H_2O. Only gas with temperature higher than 400 K was taken into account, since the contribution from gas with lower temperature is nearly neglectable. No atmospheric moderation was considered. The mean absorption coefficint data base was generated using MARD, described in section 5, with indata from HITEMP.

6.2 Result of verifications

When comparing spectra, only contribution from the inner structure is included.

[4] Software from Fluent Inc.

Table 3. Temperature and emissivities used on the different walls of the geometry. The reflectivity is defined as 1 - the emissivity, (ρ_{ss} = 1 - ε).

Wall	Temperature [K]	Emissivity†
Cone	900	0.7
Turbine	1120	1
Liner	530 - 658	0.7
Nozzle	658 - 678	0.7
Outside	200	1
Radiell flame holder	700	0.9
Circular flame holder	600	0.9

† Only used in V2, in V1 all emissivites were set to 1.

6.2.1 Verification case V1

In verification case V1, two spectra are compared. In Figure 13, all contributions are seen, while in Figure 14, contributions from the cone have been excluded.

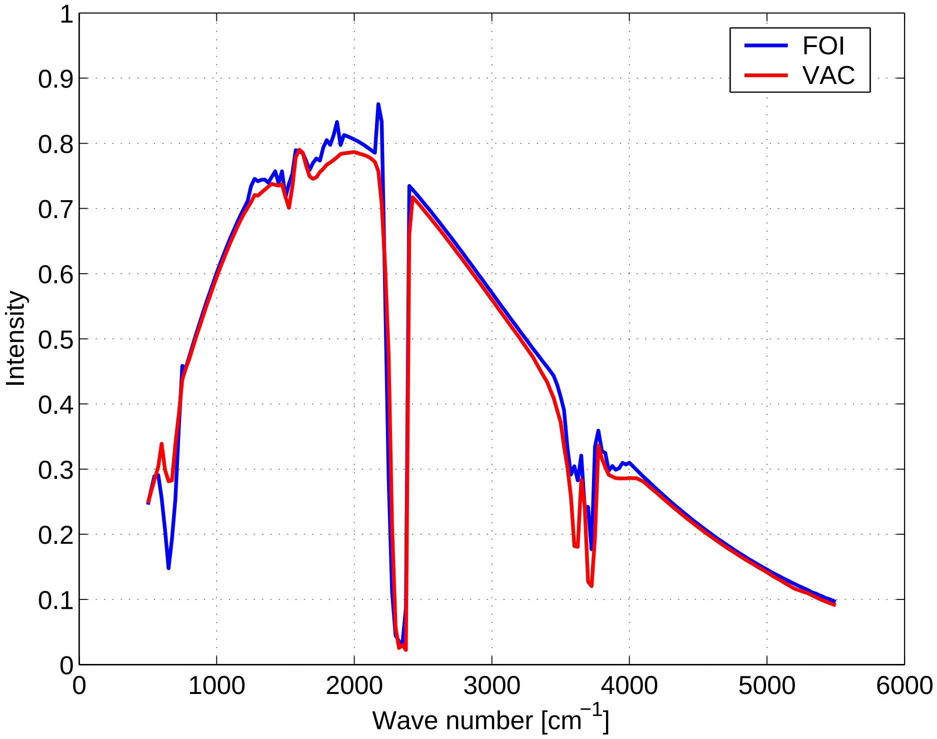


Figure 13. Verification case V1. Calculation made by FOI compared with calculation made by VAC. All parts inside are included.

As seen in Figure 13 and Figure 14, the agreement is excellent for most parts of the spectra. Differences are due to differences in the absorption coefficient data bases used. As seen in Figure 14, the peak around 2400 cm^{-1} is missing in the FOI calculation. This is due to the data base used, which is confirmed by a calculation with resolution 5 cm^{-1}, see Figure 15.

The result can also be presented as IR-radiance images, Figure 16. For each pixel the wave-number spectra has been integrated from 500 cm^{-1} to 5500 cm^{-1}.

