

MASTER

Benchmarks for radiation heat transfer impact of specular - angle dependent reflection and polarization

Wijnen, M.

Award date:
2020

[Link to publication](#)

Disclaimer

This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

BENCHMARKS FOR RADIATION HEAT TRANSFER:
IMPACT OF SPECULAR - ANGLE DEPENDENT
REFLECTION AND POLARIZATION

GRADUATION REPORT

Master Mechanical Engineering
Research Group Energy Technology

Student M. Wijnen
ID Number 0888043
Thesis supervisor G. Nakiboglu
Thesis supervisor J. van Schijndel

13th February 2020

Contents

Problem Description	2
Research Results	2
Journal Paper	4
A Specular Reflection Benchmarks	18
A.1 Introduction	18
A.2 Concentric Cylinders	20
A.3 Isothermal Parallel Plates	28
A.4 Wedge with Conducting Plates	37
B Square Passage including Angle Dependent Reflection and Polarization	50
B.1 Refinement studies	50
B.2 Analytical Approximation	53
B.3 Numerical Model	62
C Experimental Validation	66
C.1 Analytical Model Transmission Metal Tube	66
C.2 Calculation of Correction Factor γ	68
C.3 Numerical Model	70
D Proof of Concept Experiment	72
Code of Scientific Conduct	74

Problem Description

At ASML there is an increasing demand for thermal stability, in order to optimize the performance of the high tech machines for the semiconductor industry. Many important design decisions are based on computational models. This is due to the limited availability of experimental data and measurements of the complex systems.

In the latest EXE machines, which make use of an extreme ultraviolet source, a large fraction of the critical components and processes operate in a vacuum environment. In the NXT machines the thermal behaviour of systems can be controlled by the use of gas flows. However this is not an option in the EXE machines. This increases the significance of radiation heat transfer on the thermal behaviour in systems.

Within ASML there is a need for accurate computational tools to model the influence radiation heat transfer on the thermal behaviour. In order to assess the quality and performance of these tools, benchmarks for complex radiation heat transfer are required. These benchmarks can be of analytical or experimental results of radiation problems, where the influence of specular and angle dependent reflection is significant.

The goal of this research is to formulate benchmarks and use them to validate the multiphysics software package COMSOL.

Research Results

In literature both analytical as well as experimental results were found in literature for problems where complex radiation is of influence. Two modules of in the software package COMSOL were used to model the benchmark problems. The numerical results align very well with the analytical and experimental results of the benchmark problems.

Three benchmarks are formulated that consist of analytical solutions of radiation problems including specular reflection. The concentric cylinder benchmark describes the total heat flux between two cylinders. This is a relatively simple problem, where the heat flux for both specular and diffuse reflection can be described using one equation [1]. The isothermal parallel plates benchmark consists of an analytical solution for both the total heat flux as well as the distribution of the heat flux. An analytical solution for both specular as diffuse reflection is presented [2-4]. The third benchmark problem describes a wedge with conducting plates. Aside from the heat flux, is the temperature distribution along the plates calculated analytically as well [5]. This problem can be used to verify the coupling between radiation and conduction in computational tools.

The *Surface-to-Surface Radiation* module of COMSOL is used to numerically evaluate the benchmarks. By comparing numerical with the analytical results the models are verified.

A fourth benchmark has been formulated which describes the influence of angle dependent reflection and polarization on the radiation heat transfer. The benchmark consists of an analytical approximation of the transmission through a square passage. This problem was first described by Edwards and Tobin [6], however the results presented in the paper were not fully converged. By replicating the model and improving the solution method, the results were evaluated with a higher accuracy.

The benchmark was replicated in COMSOL using the *Ray Optics* module, which uses a ray tracing algorithm. The reflection of the walls in the models is defined by the Fresnel equations. The results of the numerical model approach the analytical results with a maximal error of 4.2%.

An experiment has been conducted by Miranda [7], which evaluates the transmission through a metal tube. The reflection of the tube walls depend on the angle of incidence, polarization state and the wavelength of the radiation. The experiment is modeled in COMSOL using the same ray tracing algorithm as for the square passage benchmark, only adding wavelength dependency. The transmission is evaluated for different length to diameter ratios of the metal tube. The trend of the decrease in transmission for longer tube lengths is accurately described by the numerical model. There is an offset between the numerical and experimental results of approximately 5%.

Benchmarks for Radiation Heat Transfer: Impact of Specular - Angle Dependent Reflection and Polarization

M. Wijnen

Graduate student

Energy Technology and Fluid Dynamics

Department of Mechanical Engineering

Eindhoven University of Technology

Eindhoven, 5600 MB

The Netherlands Email: mathieu.wijnen@asml.com

J. van Schijndel

Thermal Analyst

ASML

Email: jos.van.schijndel@asml.com

Gunes Nakiboglu

Assistant Professor

Energy Technology and Fluid Dynamics

Eindhoven University of Technology

Eindhoven, 5600 MB

The Netherlands

Email address: g.nakiboglu@tue.nl

In the semiconductor industry there is an increasing demand for thermal stability to optimize the performance of high tech machines. This results in a need for computational tools that are able to accurately model radiation heat transfer, with the ability to include specular and angle dependent reflection and take the influence of the polarization into account. This paper presents three analytical solutions of radiation problems that include specular reflecting surfaces. These problems are used as benchmark problems to verify the Ray Shooting algorithm of the multiphysics tool COMSOL. An analytical approximation model of a square passage is formulated that shows the impact of taking angle dependent reflectivity and polarization into account. The converged results of this model are used to verify a model created with the Ray Optics module of COMSOL. This verified tool is validated by replicating an experiment and comparing the numerical results with the experimental data.

Nomenclature

a Function defined in appendix B

A Surface area, m^2

b Function defined in appendix B

c_0 Speed of light in vacuum, $2.99792458 \cdot 10^8$ m/s

C_1 Radiation constant, $0.59552138 \cdot 10^{-16}$ W · m^2 /sr

C_2 Radiation constant, 0.014387752 m · K

D Passage width, tube diameter, m

D_{i-j} Rotation matrix

E Energy matrix

$E_{\lambda b}$ Emitted blackbody energy flux, W/m^2

f_ϕ Function defined in appendix A

F View factor

G_ϕ Function defined in appendix A

h Distance between plates, m

I Intensity matrix

J_ϕ Function defined in appendix A

k Absorptive index

K_ϕ Function defined in appendix A

Ker Radiation Transfer Kernel

L Length, m

M_ϕ Function defined in appendix A

n Refraction index

\tilde{n} Complex refraction index

N Number of reflections

N_c Conduction parameter

\mathbf{p} Unit vector parallel to plane of incidence

P Power measured in experiment, W

P_ϕ Function defined in appendix A

q Local heat flux, W/m^2

Q Total heat flux, W/m

r Radius, m

r_s, r_p Reflection coefficient

r_e Electrical resistivity, $ohm \cdot cm$

R Power reflection coefficient

s Distance from entrance to end plane, m

\mathbf{s} Unit vector perpendicular to plane of incidence

t	Plate thickness, m
T	Temperature, K
T_b	Temperature at base of wedge, K
x	Coordinate, m
y	Coordinate, m
α	Absorption coefficient
$\beta(\xi)$	Dimensionless irradiation
β	Rotation of the polarization planes, rad
δ	Phase shift, rad
γ	Experiment correction factor
γ	Dimensionless distance between plates
ε	Emission coefficient
ζ	Dimensionless coordinate
κ	Thermal conductivity, $W/(m \cdot K)$
λ_0	Radiation wavelength through vacuum, m
μ_0	Permeability of vacuum, $4\pi \cdot 10^{-7} Wb/(A \cdot m)$
θ	Dimensionless temperature
θ_i	Angle of incidence, rad
θ_t	Angle of transmission, rad
ξ	Dimensionless coordinate
ρ	Reflection coefficient
σ	Stefan-Boltzmann constant, $W/(m^2 \cdot K^4)$
τ	Transmission
ϕ	Wedge opening angle, rad

Subscripts

a	Analytical
$diff$	Diffuse reflection
exp	Experimental results
n	Normal to the plane surface
num	Numerical
p	Parallel polarized component
pp, ss, uu, uv, vu, vv	Reflection matrix elements
s	Perpendicular polarized component
$spec$	Specular reflection
u	Third Stokes parameter
v	Fourth Stokes parameter

1 Introduction

In the semiconductor industry high precision engineering is required to achieve high accuracy. Temperature and thermal expansion need to be measured and controlled on the nanometer and millikelvin scale, respectively. Due to the high complexity of the systems and components, there is limited availability of measurements or experimental results. A lot of decisions are made based on computational models and simulations.

A number of critical components in lithography machines operate in a vacuum environment. This means that the influence of convection decreases significantly and radiation heat transfer is a more important factor in determining temperatures and thermal expansion in systems. This requires accurate computational tools for calculating the radiation heat transfer. The most common computational methods make use of gray surfaces, which means that a number of assumptions are employed to simplify the calculations and decrease computation time. For gray surfaces, properties are independent of the wavelength, direction and polarization state of the

irradiation.

Due to improved functionality of computational tools, it has become possible to avoid the mentioned limitations. These tools need to be of high quality to meet the requirements for precision engineering. In order to assess the quality of these tools, the tools need to be verified with the use of benchmarks.

This paper presents a number of analytical calculations and results for radiation problems with different complexity of the physics. An analytical calculation of a problem taking angle dependent reflection and the influence of polarization into account is provided. Results of the analytical calculations are presented that can be used to assess the quality of computational tools.

Section 2 of this paper presents three analytical calculations of radiation problems including specular reflecting surfaces [1–5]. The results of these analytical calculations are used as a benchmark to verify the *Ray Shooting* algorithm in the *Surface-to-Surface Radiation* module of the software package COMSOL.

Edwards and Tobin [6] have formulated an analytical approximation for a radiation problem, which includes angle dependent reflection and takes the influence of polarization into account. However the results of this calculation presented in the paper [6], are not fully converged. In section 3 the converged results of the analytical approximation are provided. A ray tracing model is developed in COMSOL and the numerical results are compared to the results of the analytical approximation.

The analytical calculation presented in section 3 has been used to investigate the impact of including the angle dependent reflection and polarization in the models. It is found that for a high aspect ratio the impact of both physics becomes more significant.

Section 4 describes the experiment conducted by Miranda [7] and compares the experimental with numerical results obtained using the ray tracing model, which is verified in section 3. The numerical results replicate the influence of increasing the aspect ratio very well.

2 Benchmarks including Specular and Diffuse Reflecting Surfaces

For most of the tools used to calculate heat transfer problems the default way to model the reflection is diffusely. However when the wavelength of the radiation is significantly larger than the surface roughness a surface is considered optically smooth, which means that the radiation is reflected specularly [8]. In specular reflection the angle of reflection (θ_r) is equal to the angle of incidence (θ_i), which can influence the heat flux [9].

Typical temperatures inside lithography machines range from around $20^\circ C$ to a couple of hundreds degrees. According to *Wiens formula* the wavelength of the majority of the emitted radiation will be in the range of $3 - 50 \mu m$. The roughness of the surfaces in these machines is in general smaller than $3 \mu m$. This means a large number of surfaces reflect thermal radiation specularly.