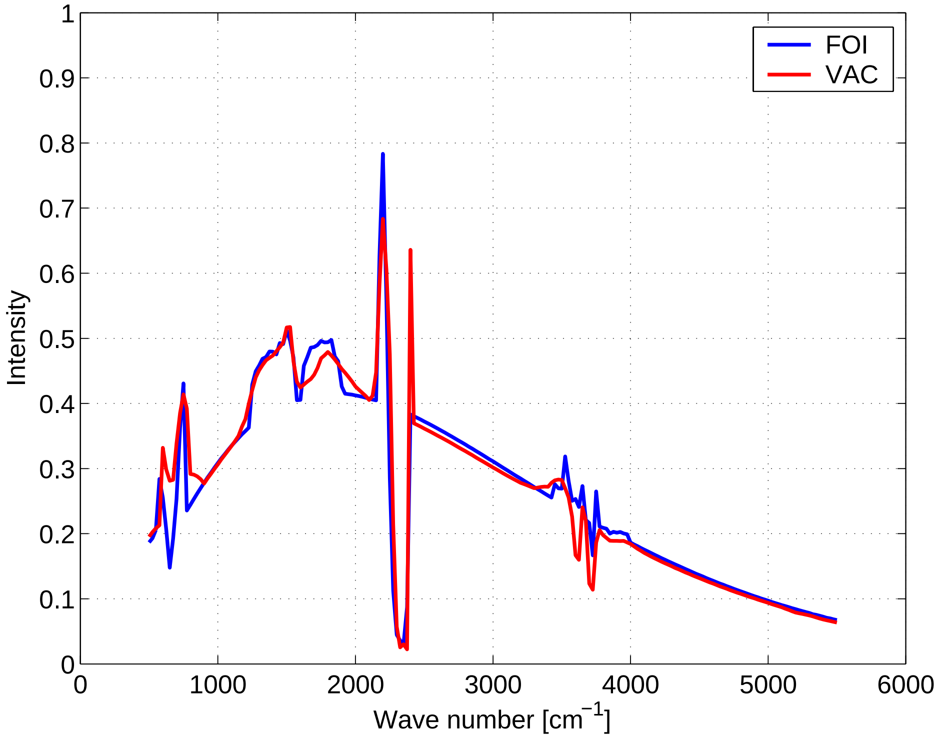


Figure 14. Verification case V1. Calculation made by FOI compared with calculation made by VAC. Contributions from the cone are excluded in these calculations.

6.2.2 Verification case V2

In verification case V2, four reflections are included in the calculations. In Figure 17, the result is presented together with calculations made by VAC.

In this verification case, the agreement is not as good as in verification case V1. This is probably due to the fact that the geometries and the meshes used in the FOI calculation and VAC calculation are not exactly the same. Further, the rays drawn through the mesh are not the same and, thus, the reflections are different.

IR-radiance images from verification case V2 are found in Figure 18.

6.2.3 Run times

All calculations were run on a Compaq GS160 computer[5]. Only one processor was used. The CPU times used for the different cases are presented in Table 4.

Table 4. Running (CPU) time for the different verification cases.

Case	# pixels	Resolution	Time
V1	25×25	25 cm^{-1}	10 min
	25×25	5 cm^{-1}	1 h
	100×100	25 cm^{-1}	4 h
V2	25×25	25 cm^{-1}	15 min
	100×100	25 cm^{-1}	13 h 46 min

In V1, no reflections were considered, while in V2 up to four reflections for each ray were included.

[5] Compaq GS160 has 16 Alpha EV67-processors on 733 MHz and 16 Gb of memory.

Figure 15. Same calculation as in Figure 14 for VAC, but in the FOI calculation a wave number resolution of 5 cm^{-1} is presented.