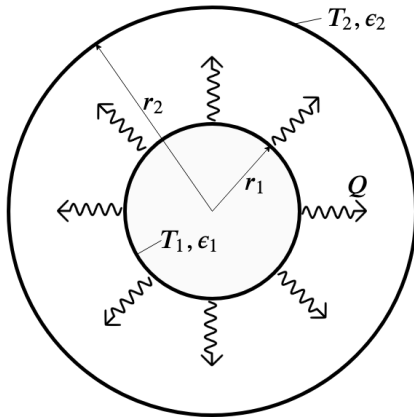


Fig. 1. A schematic representation of the concentric cylinder problem. The total heat flux, Q , between the cylinders is evaluated.

When only diffuse reflection is considered, the radiation heat transfer can be calculated with the use of view factors. This is not possible anymore when specular reflecting surfaces are included in the models, since the direction of the radiation becomes important.

In this section three benchmarks are described that can be used to verify the functionality of computational tools that include specular reflecting surfaces in the simulation. This can be done by creating computational models of the benchmarks and comparing the numerical results with the analytical solution.

For each of the benchmarks a 2D-model is created in the *Surface-to-Surface Radiation* module of the software package COMSOL. The *Ray Shooting* algorithm is used to be able to include specular reflecting surfaces [10].

The accuracy of the result is strongly dependent on the mesh and settings used. The settings in COMSOL that influence the accuracy are the ray resolution, tolerance and maximum number of adaptations. The ray resolution is the amount of rays emitted per element, the tolerance is the threshold of the minimum intensity a ray must have to be included in the calculation and the maximum number of adaptations describes how many times the tiling can be adapted [10].

The three benchmark problems described in sections 2.1 until 2.3 increase in complexity. In the concentric cylinders benchmark the total heat flux between two cylinders is evaluated. The parallel plates benchmark also evaluates the heat flux distribution along the plates. For the wedge benchmark the temperature distribution is calculated as well.

2.1 Concentric Cylinders

The problem of the first benchmark is described in [1], it consists of two infinitely long concentric cylinders, as shown in figure 1. The inner and outer cylinder are both at a constant temperature (T) and have a constant emission coefficient (ϵ). The total heat flux (Q) between the cylinders can be calculated analytically, as shown in the equations 1 and 2. Equation 1 holds when the surface of the outer cylinder reflects radiation diffusely, equation 2 is valid when this sur-

Table 1. Analytical solution and numerical results of the concentric cylinders problem for $T_1 = 500K$, $T_2 = 300K$ and $r_1 = 0.1m$. The total heat flux from the inner to the outer cylinder is calculated both analytically (Q_a) and numerically (Q_{num}), for both diffuse and specular reflection of the outer cylinder. The results are shown for four different parameter sets with different values for the outer cylinder radius (r_2) in meters and the emission coefficients of both cylinder surfaces (ϵ). The percentage difference between numerical and analytical results (ΔQ) is also given.

Parameters			Diffuse			Specular		
r_2	ϵ_1	ϵ_2	Q_a [W]	Q_{num} [W]	ΔQ [%]	Q_a [W]	Q_{num} [W]	ΔQ [%]
0.15	0.5	0.1	242.3	242.3	3.1e-3	176.2	176.6	0.25
0.15	0.9	0.8	1516.8	1516.8	1.2e-3	1424.0	1424.4	0.029
0.6	0.5	0.1	553.7	553.7	6.1e-3	176.2	178.1	1.1
0.6	0.9	0.8	1681.3	1681.3	1.3e-3	1424.0	1426.4	0.17

face reflects radiation specularly. The type of reflection of the inner cylinder does not influence the calculation of the heat flux.

$$Q_{diff} = \frac{A_1 \sigma (T_2^4 - T_1^4)}{\frac{1}{\epsilon_1} + \frac{A_1}{A_2} \left(\frac{1}{\epsilon_2} - 1 \right)} \quad (1)$$

$$Q_{spec} = \frac{A_1 \sigma (T_2^4 - T_1^4)}{\frac{1}{\epsilon_1} + \frac{1}{\epsilon_2} - 1} \quad (2)$$

with $A = 2\pi rL$.

In the software package COMSOL this problem is simulated in 2D using the *Surface-to-Surface Radiation* module. The problem is modeled with both a diffuse and specular outer cylinder surface. In order to include specular surfaces in the model the *Ray Shooting* algorithm is used.

The cylinder surfaces are subdivided into finite elements, to create a mesh of a total of 1280 elements. The ray resolution is equal to 512, the ray tolerance is 10^{-5} and the maximum number of adaptations is 5.

Table 1 shows the total heat flux calculated analytically alongside the numerical results for four different parameter sets. The results for diffuse reflection are in general more accurate than these for specular reflection. The error increases when the distance between the cylinders is increased and when the emission coefficient is decreased, thus when more reflections occur.

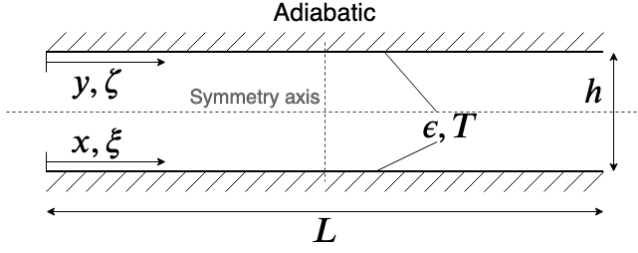


Fig. 2. A schematic representation of the isothermal parallel plates problem. Both the local as well as the total heat loss from the plates to the surroundings is evaluated.

2.2 Isothermal Parallel Plates

For the second benchmark problem the local heat loss along two isothermal parallel plates is calculated. This problem was first described by Sparrow [2,3] for diffusely reflecting surfaces. An analytical solution considering specular reflecting surfaces was later described by Eckert and Sparrow [4].

A schematic representation of the problem is given in figure 2. Both plates have the same temperature (T) and emission coefficient (ϵ), which means that the problem is symmetrical. The heat is lost to the surroundings, which are the sides of the geometry. The width of the plates is indicated by L and the distance between the plates by h .

As shown in figure 2 the local coordinates along the plates are described by x and y . In order to make the calculation dimensionless, the dimensionless coordinates ξ and ζ are introduced, as well as the height to width ratio γ .

$$\xi = \frac{x}{L}, \quad \zeta = \frac{y}{L}, \quad \gamma = \frac{h}{L} \quad (3)$$

In the paper by Sparrow et al. [3] an analytical expression is derived that describes the dimensionless heat loss distribution along one of the plates when the reflection is considered diffuse. Equation 4 describes the dimensionless heat loss along the width of the plates, for diffuse reflecting surfaces.

$$\frac{q(\xi)}{\epsilon\sigma T^4} = \frac{1 - \beta(\xi)\epsilon}{1 - \epsilon} \quad (4)$$

In this equation, $\beta(\xi)$ is defined as the dimensionless combined radiant flux, which is a sum of the emitted radiation and the reflected incident radiation. An expression for the dimensionless combined flux is shown in equation 5.

$$\beta(\xi) = 1 + \frac{\rho\gamma^2}{2} \int_0^1 \beta(\zeta) \frac{1}{[(\zeta - \xi)^2 + \gamma^2]^{3/2}} d\zeta \quad (5)$$

The combined flux of the bottom plate is a function of the combined flux of the top plate. Since the problem is symmetrical these distributions are equal: $\beta(\xi) = \beta(\zeta)$. Equation 5 can be solved numerically, through a process of iteration.

Eckert and Sparrow [4] defined an analytical calculation of the heat loss for the same geometry for specular reflecting surfaces. Equation 6 was derived for the dimensionless heat flux along the width of the plate, for specular reflecting surfaces.

$$\frac{q(\xi)}{\epsilon\sigma T^4} = 1 - \frac{\alpha}{2} \sum_{N=0}^{\infty} \left(\rho^N \left[\frac{\xi}{\sqrt{\xi^2 + (N+1)^2\gamma^2}} + \frac{1 - \xi}{\sqrt{(1 - \xi)^2 + (N+1)^2\gamma^2}} \right] \right) \quad (6)$$

In this equation α represents the absorption coefficient and is equal to the emission coefficient, as prescribed by Kirchhoff's Law. In this equation the heat loss is defined as 1 minus the amount of heat received from the other plate. The summation evaluates the heat received from the other plate for zero until an infinite amount of reflections (N). For larger values of N , the magnitude of the expression in the summation becomes smaller. In order to solve the equation a certain threshold value is chosen, when the term in the summation becomes smaller than this threshold value the calculation is stopped. The magnitude of the threshold value determines the accuracy of the analytical solution.

The dimensionless heat flux for both diffuse and specular reflection is calculated with equations 4 and 6 respectively. The results are shown in the graph in figure 3 and table 2. The values in the table are accurate up to three digits. The difference between the results for diffuse or specular reflection becomes larger for smaller values of the emissivity.

The total heat loss of one plate is defined by integrating the local heat flux over the width of the plate, as shown in equation 7. The results for the total heat flux are also shown in table 2.

$$Q = \int_0^L q dx \quad (7)$$

This benchmark problem was also calculated numerically with the *Surface-to-Surface Radiation* module in COMSOL, using the *Ray Shooting* algorithm. Every plate is divided in 800 elements, the radiation resolution is equal to 2048, the tolerance is 10^{-6} and the maximum number of adaptations is 6.

The results for the dimensionless local heat flux are shown in the graph in figure 3 alongside the analytical results. In the graph only a few points are shown for the numerical results, in reality the resolution is a lot higher.

The error of the local heat flux between the numerical and analytical results for the settings previously described is below 0.02% for the whole domain. The error of the total heat flux is even smaller. This means that the numerical results are equal to the values shown in table 2.

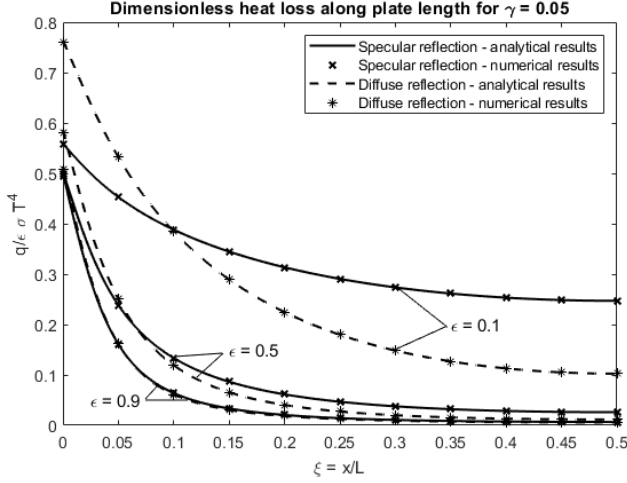


Fig. 3. The dimensionless heat loss distribution along half of the width of one of the plates for the height over width ratio: $\gamma = 0.05$. The heat loss is shown for both diffuse and specular reflection and different values for the emission coefficient. The heat loss increases when the emission coefficient becomes smaller, and is largest close to the sides of the plate. A lower emission coefficient increases the impact of the specular versus diffuse reflection. The numerical results align with the analytical results.

Table 2. Results of the parallel plates problem of the total heat loss (Q) and the local heat loss (q). The heat loss is evaluated for different values of the height over width ratio (γ), the emission coefficient (ϵ) and for both specular as well as diffuse reflection. Both the analytical and numerical calculations result in the values shown in this table.