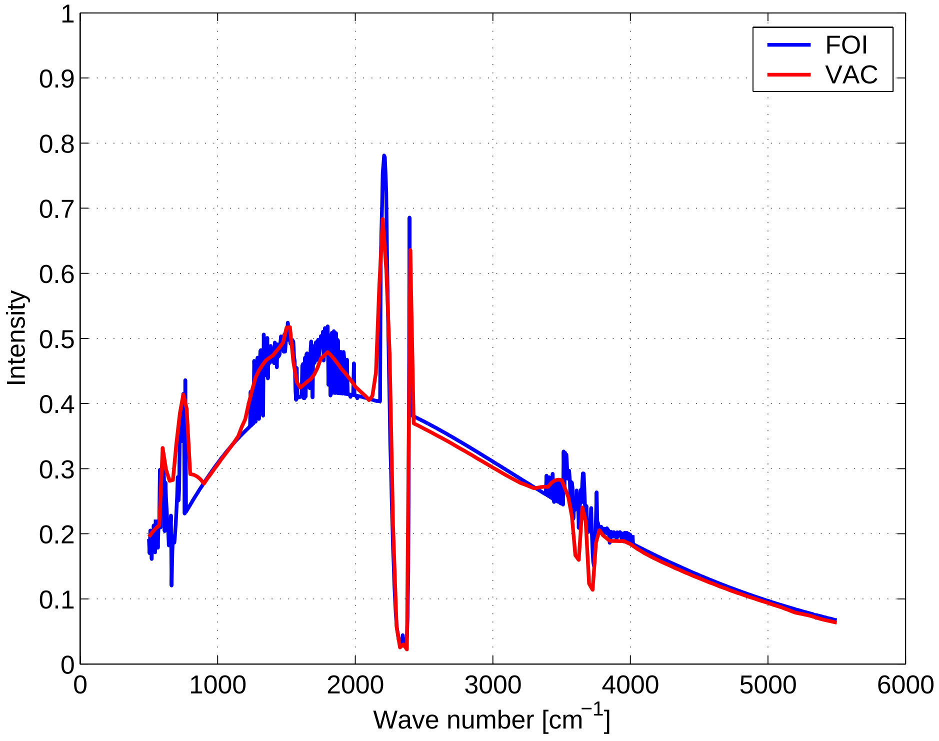
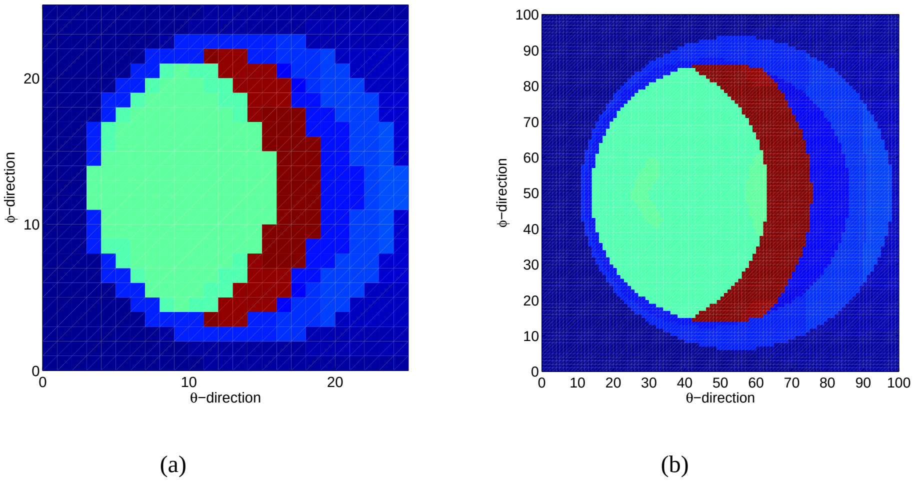


Figure 16. Verification case V1. IR-radiance image with (a) 25×25 pixels and (b) 100×100 pixels.



(a)

(b)

Figure 17. Verification case V2. Calculation made by FOI compared with calculation made by VAC.