γ	ϵ	Reflectivity	$\frac{Q}{L\sigma T^4}$	$\frac{q(0)}{\epsilon\sigma T^4}$	$\frac{q(0.25)}{\epsilon\sigma T^4}$	$\frac{q(0.5)}{\epsilon\sigma T^4}$
0.05	0.1	Diffuse	0.0252	0.764	0.180	0.102
		Specular	0.0322	0.559	0.291	0.247
0.05	0.5	Diffuse	0.0413	0.587	0.0273	0.0116
		Specular	0.0463	0.504	0.0468	0.0260
0.05	0.9	Diffuse	0.0476	0.514	0.0124	0.00562
		Specular	0.0485	0.501	0.0142	0.00667
1	0.1	Diffuse	0.0934	0.944	0.933	0.928
		Specular	0.0883	0.890	0.882	0.878
1	0.5	Diffuse	0.369	0.777	0.734	0.718
		Specular	0.350	0.735	0.697	0.682
1	0.9	Diffuse	0.550	0.668	0.604	0.580
		Specular	0.544	0.660	0.598	0.574

2.3 Wedge with Conducting Plates

The third benchmark problem consists of the calculation of the heat loss and temperature distribution along two plates shaped in a wedge, as formulated by Hering [5]. The problem is shown schematically in figure 4. The temperature at the base of the plates is prescribed by the base temperature, T_b . The angle between the plates is equal to ϕ and the plates

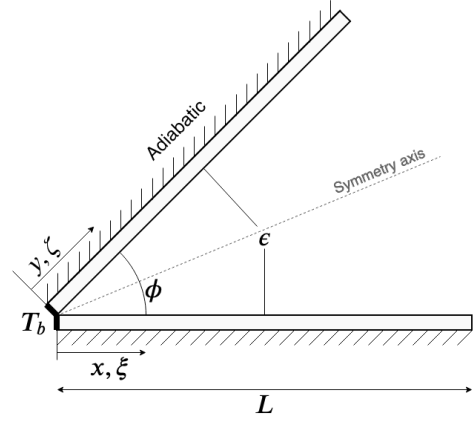


Fig. 4. A schematic representation of the wedge with conducting plates problem. The total as well as the local heat loss from the plates to the surrounding is evaluated. The temperature distribution of both plates is also derived analytically.

have the same length (L) and emission coefficient (ϵ). Just like the isothermal plates problem, this problem is symmetrical.

In order to evaluate the problem in the dimensionless form, the coordinates and local temperature are defined in dimensionless form, as shown in equation 8. To describe the thermal conduction of the plates, the conduction parameter N_c is prescribed. This is shown in equation 8, where κ is the thermal conductivity and t the thickness of the plate. A high value of N_c represents a poorly conducting material, and $N_c = 0$ represents a perfect conductor, resulting in a constant temperature along the plate.

$$\xi = \frac{x}{L}, \quad \zeta = \frac{y}{L}, \quad \theta = \frac{T}{T_b}, \quad N_c = \frac{L^2 \sigma T_b^3}{\kappa t} \quad (8)$$

In equation 9 the boundary conditions of the problem are shown. At $\xi = 0$ the temperature is equal to T_b and the heat loss of the plate ends is assumed to be negligible.

$$\theta(0) = 1, \quad \left. \frac{d\theta}{d\xi} \right|_{\xi=1} = 0 \quad (9)$$

The temperature distribution along one of the plates can be described by equation 10. Because the problem is symmetrical the temperature of the two plates is equal, $\theta(\xi) = \theta(\zeta)$. The variable K_ϕ depends on the emission coefficient, the angle between the plates and the two dimensionless coordinates. The expression for K_ϕ can be found in appendix A. The value of θ can be derived for every value of ξ , by means of iteration.

$$\theta(\xi) = 1 + \epsilon N_c \left(\int_0^\xi \theta^4(\zeta)(\xi - \zeta) d\zeta - \int_0^1 \theta^4(\zeta) K_\phi(\xi, \zeta) d\zeta \right) \quad (10)$$

Table 3. Analytically derived values for the wedge problem of the total heat loss (Q), the local heat loss (q) and the local temperature (T) of the plate. The results are evaluated for different values of the opening angle (ϕ), the emission coefficient (ϵ) and the conduction parameter (N_c). The difference between the numerical results and the results shown in this table is smaller than 1% except for $q(0)$, where the maximum error is slightly larger.

ϕ	ϵ	N_c	$\frac{Q}{L\sigma T_b^4}$	$\frac{q(0)}{\epsilon\sigma T_b^4}$	$\frac{q(0.5)}{\epsilon\sigma T_b^4}$	$\frac{q(1)}{\epsilon\sigma T_b^4}$	$\frac{T(0.5)}{T_b}$	$\frac{T(1)}{T_b}$
60	0.1	0.5	0.0884	0.903	0.878	0.886	0.984	0.978
60	0.1	4	0.0667	0.903	0.632	0.572	0.906	0.877
60	0.9	0.5	0.363	0.303	0.396	0.499	0.929	0.903
60	0.9	4	0.181	0.303	0.167	0.161	0.765	0.700
135	0.1	0.5	0.0933	0.985	0.926	0.908	0.983	0.977
135	0.1	4	0.0699	0.985	0.659	0.578	0.903	0.873
135	0.9	0.5	0.579	0.868	0.609	0.544	0.899	0.868
135	0.9	4	0.272	0.868	0.219	0.147	0.699	0.629

The local heat loss along the plates, q , is a function of the temperature. When the temperature distribution is known the dimensionless heat loss can be derived using equation 11. The expression for G_ϕ is shown in appendix A.

$$\frac{q(\xi)}{\epsilon\sigma T_b^4} = \theta^4(\xi) - \epsilon \int_0^1 \theta^4(\zeta) G_\phi(\xi, \zeta) d\zeta \quad (11)$$

From equations 33 and 34 can be derived that the expression G_ϕ is not well defined for $\xi = \zeta = 0$. To overcome this equation 12 is defined. The expression for L_ϕ is presented in appendix A.

$$\frac{q(0)}{\epsilon\sigma T_b^4} = 1 - \epsilon L_\phi(\epsilon) \quad (12)$$

Equation 13 describes how the total heat loss of one plate, Q , can be calculated. The expression of M_ϕ is provided in appendix A.

$$\frac{Q}{L\sigma T_b^4} = \epsilon \int_0^1 \theta^4(\zeta) M_\phi(\zeta) d\zeta \quad (13)$$

Analytical results for the local heat loss and temperature as well as the total heat loss are provided in table 3. Figure 5 presents the heat loss and temperature distributions for various values of the emission coefficient and the conduction parameter.

This benchmark problem was also calculated numerically with the *Surface-to-Surface Radiation* module in COMSOL, using the *Ray Shooting* algorithm. Each plate is divided into 400 elements, the radiation resolution is set to 512, a tolerance of 10^{-5} is chosen and the maximum number of adaptations is equal to 5.

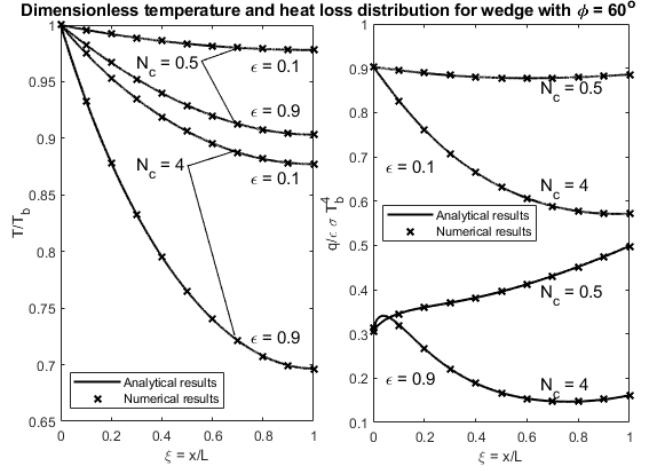


Fig. 5. The dimensionless temperature distribution (left) and the heat loss distribution (right) of the wedge with an opening angle $\phi = 60^\circ$. Both analytical and numerical results are shown for different values of the emission coefficient and conduction parameter. A low value of the conduction parameter and emission coefficient results in a more uniform temperature of the plate. The heat loss becomes larger for a higher temperature and decreases for a higher view factor, which is larger close to the base.

The resulting heat loss and temperature distribution are shown in the graphs of figure 5 alongside the analytical results. The real resolution of the numerical results is much higher than shown in the figure.

The evaluation of the element in the corner of the wedge ($\xi = 0$) is quite sensitive in the numerical results, which results in a slightly larger error for the local heat flux. For the rest of the domain the error in local heat flux is always below 1%. The error in the local temperature stays below 0.25% for the whole domain.

3 Angle Dependent Reflecting Surfaces and the Effect of Polarization

The influence of specular reflection with respect to diffuse reflection is described in the previous section. The magnitude of the reflection coefficient is considered constant. However research shows that the reflection coefficient depends on the angle of incidence and the polarization state of the irradiation [6, 11, 12].

In section 3.1 the Fresnel equations are treated, which describe the angle dependent reflection of surfaces and in what way this affects the polarization state of the radiation. Section 3.2 describes an analytical calculation of the transmittance of thermal radiation through a square passage that can be used as a benchmark problem.

A numerical model of this benchmark is created in COMSOL using the *Ray Optics* module. The equations described in section 3.1 are used to model the surfaces. The influence of including the angle dependency and polarization in the models is discussed in section 3.3.

3.1 Fresnel Equations

Equations 14 and 15 show the Fresnel equations, which are derived from Electromagnetic Theory and describe the reflection coefficients [1, 13].

$$r_s = \frac{\cos(\theta_i) - \tilde{n} \cos(\theta_t)}{\cos(\theta_i) + \tilde{n} \cos(\theta_t)} \quad (14)$$

$$r_p = \frac{\tilde{n} \cos(\theta_i) - \cos(\theta_t)}{\tilde{n} \cos(\theta_i) + \cos(\theta_t)} \quad (15)$$

In the above equations θ_i is the angle of incidence, θ_t the angle of transmission and \tilde{n} the complex refractive index. The angle of transmission can be derived by equation 16 and is a function of the complex refractive index and the angle of incidence. The complex refractive index consists of a real and an imaginary part, which are described by the material's refractive index (n) and absorptive index (k) respectively, as shown in 17.

$$\theta_t = \arcsin\left(\frac{1}{\tilde{n} \sin(\theta_i)}\right) \quad (16)$$

$$\tilde{n} = n - ik \quad (17)$$

The fraction of energy that is reflected by a surface is described by the power reflection coefficient R , derived by equation 18.

$$R_s = |r_s|^2 \quad R_p = |r_p|^2 \quad (18)$$

The perpendicular (R_s) and parallel (R_p) components of the reflection coefficient for varying angle of incidence are shown in figure 6 for two different materials. For dielectrics applies that $k = 0$ and for metals applies $k = n$, since it is assumed that the Hagen-Ruebens relation holds [1, 14].

When considering unpolarized radiation the surface reflection coefficient is the average of the perpendicular and the parallel component, as shown in equation 19.

$$R = \frac{R_s + R_p}{2} \quad (19)$$

From figure 6 it can be seen that there is a significant difference between the reflection of perpendicular and parallel polarized radiation, especially for larger angles of incidence. The amount of reflection is thus not only dependent on the angle of incidence but also depends on the state of polarization of the irradiation.

Due to the difference in reflectivity for perpendicular and parallel polarized radiation, unpolarized radiation will become partly polarized after a reflection. Radiation that is reflected of a dielectric surface can become fully polarized when the angle of incidence is equal to the Brewster's angle.

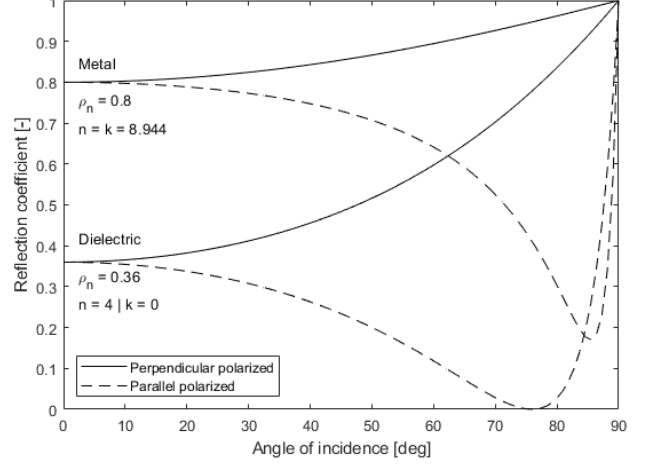


Fig. 6. The angle dependent reflection coefficient for the parallel and perpendicular polarized radiation. The distribution for two different materials are shown: a dielectric with $\rho_n = 0.36$ and a metal with $\rho_n = 0.8$. The reflection coefficient for the perpendicular polarized radiation is larger than for parallel polarized radiation for all angles of incidence.

3.2 Square Passage Benchmark

Edwards and Tobin [6] derived an analytical approximation to derive the transmittance of radiation through a square passage, including the effect of angle dependent reflection and polarization. In section 3.2.1 the analytical approximation is described. The results of this model are discussed in section 3.2.2. A numerical model has been developed and the analytical and numerical results are compared in section 3.2.3.

3.2.1 Analytical Approximation Model

Figure 7 shows a schematic representation of the problem. The width and height of the square passage is indicated by D and the length of the passage by L . The walls of the passage are considered opaque and optically smooth. The passage contains a nonabsorbing, nonscattering medium. The square at the front end is a black body that emits unpolarized radiation in a Lambertian distribution. The amount of radiation emitted at the front of the passage that reaches the end of the tube, the transmittance (τ), is derived by using equations 20 until 27.

The amount and state of radiation is given by the intensity vector, $[I]$. In this vector, I_p , represents the fraction of parallel polarized radiation, I_s is the fraction of perpendicular polarized radiation and I_u and I_v represent the third and fourth Stokes coefficients. If the radiation undergoes a reflection the next intensity vector can be determined by multiplying the intensity vector with the reflection matrix, $[\rho]$, as shown

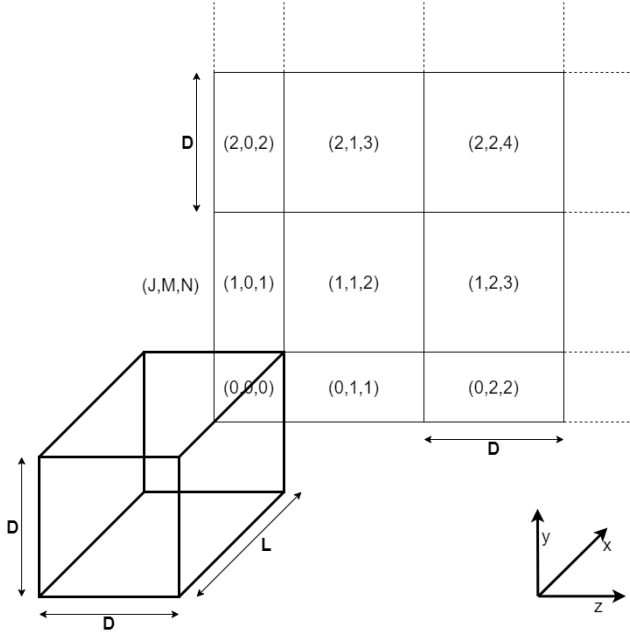


Fig. 7. Schematic representation of the square duct benchmark including the mirror images of the end planes. A radiation source is situated at the front of the duct and the percentage of radiation that reaches the end of the passage, the transmission τ , is evaluated. The variable J represents the amount of reflections with the top and bottom wall, M the amount of reflections with the side walls and N represents the total amount of reflections.

in equation 20.

$$[I_2] = \begin{bmatrix} I_{p2} \\ I_{s2} \\ I_{u2} \\ I_{v2} \end{bmatrix} = \begin{bmatrix} \rho_{pp} & 0 & 0 & 0 \\ 0 & \rho_{ss} & 0 & 0 \\ 0 & 0 & \rho_{uu} & \rho_{uv} \\ 0 & 0 & \rho_{vu} & \rho_{vv} \end{bmatrix} \cdot \begin{bmatrix} I_{p1} \\ I_{s1} \\ I_{u1} \\ I_{v1} \end{bmatrix} = [\rho] \cdot [I_1] \quad (20)$$

The elements of the reflection matrix depend on the angle of incidence, the wall's refractive index and absorption coefficient. The expressions for the matrix elements are given in appendix B.

The polarization state of the irradiation is determined with respect to the surface it coincides. If the surfaces are parallel to each other the orientation of the polarization planes does not change. However when the next surface is at a different angle than the first surface, the orientation of the polarization planes changes. In the square duct this happens when the radiation first reflects with the bottom or top surface and then by with one of the sides, or the other way around. To take this into account the intensity vector, I_2 , is multiplied by a rotation matrix, D_{1-2} , as shown in equation 21. The rotation matrix is defined in equation 22.

$$[I_3] = [\rho_2] \cdot [D_{1-2}] \cdot [I_2] = [\rho_2] \cdot [D_{1-2}] \cdot [\rho_1] \cdot [I_1] \quad (21)$$

$$[D_{1-2}] = \begin{bmatrix} \cos^2(\beta) & \sin^2(\beta) & \frac{1}{2} \sin(2\beta) & 0 \\ \sin^2(\beta) & \cos^2(\beta) & -\frac{1}{2} \sin(2\beta) & 0 \\ -\sin(2\beta) & \sin(2\beta) & \cos(2\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

where the rotation of polarization planes (β) is defined as:

$$\begin{aligned} \mathbf{p}_1 \cdot \mathbf{p}_2 &= \cos(\beta) \\ \mathbf{p}_1 \cdot \mathbf{s}_1 &= \sin(\beta) \end{aligned} \quad (23)$$

The mirror-image technique of Sparrow and Eckert [15] is used to calculate the transmittance. In figure 7 the mirror images are shown for one fourth the end surface. In this figure, J corresponds with the number of reflections with the top and bottom walls, M with the two side walls and N is the total number of reflections. The front surface is subdivided into smaller elements, dA_1 . The mirror images on the end plane are also subdivided into smaller elements, dA_2 . For each pair of elements the number of reflections is determined and for each reflection the reflection and rotation matrix are calculated.

The percentage of transmission through the tube is calculated, so the initial energy entering the tube is equal to 100%. Since the initial radiation is unpolarized, half of the radiation is parallel polarized and the other half is perpendicularly polarized. The energy matrix and initial intensity matrix can be defined as:

$$\begin{aligned} [E] &= \left[\frac{dA_1}{A_1} \cdot 100 \quad \frac{dA_1}{A_1} \cdot 100 \quad 0 \quad 0 \right] \\ [I_1] &= \begin{bmatrix} 1/2 \\ 1/2 \\ 0 \\ 0 \end{bmatrix} \end{aligned} \quad (24)$$

The equation for the power transfer between elements is:

$$\begin{aligned} dQ_{dA_1-dA_2} &= \int_0^\infty [E] \cdot [\rho_N] \cdot [D_{(N-1)-N}] \cdots \\ &\cdots [D_{1-2}] \cdot [\rho_1] \cdot [I_1] \text{Ker}_{dA_1-dA_2} dA_1 dA_2 d\lambda \end{aligned} \quad (25)$$

The complex refractive index of the wall material is dependent of the wavelength (λ) of the incident radiation. The complex refractive index is a chosen constant, thus the wavelength dependency is not taken into account in this calculation. The radiation transfer kernel, Ker , describes the view factor between two elements. The expression for the kernel is shown in equation 26.

$$\text{Ker}_{dA_1-dA_2} = \frac{\cos(\theta_{dA_1}) \cos(\theta_{dA_2})}{\pi r_{dA_1-dA_2}^2} \quad (26)$$

In this equation θ_{dA_1} and θ_{dA_2} are the angle of incidence at the front and the end of the passage. Since these faces are

Table 4. This table shows the derived values for the percentage of transmission through the square passage for three dielectrics and two metals with different values for the reflection at normal incidence. For every material three values for the transmission are shown: the value derived by Edwards and Tobin [6] ($\tau_{[6]}$), the converged analytical value (τ_a) and numerical results derived using the *Ray Optics* module in COMSOL (τ_{num}).

Dielectric: $k = 0$

L/D	$\rho_n = 0.04$			$\rho_n = 0.1111$			$\rho_n = 0.36$		
	$\tau_{[6]}$	τ_a	τ_{num}	$\tau_{[6]}$	τ_a	τ_{num}	$\tau_{[6]}$	τ_a	τ_{num}
0	100	100	100	100	100	100	100	100	100
0.3	59.8	61.1	60.8	61.9	63.4	63.2	69.9	70.7	70.5
0.5	44.4	45.6	45.5	46.3	48.6	48.2	55.9	57.2	57.2
1	21.1	25.1	24.6	23.1	27.6	27.2	31.0	35.9	35.8
3	5.69	7.11	6.91	6.54	8.00	7.82	8.96	10.1	10.1
5	3.47	4.05	3.94	3.96	4.57	4.47	4.52	5.18	5.15
10	1.81	1.95	1.94	2.08	2.23	2.23	2.25	2.44	2.47
30	0.616	0.632	0.640	0.714	0.730	0.740	0.789	0.809	0.822

Metal: $k = n$

L/D	$\rho_n = 0.5$			$\rho_n = 0.8$		
	$\tau_{[6]}$	τ_a	τ_{num}	$\tau_{[6]}$	τ_a	τ_{num}
0	100	100	100	100	100	100
0.3	75.43	75.8	75.7	88.5	87.8	87.7
0.5	63.52	64.1	64.1	81.3	80.8	80.8
1	42.53	44.1	44.1	63.9	66.6	67.0
3	14.52	15.1	15.21	34.8	34.5	35.3
5	7.88	8.04	8.05	20.1	20.0	20.5
10	3.54	3.65	3.69	6.65	6.86	7.15
30	1.13	1.19	1.21	1.20	1.23	1.27

parallel to each other, these angles are always equal. The variable $r_{dA_1-dA_2}$ describes the distance between dA_1 and dA_2 .

The total transmittance through the passage can now be calculated by summing the power transfer between elements over all the elements at the front and end of the passage. Due to the symmetry in the model only one fourth of the end plane needs to be calculated and the result is then multiplied by four, as shown in equation 27.

$$\tau = 4 \cdot \sum_{A_1} \sum_{A_2} dQ_{dA_1-dA_2} \quad (27)$$

3.2.2 Analytical Model Results

The calculation described in this section is an analytical approximation of the problem. The accuracy of the result depends on the amount of reflections (N) taken into account and the amount of subdivisions of the end planes 1 and 2 (dA_1, dA_2).