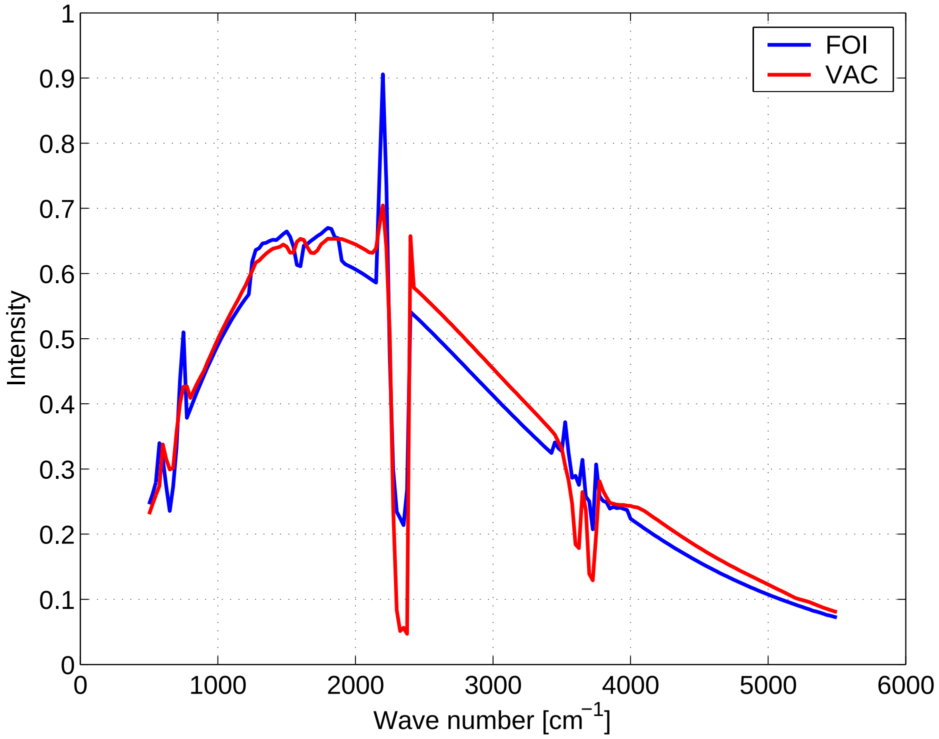
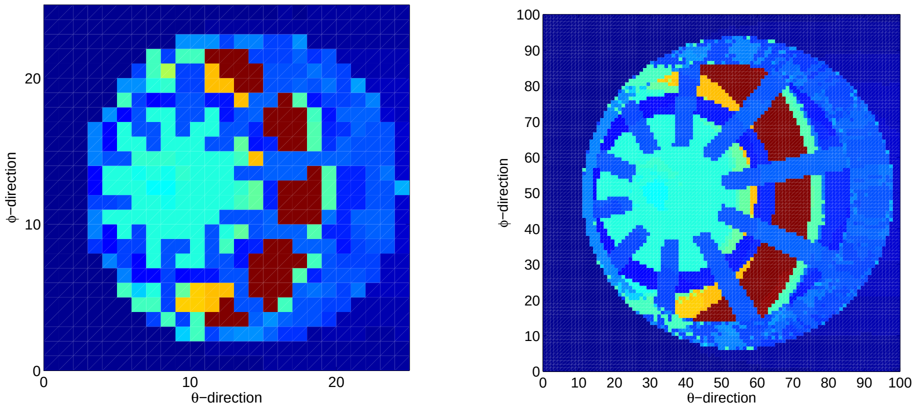


Figure 18. Verification case V2. IR-radiance image with (a) 25×25 pixels and (b) 100×100 pixels.



(a)

(b)

7 Conclusions and outlook

The IR-signature calculation code, SIGGE V1.0, can handle radiation from hot gases as well as radiation from walls including reflections. This report has described the theory used and how the theory has been implemented. The code has been verified against calculations made by VAC. The verification shows satisfactory agreements. Thus, SIGGE is now a code which could and should be used in real applications.

The band model used in the present code assumes an uniform gas, which, for most cases, is not true. The SNB model for a nonuniform gas (classical Curtis-Godson approximation) with an inverse-exponential distribution within the band [17] will be implemented in SIGGE shortly. The code will be further developed with improvements of the bandmodels used and with coupling between source radiation and the atmosphere. Validations with experimental data shall be done.

Also, to be able to fast evaluate new geometries the code has to be faster, thus parallel.

Acknowledgements

Johannes Johansson and Petter Eneroth (FFA) are acknowledged for making the meshes used in the verification.

The Department of Military Engines at Volvo Aero Corporation in Trollhättan is acknowledged for contribution with the verification geometry, solution and their own calculations. A special thanks for all fruitful discussions about all the models.

This work has been financed by the Swedish Defence Materiel Administration (FMV).

Appendix A

Distribution of species

As discussed in section 4.6.8, the partial absorption coefficients, $\kappa^k_{p/\rho,\eta}$, has to be multiplied with the partial pressure, p_k, or the partial density, ρ_k. Thus, these quantities must be known or must be possible to calculate.

In the program package SIGGE, three options are available by using one of the subroutines *molefrac*, *massfrac* or *const_frac*, see section 4.6.1.

A.1 Mole fraction

If the mole fraction, $x_{k,i}$, of each specie, k, is given for each cell i, in the CFD-solution, the partial pressure and partial density are simply given by:

$$\left\{ \begin{array}{c} p_{k,i} \\ \rho_{k,i} \end{array} \right\} = x_{k,i} \times \left\{ \begin{array}{c} p_i \\ \rho i \end{array} \right\} \tag{30}$$

where p_i and ρ_i are the total pressure and density in the i^{th} cell.