The results of the model calculated by Eckert and Tobin [6] include at least the first 25 reflections. The areas 1 and 2

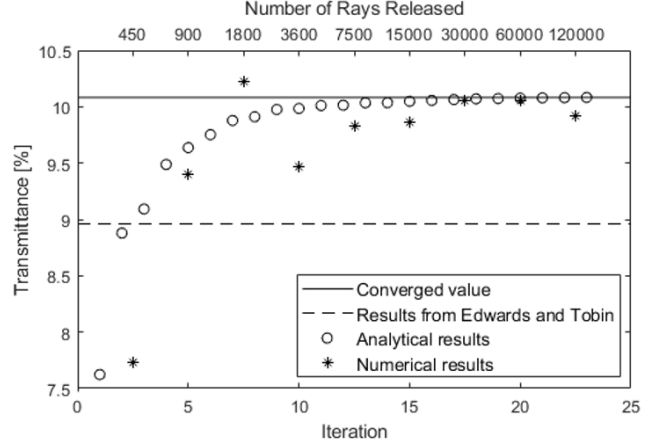


Fig. 8. The results for the analytically and numerically calculated transmittance for $\rho_n = 0.36$ and $L/D = 3$. For each iteration of the analytical calculation either the amount of reflections or discretizations is increased. The number of released rays is increased to achieve more accurate numerical results. The analytical results show good convergence, which is higher than the value described in the paper of Edwards and Tobin. The numerical results show a good approximation of the result, however are not fully converged.

were subdivided in 3×3 for the first 5 reflections. The areas were not subdivided for calculations with more than 5 reflections.

In order to obtain more accurate results a function is created that gradually increases the amount of reflections and subdivisions taken into account. Another function is defined that ensures that the number of subdivisions decreases gradually with the amount of reflections. By increasing both number of reflections and subdivisions the solution converges to a final value. When the transmittance is accurate for three digits, the calculation is stopped. Figure 8 shows how the calculation converges to a result.

Table 4 shows the analytically derived percentage of transmittance (τ_a) for different length over width ratios of the square passage and five different materials.

The results are shown alongside the results from Edwards and Tobin [6] ($\tau_{[6]}$). There is a significant difference between the values especially for length to width ratios of 1,3 and 5. For these values the largest amount of subdivisions was required to achieve convergence.

3.2.3 Ray Tracing Model

The square passage is also modeled in the software package COMSOL using the *Ray Optics* module [10]. A ray tracing method is used to describe the heat transfer. In the simulation 60,000 rays are released at the front end in a Lambertian distribution. When a ray is reflected from one of the side walls, a fraction of its energy is transferred to this wall. The transmittance is derived by evaluating the percentage of energy is transferred to the walls and subtracting this

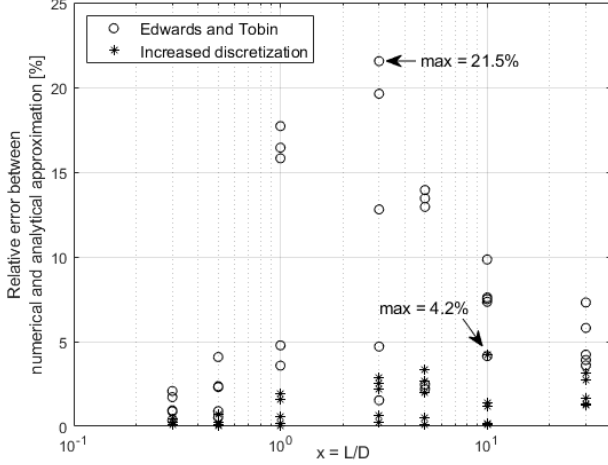


Fig. 9. The relative error between the numerical results and the results from the analytical approximation. The error for the values derived by Edwards and Tobin are shown as well as the error for the converged values. The maximum error between the numerical and converged analytical results is equal to 4.2%. The difference between the converged results and the results from Edwards and Tobin [6] can be clearly seen from this graph.

value from the initial energy at the beginning of the passage. The reflection coefficient of the passage walls is defined as described in equations 14 until 18.

The numerical results are shown alongside the analytical results in table 4. The error between the results from the simulation and the analytical model are shown in figure 9. The results from the simulation correspond well with the analytical calculation. The maximum error is equal to 4.2% for $L/D = 10$ and $\rho_n = 0.8$.

The error between the numerical results and the results derived by Edwards and Tobin [6] are also shown in figure 9. The improvement in the analytical results can be seen clearly, since the relative error is consistently larger for the results from Edwards and Tobin.

3.3 Effect of Polarization and Angle Dependent Reflection on the Transmittance

Including polarization and angle dependency to heat transfer models significantly increases the complexity of the calculation. It is thus interesting to know what their influence is on the transmittance. Figure 10 shows the transmission for varying length to width ratio of the passage for a dielectric and a metal. Three cases are considered: 1. including both angle dependency and polarization, 2. including angle dependency only and 3. considering a constant reflection coefficient.

It can be concluded that the influence of polarization increases for higher length over width ratio. The influence of the angle dependent reflection on the transmission increases for higher values of ρ_n .

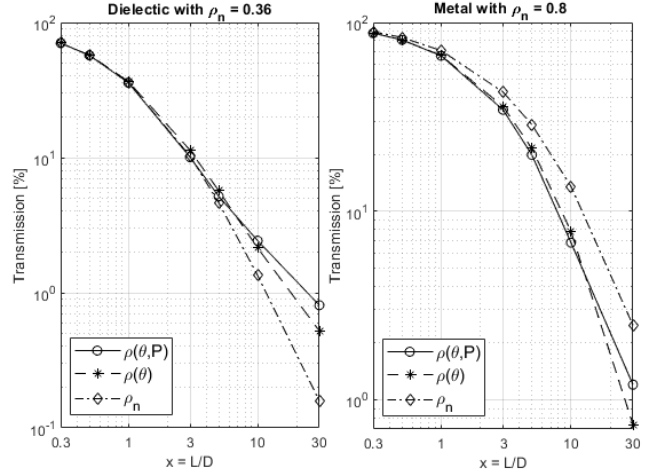


Fig. 10. The percentage of transmission through the square duct as a function of the duct length to width ratio for a dielectric and a metal wall. The different lines indicate which physics are included in the calculation; for $\rho(\theta, P)$ both angle dependency and polarization are included, for $\rho(\theta)$ only angle dependency is included and for ρ_n the normal reflectivity is used. The effect of polarization increases for a higher L/D ratio.

4 Experimental Validation for Models including Angle Dependent Reflection and Polarization

Miranda [7] derived an analytical calculation for the transmission of infrared radiation through a cylindrical metal tube with specular internal surfaces. An experiment was conducted to validate the analytical calculation. By considering the reflection coefficients for the parallel and perpendicular polarization states, ρ_{\perp} and ρ_{\parallel} , the analytical model takes the effect of polarization into account. However this analytical model does not account for the rotation of the polarization planes, described by the rotation matrix $[D]$ in section 3. After multiple reflections the radiation will become fully perpendicular polarized, despite the fact that the orientation of the polarization planes rotates. This will result in an overestimation of the transmission that increases for longer tube lengths [6, 11, 16]. This can also be seen in the results from the model, which are presented in section 4.3.

Despite the fact that the analytical model is not an accurate representation of the transmission the results from the experiment can be compared to the verified model from COMSOL described in section 3.

In section 4.1 a description of the conducted experiment by Miranda is given. The experiment results are compared with results from a numerical model, which is discussed in section 4.2.

4.1 Description of the Experiment

The experiment consists of a stainless steel tube with a radius of $R = 3.35\text{cm}$, which is sealed on both ends to create a vacuum inside. The tube is submerged into liquid nitrogen to keep the side walls at a temperature of 77K . At both ends of the tubes a concentric copper disk with a radius of

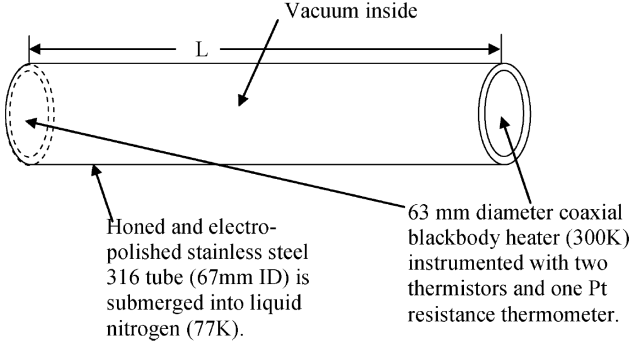


Fig. 11. Schematic representation of the experiment conducted by Miranda (1996) [7]. Figure from Qu et al. (2007) [14]

$r_0 = 3.15\text{cm}$ is situated, which are kept at a uniform temperature of 300K . Figure 11 shows a schematic representation of the experiment.

The transmission of the thermal radiation through the tube is calculated by equation 28. The amount of electrical power (P) supplied to the heaters required to keep both disks at a uniform temperature is measured as a function of the distance, L , between the disks. The amount of power that needs to be supplied to the heaters when the distance between the disks is zero is described by P_0 . When the disks are so far apart that there is no heat transfer between the disks, a maximum amount of power must be supplied, P_{max} . The emission coefficient of the disks, ϵ , is determined experimentally by calculating the ratio between the measured emitted power and the theoretical black body emission.

$$\tau'_{exp}(L) = \frac{P_{max} - P(L)}{\gamma(P_{max} - P_0 - (1 - \epsilon)(P(L) - P_0))} \quad (28)$$

The factor γ in equation 28 accounts for the difference between the transmission through the whole tube and the transmission received by the copper disk. This factor is introduced by Miranda to match the experiment results with the results from the analytical model, which can not take the smaller radius of the disk into account. This correction factor is calculated by equation 29.

$$\gamma = \frac{F}{\tau_{[6]}} + \left(1 - \frac{F}{\tau_{[6]}}\right) \left(\frac{r_0}{R}\right)^2 \quad (29)$$

In equation 29, F is the view factor for the radiation leaving one disk and that reaches the other disk directly. For the values of $\tau_{[6]}$ the estimates from the analytical model from Miranda were taken. As mentioned before these analytical results overestimate the transmission for high length over width ratios.

In table 5 the experimental data as displayed by Miranda [7], and calculated using equation 28 is shown in the column τ'_{exp} . The values of γ are shown in the next column. The radiation heat transfer measured from disk to disk, τ_{exp} , is obtained by

Table 5. The experimental results alongside the numerical results for the transmission of thermal radiation through a cylinder for different length to diameter ratios of the tube ($D = 6.7\text{cm}$). The values for the transmission through the tube as given by Miranda [7] are shown in the column τ'_{exp} . The correction factor γ is used to convert the results from Miranda in such a way that the analytical model is not included in the experiment data. The radiation heat transfer measured from disk to disk, τ_{exp} , is obtained by multiplying the τ'_{exp} with γ . This is also calculated numerically and the numerical results, τ_{num} , are shown in the last column.