A.2 Mass fraction

The distribution of species in the CFD-solution could be given as a mass fraction, which is the fraction between the ambient air and the combustion gas. For simplicity, it is assumed that the ambient air will not contain species that will contribute to the IR-signature. The mole fractions are assumed to be constant in all cells and should be given in the *input file* of SIGGE, see section 4.1. The mole fractions for the different species in the combustion gas can, for example, be calculated with the program CET89 [18].

By using the mass fraction from the CFD-solution and the mole fraction, the concentration of species could be approximated for the i^{th} cell as:

$$\left\{ \begin{array}{c} p_{k,i} \\ \rho_{k,i} \end{array} \right\} = \textit{mass fraction for gas} \times \textit{mole fraction for species} \times \left\{ \begin{array}{c} p_i \\ \rho_i \end{array} \right\} \tag{31}$$

where, again, p_i and ρ_i are the total pressure and density in the i^{th} cell.

Doing this simplified calculation some errors will occur. When the ambient air is mixed in, the molecule weight will change, therefore, the mole fraction should be multiplied with the change in molecule weight. The mole fraction varies non-linearly with the equivalence ratio, which also will give rise to an error. These corrections are not implemented in the current version of SIGGE, therefore, the partial pressure and the partial density are impaired by a few percents error.

A.3 Constant fraction

If no distribution of species is given in the CFD-solution, a constant fraction for all cells can be given in the *input file*. For most cases, this is not recommended.

Appendix B

The input file

The *input file* is common for the modules RAYFINDER and RADCALC. In this file, some parameters are given to define different cases or scenarios. The input and output files are also given in this file.

Example of how a parameter should be defined is described in Figure 19.

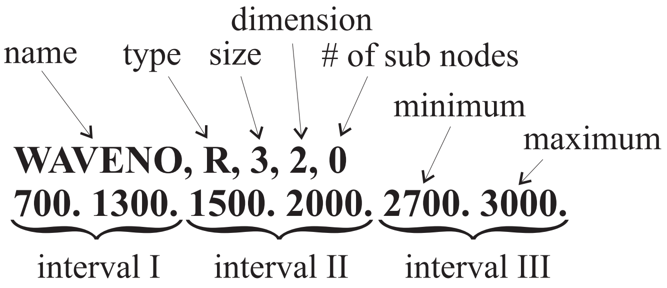


Figure 19. Example of the definition of one parameter in the *input file*. Here is the parameter WAVENO defined, which is of type *REAL*, has size 3 and dimension 2. In this case the number of intervals is defined with the size.

General parameters set in the *input file* are:

SIGGE_VERSION *Program version*

CASENAME *Case description name*

AUTHOR *work done by. . .*

For RAYFINDER, the following parameters are defined:

RUNRAYFINDER *if RAYFINDER should be ran or not*

- 0: No
- 1: Yes

FOV_THETA *field-of-view in the θ-direction (degrees)*

FOV_PHI *field-of-view in the ϕ-direction (degrees)*

NRAY_THETA *number of rays per viewpoint in the θ-direction*

NRAY_PHI *number of rays per viewpoint in the ϕ-direction*

NOREFL *number of reflections on walls*

WTEMP *use temperature on wall from CFD-solution or from wall definition file*

- 0: CFD-solution
- 1: Wall definition file

For RADCALC, the following parameters are defined:

RUNRADCALC *if RADCALC should be ran or not*

- 0: No
- 1: Yes

WAVENO *min and max of wave number, number of intervals should also be given*

DWAVENO *resolution which will define the number of wave numbers*

DATABASE_TYPE *the type or unit of the absorption coefficients*

- 1: mass absorption coefficients
- 2: pressure absorption coefficients

TRANSMODEL *the transmissivity model*

- 1: Simplest model, same for all gases

GASMODEL *type of gas model*

- 0: No gas model (gives intensity: $\mathrm{I}_{\eta}(gas)$ = 0)
- 1: Basic gas model

SPNAME *list of species, order should be the same as in the data base of absorption coefficients*

SPTYPE *gives on which form the distribution of species is given*

- 0: No distribution of species given in the *indata file*
- 1: Mass fraction given in *indata file*
- 2: Mole fraction of species given in *indata file* (not implemented)

SPDIS *distribution of species in gas if not given in the indata file*
(only if SPTYPE *set to 0)*

MOLEFRAC *needed to convert the mass fraction to pressure and density distribution of the species (only if* SPTYPE *is set to 0)*

ATMVAL *which temperature and pressure values should be used in the atmosphere*

- 0: Use free-stream-values
- 1: Use values set by ATM

ATM *temperature [K] and pressure [Pa] in the atmosphere, if free-stream-values not are used (i.e.* ATMVAL *is set to 1)*

SPATM *distribution of species in the atmosphere*

WALLMODEL *type of wall model*

- 0: No wall model (gives intensity: $\mathrm{I}_{\eta}(wall)$ = 0)
- 1: Basic wall model

ATMOMODEL *type of model for atmospheric background*

- 0: No atmosphere model (gives intensity: $\mathrm{I}_{\eta}(atmosphere)$ = 0)
- 1: Basic atmosphere model (not implemented)

Input and output files defined in the *input file*:

BCFILE *Boundary condition file*

MSHFILE *Mesh file*

SOLFILE *CFD-solution file*

RAYFILE *Ray definition file*

WALLFILE *Wall definition file*

ABSFILE *Absorption coefficient data base file*

RAYTEMPFILE *Ray data file. Contains information about the rays, including densities, temperature and such things. This file must exist before RADCALC starts, if it does not exists, it must be generated by RAYFINDER, i.e run RAYFINDER (set option in input file).*

OUTFILE *Output file with wave number, intensity and other things, see section 4.2.*

In the Tables 5, 6, 7 and 8, the descriptors for the parameters defined in the *input file* are described.

Table 5. The descriptors for the general parameters set in the *input file*.

name	type	size	dimension	sub nodes
SIGGE	N	0	0	34
SIGGE_VERSION	SI	1	1	0
CASENAME	LI	1	1	0
AUTHOR	LI	1	1	0

Table 6. The descriptors for RAYFINDER set in the *input file*.

name	type	size	dimension	sub nodes
RUNRAYFINDER	I	1	1	0
NRAY_THETA	I	1	1	0
NRAY_PHI	I	1	1	0
FOV_THETA	I	1	1	0
FOV_PHI	I	1	1	0
NOREFL	I	1	1	0
WTEMP	I	1	1	0

Table 7. The descriptors for RADCALC set in the *input file*.

name	type	size	dimension	sub nodes
RUNRADCALC	I	1	1	0
WAVE_NO	R	# of intervals	2	0
DWAVENO	I	# of intervals	1	0
DATABASE_TYPE	I	1	1	0
TRANSMODEL	I	1	1	0
GASMODEL	I	1	1	0
SPNAME	L	1	# of species	0
SPDIS	R	1	# of species	0
FIMAX	I	1	1	0
MOLEFRAC	R	1	# of species	0
ATMVAL	I	1	1	0
ATM	I	1	2	0
SPATM	R	1	# of species	0
WALLMODEL	I	1	1	0
ATMOMODEL	I	1	1	0

Table 8. The descriptors for the file names set in the *input file*.

name	type	size	dimension	sub nodes
BCFILE	L	1	1	0
MSHFILE	L	1	1	0
SOLFILE	L	1	1	0
RAYFILE	L	1	1	0
WALLFILE	L	1	1	0
ABSFILE	L	1	1	0
RAYTMPFILE	L	1	1	0
OUTFILE	L	1	1	0

Appendix C

The ray definition file

The sensor position and the spatial orientation of the sensor is specified by the user in the *ray definition file*. An arbitrary number of sensor cases can be used and for each case a sub-data set is defined. For each case the following parameters must be specified:

VIEWPOS *coordinates of the sensor position*

TARGETPOS *coordinates of a point on the sensors line of sight in the centre of the field-of-view*

THETADIR *vector defining $\hat{\theta}$-direction*

PHIDIR *vector defining $\hat{\phi}$-direction*

In Table 9 is the descriptors described.

Table 9. The descriptors for parameters set in the *ray definition file*.

name	type	size	dimension	sub nodes
RAYDATA	N	0	0	# of ray sets
RAY	N	0	0	4
VIEWPOS	RD	3	1	0
TARGETPOS	RD	3	1	0
THETADIR	RD	3	1	0
PHIDIR	RD	3	1	0

Appendix D

The wall definition file

Different walls in the case can have different properties. These properties are defined in the *wall definition file*, where each wall as a unique name to which the properties are associated. For each wall a sub-data set is defined and for each wall the following parameters must be specified:

NAME *Unique name of the wall. Max 12 character.*

TEMPERATURE *Temperature on the wall given as* $T = a + b \cdot x$.