L/D	τ'_{exp}	γ	τ_{exp}	τ_{num}
0.0597 ± 0.015	99.2 ± 0.8	0.987	97.9 ± 0.8	93.2
0.157 ± 0.022	98.2 ± 0.8	0.969	95.1 ± 0.8	88.9
1.46 ± 0.03	90.0 ± 0.8	0.896	80.7 ± 0.8	75.8
3.79 ± 0.03	71.2 ± 0.9	0.887	63.1 ± 0.9	60.0
7.51 ± 0.03	51.5 ± 1.1	0.885	45.6 ± 1.1	43.5
11.91 ± 0.03	37.2 ± 1.1	0.885	32.9 ± 1.1	31.4
17.13 ± 0.03	26.8 ± 1.3	0.885	23.7 ± 1.3	22.5

multiplying the τ'_{exp} with γ . This is also calculated numerically and the numerical results, τ_{num} , are shown in the last column.

4.2 Numerical Model

This problem is also calculated numerically in COMSOL using the *Ray Optics* module [10]. The model uses the same ray tracing method as described in section 3.2.3. The geometry from figure 11 is replicated, where one disk is modeled as the source of radiation and the other disk is the receiver. At the source 100,000 rays are released and the ray directions are distributed in a Lambertian distribution. In contrast to the model from section 3.2.3 this model does include wavelength dependency of the reflection. The reflection coefficient of the side wall of the cylindrical tube is still calculated by equations 14 until 18. However the values for the extinction coefficient, k , and the refraction index, n , depend on the wavelength of the irradiation, described by the Maxwell's equations. Since the electrical resistivity, r_e , of metal is small and the wavelength of thermal radiation is quite long, $\lambda_0 > \sim 5\mu\text{m}$, the Hagen-Rubens equation can be used in the numerical model.

$$k = n = \sqrt{\frac{\lambda_0 \mu_0 c_0}{4\pi r_e}} = \sqrt{\frac{0.003\lambda_0}{r_e}} \quad (30)$$

The electrical resistivity of the 316 stainless steel at 77K is $r_e = 5 \cdot 10^{-5}\Omega\text{cm}$. The speed of light and magnetic permeability in vacuum, c_0 and μ_0 , are physical constants. The wavelength of the radiation, λ_0 , is assigned to each ray at the release. In order to assign the wavelengths an inverse probability function that describes the emission of gray or black

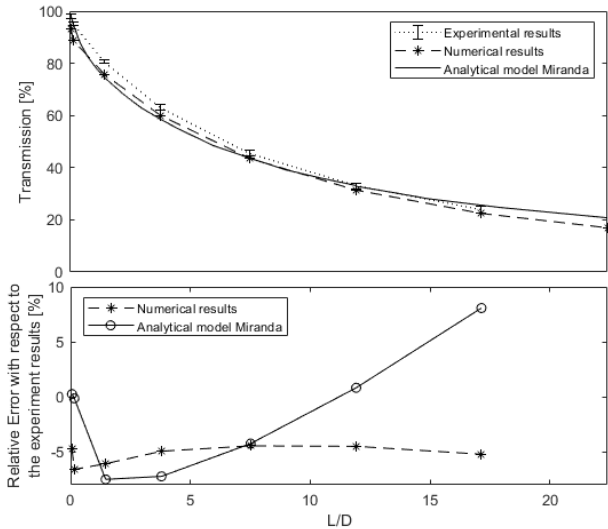


Fig. 12. The upper graph shows percentage of transmission through a tube as function of the length to diameter ratio. The experimental results are shown alongside the results from the numerical model and the analytical model from Miranda [7]. The bottom graph shows the relative error from the numerical model and the analytical model by Miranda with respect to the experiment results.

bodies is used [1]. The wavelength is distributed conform the Planck's wavelength distribution.

4.3 Model Results and Comparison with Experiment

Table 5 and figure 12 show the results of the numerical model alongside the experimental data. The heat transfer calculated by the numerical model is between 4 and 7% lower than the measured transmission from the experiment for all measured lengths. The numerical model seems to replicate the trend of the results very well.

In the paper by Miranda [7] various possible causes for errors are mentioned. According to Miranda the error is dominated by the uncertainties associated with P_0 and the average disk temperature. An error in either one of these parameters forms a systematic error, which means that the effect will not change for different tube lengths. This means that the offset between the experiment results and the numerical model results could be caused by these uncertainties.

In figure 12 the radiative heat transfer from disk to disk calculated by the model described by Miranda [7] is also added. The trend for this model is clearly different from the experiment data. Since the effect of not taking the rotation of the polarization planes into account is stronger for longer tube lengths it is expected that the error between experiment and model will increase for higher length to width ratios.

5 Discussion

As described in section 3, the results from the numerical model constructed with the *Ray Optics* module in COMSOL show good agreement with the analytical results, as can be

seen in table 4 and figure 9. However the error of the numerical model increases when the reflection coefficient and the length over width ratio increase. Problems including highly reflecting surfaces with many reflections will require more computational expensive models to reach the accuracy displayed in figure 9.

From the results of the experimental validation described in section 4 is concluded that the trend of the transmission as a function of the length to diameter ratio is modeled accurately by the numerical model. However it is not known with confidence what causes the offset of roughly 5% between the experimental and numerical results. The offset can be caused by numerical errors in the computation. It can also be caused by the extrapolation conducted by Miranda to determine the value of P_0 .

6 Conclusion

This paper describes benchmarks which can be used to verify computational tools for modeling thermal radiation, in which the direction and polarization state of the irradiation is of importance.

The benchmarks in section 2 describe three problems with increasing complexity including specular reflecting surfaces. The function to model specular reflection with the *Surface-to-Surface Radiation* module from COMSOL, was verified by comparing the numerical and the analytical results. The numerical results will converge to the analytical solution. For the settings described in this paper the error is smaller than 1.5% for all results shown in section 2.

Section 3 describes a more complex problem including the effect of angle dependent reflection and the polarization state of the irradiation. The accuracy of the results from the analytical approximation, with respect to the results from the paper of Edwards and Tobin [6], has been increased. The *Ray Optics* module of COMSOL was used to model the problem described in section 3. The error between the numerical and analytical results is below 5% for all evaluated materials and length ratios. This is a large improvement when comparing the model to the results from Edwards and Tobin, which resulted in a maximum error of more than 20%.

The impact of including of the angle dependent reflection and polarization in the radiation transfer through a passage is shown in section 3.3. Excluding polarization from the calculation can lead to an error of more than a factor 2 for large aspect ratios.

The ray tracing model verified in section 3 was used to model the experiment conducted by Miranda [7]. The numerical results are approximately 5% lower than the experimental results for all aspect ratios. The developed model shows better agreement with the experiment than the results from Miranda [7], for which the error varies between -7% and +8%. This is due to the fact that the influence of the rotation of polarization planes is not taken into account by Miranda.

The 5% offset of the numerical model could be caused by the uncertainty in the extrapolation to calculate P_0 .

References

- [1] Howell, J. R., Meng, M. P., and Siegel, R., 2016. *Thermal radiation heat transfer (6th edition)*. CRC Press.
- [2] Sparrow, E. M., 1960. "Application of variational methods to radiation heat-transfer calculations". *Journal of Heat Transfer* 82(4):375-380.
- [3] Sparrow, E. M., Gregg, J. L., Szel, J. V., and Manos, P., 1961. "Analysis, results, and interpretation for radiation between some simply-arranged gray surfaces". *Journal of Heat Transfer* 83(2):207-214.
- [4] Eckert, E. R. G., and Sparrow, E. M., 1961. "Radiative heat exchange between surfaces with specular reflection". *International Journal of Heat and Mass Transfer* 3(1):42-54.
- [5] Hering, R. G., 1966. "Radiative heat exchange between conducting plates with specular reflection". *Journal of Heat Transfer* 88(1):29-36.
- [6] Edwards, D. K., and Tobin, R. D., 1967. "Effect of polarization on radiant heat transfer through long passages". *Journal of Heat Transfer* 89(2):132-138.
- [7] Miranda, P. C., 1996. "A calculation and experimental verification of the infrared transmission coefficient of straight cylindrical metal tubes". *Journal of Heat Transfer* 118(2):495-497.
- [8] Birkebak, R. C., and Eckert, E. R. G., 1965. "Effects of roughness of metal surfaces on angular distribution of monochromatic reflected radiation". *Journal of Heat Transfer* 87(1):85-93.
- [9] Leen, L., Severins, I., Peeters, J., and Steenackers, G., 2018. "Diffuse versus specular reflection: the influence of hot spots on reflected apparent temperature". In QIRT 2018 Proceedings: 412-417.
- [10] *COMSOL Multiphysics Reference Manual, version 5.4*. COMSOL, Inc, www.comsol.com.
- [11] Edwards, D. K., and Bevans, J. T., 1965. "Effect of polarization on spacecraft radiation heat transfer". *AIAA Journal* 3(7):1323-1329.
- [12] Oppenheim, U. P., and Feiner, Y., 1995. "Polarization of the reflectivity of paints and other rough surfaces in the infrared". *Applied Optics* 34(10):1664-1671.
- [13] Modest, M. F., 2013. *Radiative heat transfer (3rd edition)*. Elsevier.
- [14] Qu, Y., Howell, J. R., and Ezekoye, O. A., 2007. "Monte carlo modeling of a light-pipe radiation thermometer". *IEEE Transactions on Semiconductor Manufacturing* 20(1):39-49.
- [15] Sparrow, E. M., Eckert, E. R. G., and Jonsson, V. K., 1962. "An enclosure theory for radiative exchange between specularly and diffusely reflecting surfaces". *Journal of Heat Transfer* 84(4):294-299.
- [16] Wallace, A. M., Liang, B., Trucco, E., and Clark, J., 1999. "Improving depth image acquisition using polarized light". *International Journal of Computer Vision* 32(2):87-109.

A Equations wedge model

This section shows the equations for the variables required to evaluate the analytical calculation of the wedge benchmark described in section 2.3. The equations were derived by Hering [5].

$$K_{\phi}(\xi, \zeta) = \begin{cases} \xi + \varepsilon[J_{\phi}(\xi, \zeta)], & 90^{\circ} \leq \phi < 180^{\circ} \\ \xi + \varepsilon[J_{\phi}(\xi, \zeta) + (1 - \varepsilon)J_{2\phi}(\xi, \zeta)], & 60^{\circ} \leq \phi < 90^{\circ} \\ \xi + \varepsilon[J_{\phi}(\xi, \zeta) + (1 - \varepsilon)J_{2\phi}(\xi, \zeta) \\ \quad + (1 - \varepsilon)^2 J_{3\phi}(\xi, \zeta)], & 45^{\circ} \leq \phi < 60^{\circ} \end{cases} \quad (31)$$

$$J_{\phi}(\xi, \zeta) = \frac{1}{2} \left[\frac{\xi(\zeta - \cos(\phi))}{\sqrt{1 - 2\phi \cos(\phi) + \zeta^2}} + \cos(\phi) \left(\sqrt{\xi^2 - 2\xi\zeta \cos(\phi) + \zeta^2} - \zeta \right) - \zeta \sin^2(\phi) \left[\ln \left(\xi - \zeta \cos(\phi) + \sqrt{\xi^2 - 2\xi\zeta \cos(\phi) + \zeta^2} \right) - \ln(\zeta(1 - \cos(\phi))) \right] \right] \quad (32)$$

$$G_{\phi}(\xi, \zeta) = \begin{cases} f_{\phi}(\xi, \zeta), & 90^{\circ} \leq \phi < 180^{\circ} \\ f_{\phi}(\xi, \zeta) + (1 - \varepsilon)f_{2\phi}(\xi, \zeta), & 60^{\circ} \leq \phi < 90^{\circ} \\ f_{\phi}(\xi, \zeta) + (1 - \varepsilon)f_{2\phi}(\xi, \zeta) \\ \quad + (1 - \varepsilon)^2 f_{3\phi}(\xi, \zeta), & 45^{\circ} \leq \phi < 60^{\circ} \end{cases} \quad (33)$$

$$f_{\phi}(\xi, \zeta) = \frac{\sin^2(\phi)}{2} \frac{\xi\zeta}{(\xi^2 + \zeta^2 - 2\xi\zeta \cos(\phi))^{3/2}} \quad (34)$$

$$L_{\phi}(\varepsilon) = \begin{cases} \frac{1}{2}(1 + \cos(\phi)), & 90^{\circ} \leq \phi < 180^{\circ} \\ \frac{1}{2}[(1 + \cos(\phi)) \\ \quad + (1 - \varepsilon)(1 + \cos(2\phi))], & 60^{\circ} \leq \phi < 90^{\circ} \\ \frac{1}{2}[(1 + \cos(\phi)) + (1 - \varepsilon)(1 + \cos(2\phi)) \\ \quad + (1 - \varepsilon)^2(1 + \cos(3\phi))], & 45^{\circ} \leq \phi < 60^{\circ} \end{cases} \quad (35)$$

$$M_{\phi}(\zeta) = \begin{cases} 1 - \varepsilon P_{\phi}(\zeta), & 90^{\circ} \leq \phi < 180^{\circ} \\ 1 - \varepsilon[P_{\phi}(\zeta) + (1 - \varepsilon)P_{2\phi}(\zeta)], & 60^{\circ} \leq \phi < 90^{\circ} \\ 1 - \varepsilon[P_{\phi}(\zeta) + (1 - \varepsilon)P_{2\phi}(\zeta) \\ \quad + (1 - \varepsilon)^2 P_{3\phi}(\zeta)], & 45^{\circ} \leq \phi < 60^{\circ} \end{cases} \quad (36)$$

$$P_{\phi}(\zeta) = \frac{1}{2} \left[1 + \frac{\cos(\phi) - \zeta}{(1 + \zeta^2 - 2\xi\zeta \cos(\phi))^{1/2}} \right] \quad (37)$$

B Reflection Matrix equations

This section describes the expressions for the elements of the reflection matrix for the analytical calculation of the square passage benchmark described in section 3.2.1, and first formulated by Edwards and Tobin [6].

$$\rho_{ss} = \frac{(\cos(\theta_i) - a)^2 + b^2}{(\cos(\theta_i) + a)^2 + b^2} \quad (38)$$

$$\rho_{pp} = \frac{[(n^2 - k^2) \cos(\theta_i) - a]^2 + [2nk \cos(\theta_i) - b]^2}{[(n^2 - k^2) \cos(\theta_i) + a]^2 + [2nk \cos(\theta_i) + b]^2} \quad (39)$$

$$\rho_{uu} = \rho_{vv} = \sqrt{\rho_{ss}\rho_{pp}} \cos(\delta) \quad (40)$$

$$\rho_{vu} = -\rho_{uv} = \sqrt{\rho_{ss}\rho_{pp}} \sin(\delta) \quad (41)$$

$$\delta = 2\pi + (\delta_p - \delta_s) \quad (42)$$

$$\tan(\delta_p) = \frac{2 \cos(\theta_i) [(n^2 - k^2)b - 2nk a]}{(n^2 + k^2)^2 \cos^2(\theta_i) - (a^2 + b^2)} \quad (43)$$

$$\tan(\delta_s) = \frac{2b \cos(\theta_i)}{\cos^2(\theta_i) - a^2 - b^2} \quad (44)$$

$$a^2 = \frac{1}{2} \left[\sqrt{(n^2 - k^2 - \sin^2(\theta_i))^2 + 4n^2k^2} + (n^2 - k^2) - \sin^2(\theta_i) \right] \quad (45)$$

$$b^2 = \frac{1}{2} \left[\sqrt{(n^2 - k^2 - \sin^2(\theta_i))^2 + 4n^2k^2} - (n^2 - k^2) + \sin^2(\theta_i) \right] \quad (46)$$

A Specular Reflection Benchmarks

A.1 Introduction

Appendix A contains details on the benchmarks on specular reflection, described in section 2 of the paper. Three benchmarks have been derived for the verification of models including specular reflection.

- Concentric Cylinders - section A.2
- Isothermal Parallel Plates - section A.3
- Wedge with Conducting Plates - section A.4.2

From each of the benchmarks an analytical and a numerical model has been created. In this appendix the MatLab models for all analytical calculations are shown, as well as the numerical models developed in COMSOL.

The numerical model has been compared to the analytical model to assess the quality of these models. The results of this comparison are shown for each benchmark. By performing a number of studies the influence of a number of settings are evaluated. Which studies are performed is explained in section A.1.1.

A.1.1 Study explanations

In order to know how well the numerical models in COMSOL calculate the problem, a mesh, setting and parameter study are performed. This section explains how every study is performed.

Mesh Study

In the mesh study the number of elements on the relevant edges are doubled for every refinement. It is expected that for a finer mesh the result will approach the analytical solution. The error between the model (sub m) and the analytical solution (sub a) is derived by equation A.1.

$$Error = \frac{|Var_m - Var_a|}{Var_a} \cdot 100 [\%] \quad (A.1)$$

For every benchmark the error will be plotted against the amount of elements in the mesh. The increase in elements will also cause an increase in computation time, which will also be plotted.

However the mesh is not the only parameter that influences the quality of the result. This means that the error may not approach zero but value slightly larger, which is then the most accurate result for these settings. In order to make a good assumption of what the quality of the mesh should be, the difference in error for every step is calculated. The difference for every step, is an indication of how much the accuracy of the solution improves for that refinement. It is expected that the solution difference gets smaller after every step. This difference is plotted for every step.

Setting Study

There are three radiation settings which influence the quality of the solution: *Ray resolution*, *Tolerance* and *Maximum number of adaptations*. In the COMSOL Documentation the computation of the *Ray shooting* algorithm and these settings is explained as followed:

*"To compute the radiation intensity on surfaces, the ray shooting algorithm emits n rays in 2D and n^2 rays in 3D where n is the value selected for **Radiation resolution**. The trajectories of these rays are computed as they are absorbed, reflected or transmitted on the model surfaces until their intensity becomes too small or if the rays go far away from the geometry. The threshold where the ray trajectory is no longer computed is controlled by the **Tolerance**. During the rays trajectory computation the tiling is adapted up to a numbers of time defined by the **Maximum number of adaptations**."*

The influence of these settings on the quality of the solution and computation time is evaluated in the same way as the number of elements.

Parameter Study

The mesh and setting study are performed for one set of parameters. It is important to know if the quality of the solution is different for certain parameter sets. This is done by computing the model for different sets of parameters and evaluate the quality of the solution and computational time.

A.2 Concentric Cylinders

A.2.1 Refinement and Parameter Studies

For the mesh and setting refinement studies the parameters from the table below are used.

Parameter	Value
r_i	0.1 <i>m</i>
r_o	0.3 <i>m</i>
L	1 <i>m</i>
T_i	500 <i>K</i>
T_o	300 <i>K</i>
ϵ_i	0.5
ϵ_o	0.1

When a setting is not changed it has the value stated in the table below.

Setting	Value
Radiation Resolution (Mesh Study)	512 (1024)
Tolerance	1e-4
Maximum number of adaptations	6
Total number of elements	640

The total heat flux is evaluated over the inner cylinder and over the outer cylinder. Both total fluxes are plotted in the results plots.

Mesh Study

From figures A.1 and A.2 can be derived that the model converges to a solution for the total heat flux. The accuracy is at the cost of the computational time. The accuracy of the solution keeps increasing for a higher number of elements. The ideal number of elements, depends strongly on the computational resources available.

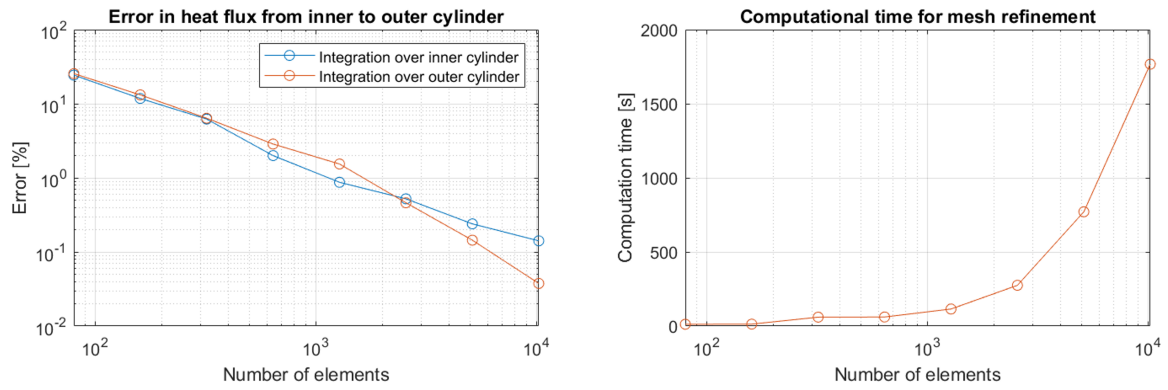


Figure A.1: Computation time and error in total heat flux for mesh refinement.

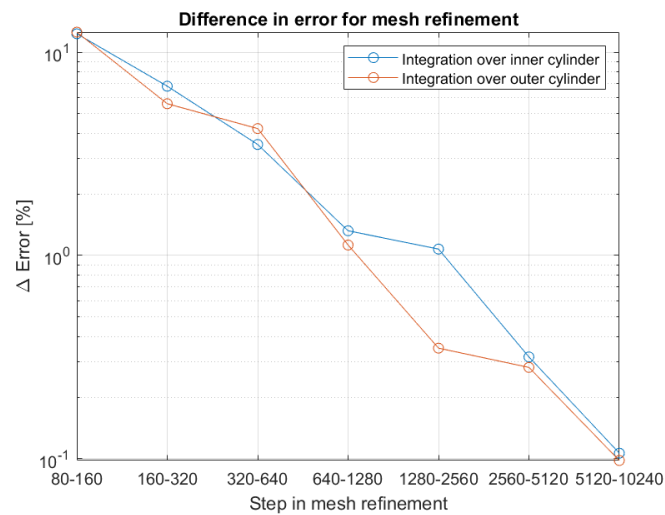


Figure A.2: Change in error for every refinement step.

Ray Resolution

From figures A.3 and A.4 can be derived that the model converges to a solution for the total heat flux. The accuracy is at the cost of the computational time. When changing the ray resolution above 512 the change in error can be still above 10^{-1} . It is seen that the value for the inner and outer cylinder approach each other which is expected from energy conservation.