EMISSIVITY *Emissivity for the wall given as* $\varepsilon = a + b \cdot \alpha + c \cdot \alpha^2$, *where* α *is the angle between the wall normal and the ray.*

REFLECTIVITY *Specular reflectivity for the wall given as* $\rho = a + b \cdot \alpha + c \cdot \alpha^2$, *where* α *is the angle between the wall normal and the ray.*

In Table 10 is the descriptors described.

Table 10. The descriptors for parameters set in the *wall definition file*.

name	type	size	dimension	sub nodes
WALLDATA	N	0	0	# of walls
WALL	N	0	0	4
NAME	LI	1	1	0
TEMPERATURE	RD	2	1	0
EMISSIVITY	RD	3	1	0
REFLECTIVITY	RD	3	1	0

Appendix E

The absorption data base file

To be able to calculate the IR-intensity, the mean absorption coefficients, $\overline{\kappa}_\eta$, have to be known, see Eq. (12). The absorption coefficients are different for different species and are dependent on the temperature and wave number. Thus, in the *absorption data base file*, the different species have their own sets of absorption coefficients, which are connected to temperature and wave number. The structure of the file is presented in Figure 20.

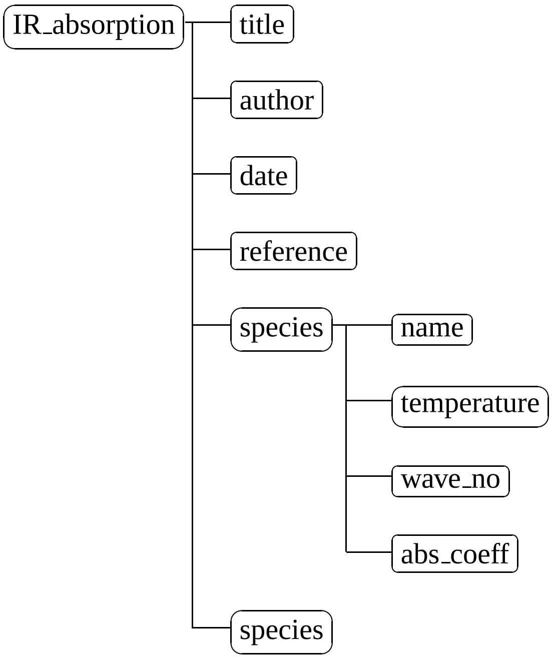


Figure 20. The structure of the absorption data base file.

The mean absoprption coefficients can be extracted with the program MARD from files generated by HITRAN and/or HITEMP, see section 5.

References

[1] M. Andersson. *SIGGE V1.0β, Users Guide*, FOI-R-0554-SE, (2002).

[2] M. Andersson. *Validation of the Computer Program SIGGE against Spectral IR-measurements on an Engine Test Rig*, FOI-R-0555-SE, (2002).

[3] M. Andersson. *Derivation of Equations used by the Computer Program SIGGE for Calculating IR-intensity*, FOI-R-0553-SE, (2002).

[4] C.B. Ludwig et. al. *Handbook Infrared Radiation from Combustion Gases*, NASA N73-27, (1973).

[5] H. Samuelsson. *Theoretical Calculation of IR Radiation Intensity from Jet Engines and of Atmospheric Trasmission*, Paper presented at the Third European Symposium on Military Infrared RRE. Malvern, England, (May 1971).

[6] R. Kihlén. *Calculation of IR-radiation from Jet Engines in the wavelength area 1.8 - 16.7 μm*, SAAB RA-MIL-3 B145, (1973). (In Swedish).

[7] G. J. Zissis (editor). *Sources of Radiation, The Infrared & Electro-Optical Systems Handbook vol. 1* Infrared Information Analysis Center and SPIE Optical Engineering Press, (1996).

[8] S. Wallin. *Standardized Data Format, version 1*, FFAP-A-950, (1992).

[9] T. Sjögren and P. Eliasson. *Description and Validation of EDGE*, FFA TN 1998-61, (1998).

[10] M. Sillén and A. Karlsson. *Användarhandledning Spider/Tritet/Edge*, FFO-2001-0062, (2001).

[11] *EDGE V2.3 GUIDE*, internal users guide at the Department of Computational Aerodynamics, Aeronautics Division (FFA), FOI, (2002).

[12] R. Löhner and P. Parikh. *Generation of three-dimensional unstructured grids by the advancing-front method*, International Journal for Numerical Methods in Fluids, vol. 8, pp.1135-1149, (1988).

[13] *The HITRAN Database*, http://www.hitran.com.

[14] L.S. Rothman et. al. *The HITRAN Molecular Spectroscopic Database and HAWKS (HITRAN Atmospheric Workstation): 1996 Edition*, Journal of Quantitative Spectroscopy and Radiative Transfer, vol. **60**, pp. 665-710, (1998).

[15] L.E. Eriksson. *A Third Order Accurate Upwind-Biased Finite-Volume Scheme for Unsteady Compressible Viscous Flow*, Technical report, VAC Report 9370-154, Volvo Aero Corporation, Sweden (1990).

[16] M. Ferring. *Numerical Analysis of IR-emission from Exhaust Plumes and Aircraft Hot Parts*, Technical report, VAC Report 331-217, Volvo Aero Corporation, Sweden (1973).

[17] J. Taine and A. Soufiani. *Gas IR Radiative Properties: From Spectroscopic Data to Approximate Models*, Advances in Heat Transfer, vol. **33**, pp. 295-414, (1999).

[18] *CET89 - Chemical Equilibrium with Transport Properties, 1989*, COSMIC, The University of Georgia.

Issuing organisation FOI – Swedish Defence Research Agency Division of Aeronautics, FFA SE-172 90 STOCKHOLM	Report number, ISRN FOI-R–0828–SE	Report type Methodology report
	Month year January 2003	Project number G893
	Customers code 3. Flygteknisk forskning	
	Research area code 7. Vehicles	
	Sub area code 73. Aeronautical Research	
Author(s) Marlene Andersson	Project manager Marlene Andersson	
	Approved by Bengt Winzell Head, Computational Aerodynamics Department	
	Scientifically and technically responsible Marlene Andersson	

Report title

SIGGE V1.0
Theory, Implementation and Verification

Abstract

To be able to predict the IR-signature of an aircraft, the heat radiated, in form of IR-radiation, has to be calculated. A program package, SIGGE, has been developed for this purpose. The already existing code has been improved. The code can now handle radiation from hot gases and from walls with complex geometries, including specular reflections.

This report describes some key equations used and a detailed description of SIGGE is presented. The code has been verified against other calculations with satisfactory result

Keywords

IR-signature, SIGGE, radiative heat transfer, documentation, verification

Further bibliographic information

ISSN 1650-1942	Pages 55	Language English
	Price Acc. to price list	
	Security classification Unclassified	

Utgivare Totalförsvarets Forskningsinstitut – FOI Avdelningen för Flygteknik, FFA SE-172 90 STOCKHOLM	Rapportnummer, ISRN FOI-R–0828–SE	Klassificering Metodrapport
	Månad år Januari 2003	Projektnummer G893
	Verksamhetsgren 3. Flygteknisk forskning	
	Forskningsområde 7. Bemannade och obemannade farkoster	
	Delområde 73. Flygteknisk forskning	
Författare Marlene Andersson	Projektledare Marlene Andersson	
	Godkänd av Bengt Winzell Chef, Institutionen för beräkningsaerodynamik	
	Tekniskt och/eller vetenskapligt ansvarig Marlene Andersson	

Rapporttitel

SIGGE V1.0
Teori, Implementering och Verifiering

Sammanfattning

För att kunna förutsäga IR-signaturen för en flygande farkost måste värmestrålningen, i form av IR-strålning, från farkosten beräknas. Ett programpaket, SIGGE, har utvecklats för detta ändamål. Den redan existerande koden har förbättras. Koden kan nu hantera strålning från varma gaser samt strålning från väggar med komplexa geometrier, inklusive spekulära reflektioner.

Den här rapporten beskriver några nyckelekvationer och en detaljerad beskrivning av SIGGE presenteras. Koden har verifierats mot andra beräkningar med ett tillfredställande resultat.

Nyckelord

IR-signatur, SIGGE, dokumentation, verifiering

Övriga bibliografiska uppgifter

ISSN 1650-1942	Antal sidor 55	Språk Engelska
Distribution enligt missiv	Pris Enligt prislista	
	Sekretess Ej hemlig	