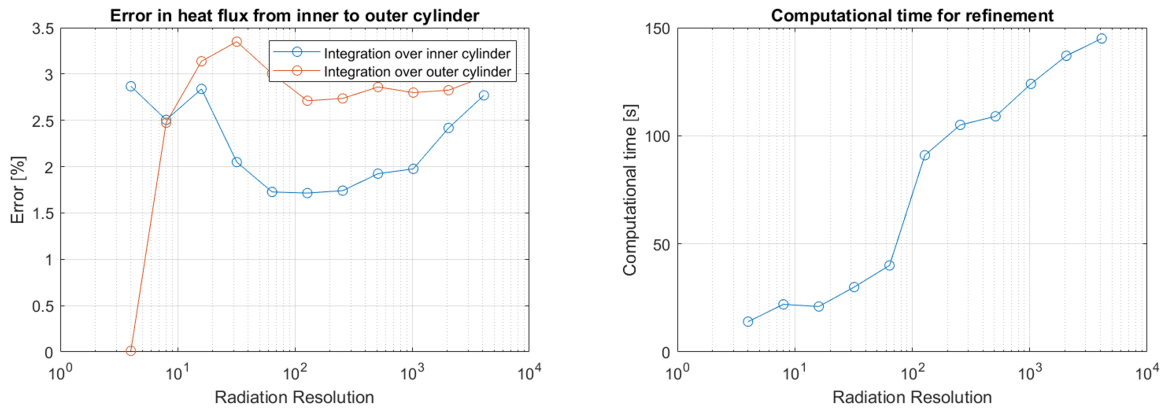


Figure A.3: Computation time and error in total heat flux for ray resolution refinement.

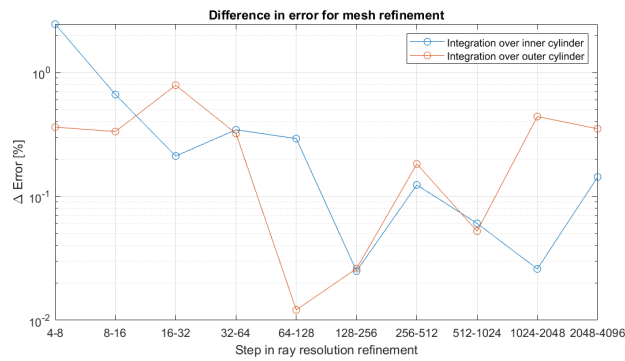


Figure A.4: Change in error for every refinement step.

Tolerance

From figures A.5 and A.6 can be derived that for a tolerance equal to or lower than 10^{-2} the solution is converged. The tolerance does have a large impact on the computational time.

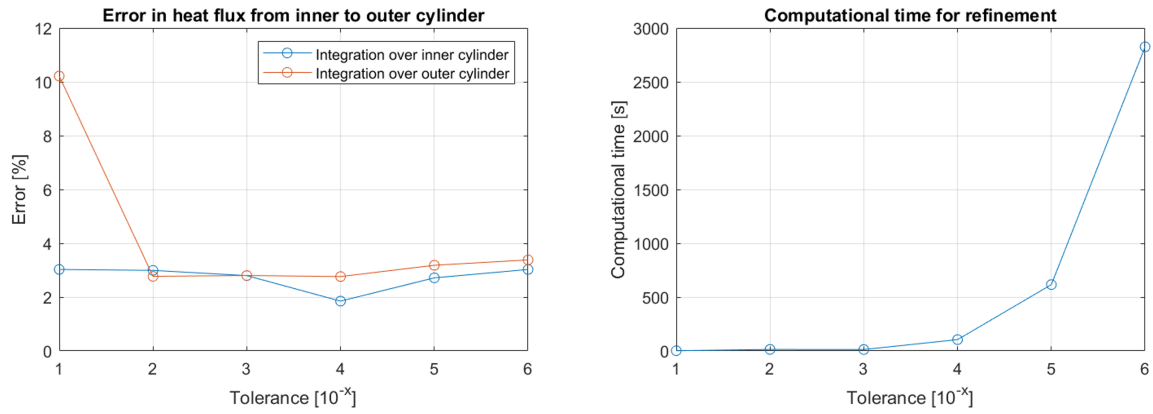


Figure A.5: Computation time and error in total heat flux for tolerance refinement.

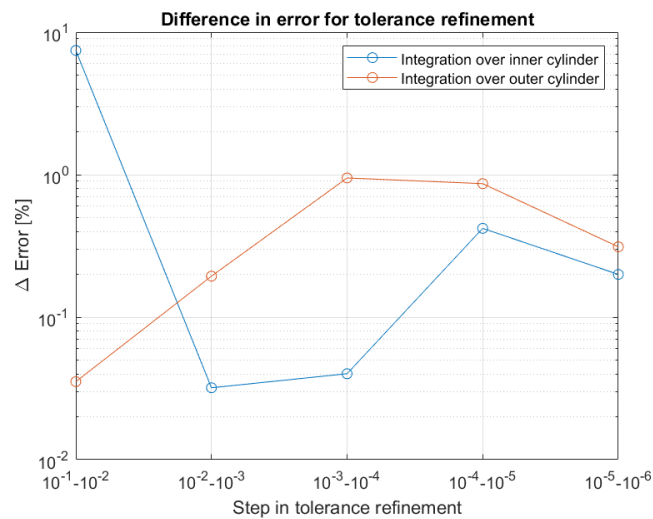


Figure A.6: Change in error for every refinement step.

Maximum number of adaptations

From figures A.7 and A.8 can be derived that the model converges to a solution for the total heat flux. Increasing the maximum number of adaptations above 4 increases the computational time significantly, however there is no clear relation when increasing the value even more. At a value of 4 for the maximum number of adaptations the difference in error stabilizes.

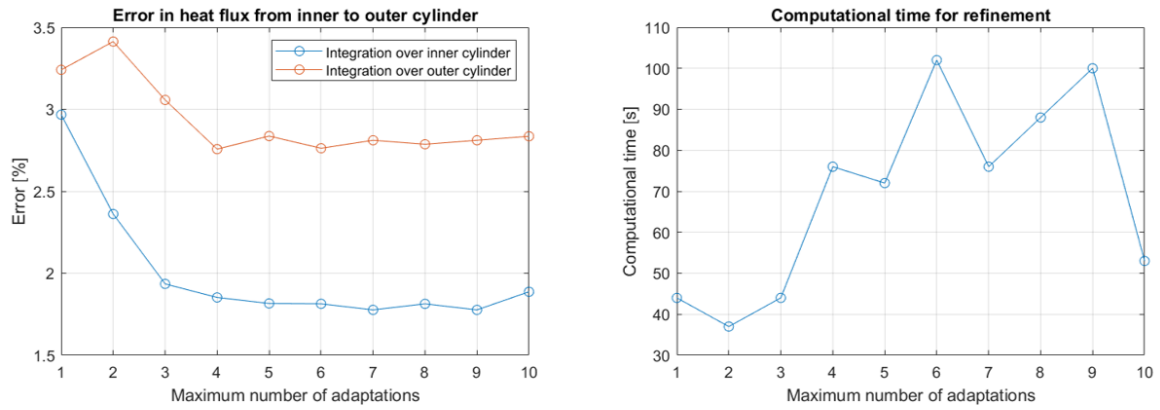


Figure A.7: Computation time and error in total heat flux for maximum number of adaptations refinement.

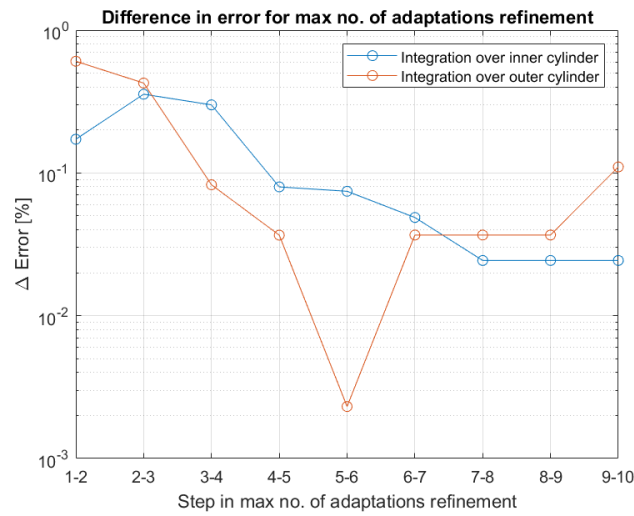


Figure A.8: Change in error for every refinement step.

Parameter Study

The settings for the parameter study are shown in the table below. The values for these settings are chosen based on the results from the mesh and setting study.

Setting	Value
Ray Resolution	512
Tolerance	1e-5
Maximum number of adaptations	5
Total number of elements	1280

For the parameter study the outer cylinder radius, the emissivity of the inner and outer cylinder are varied. The inner cylinder radius is not changed, because the difference in ratio between the two cylinders changes when r_o is changed. The table below shows the analytical value for the heat flux ($Q_{analytic}$) and the COMSOL results evaluated on the inner cylinder (Q_{COM_i}) and the outer cylinder (Q_{COM_o}). The error between the COMSOL model and the analytical solution is shown for both cases where the integration is performed over the inner and outer cylinder. The computation time (t_{comp}) is also shown in the table.

From the results can be derived that the error increases when the distance between the cylinders is larger and when there is a lot of reflection, so a low emissivity. The error is larger when the flux is evaluated over the outer cylinder than when evaluated over the inner cylinder.

r_o	ϵ_1	ϵ_2	$Q_{analytic}$	Q_{COM_i}	Q_{COM_o}	Q_{err_i} [%]	Q_{err_o} [%]	t_{comp} [s]
0.15	0.5	0.1	176.19672	177.09931	178.77724	0.51226	1.4646	218
0.15	0.5	0.5	646.05464	646.62934	648.20507	0.088955	0.33286	6
0.15	0.5	0.8	861.40619	861.60328	862.26813	0.02288	0.10006	2
0.15	0.9	0.1	191.68654	192.75594	194.58213	0.55789	1.5106	245
0.15	0.9	0.5	918.07765	919.20843	921.53788	0.12317	0.3769	5
0.15	0.9	0.8	1423.9572	1424.5278	1425.5989	0.040075	0.11529	4
0.3	0.5	0.1	176.19672	178.68903	182.63325	1.4145	3.653	480
0.3	0.5	0.5	646.05464	648.04356	649.64417	0.30786	0.55561	7
0.3	0.5	0.8	861.40619	862.17606	862.63008	0.089373	0.14208	5
0.3	0.9	0.1	191.68654	194.63715	198.93687	1.5393	3.7824	303
0.3	0.9	0.5	918.07765	922.10538	924.39172	0.43871	0.68775	14
0.3	0.9	0.8	1423.9572	1426.0901	1426.8332	0.14979	0.20197	3
0.6	0.5	0.1	176.19672	181.58937	190.36324	3.0606	8.0402	343
0.6	0.5	0.5	646.05464	650.74039	652.86486	0.72529	1.0541	16
0.6	0.5	0.8	861.40619	862.95024	863.99587	0.17925	0.30063	6
0.6	0.9	0.1	191.68654	198.08251	207.65824	3.3367	8.3322	565
0.6	0.9	0.5	918.07765	927.56638	930.68782	1.0335	1.3735	11
0.6	0.9	0.8	1423.9572	1428.208	1429.9324	0.29852	0.41962	5

A.2.2 Analytical Calculation

The analytical calculation of the concentric cylinder benchmark is calculated using the following MatLab function::

```
% Function that calculates the heat flux from the inner to the outer
% cylinder in a concentric cylinder setup. The inputs are:
% r1 - Radius of inner cylinder [m]
% r2 - Radius of outer cylinder [m]
% L - Length of cylinders [m] (in 2D model choose L = 1)
% T1 - Temperature of inner cylinder [K]
% T2 - Temperature of outer cylinder [K]
% e1 - Emission coefficient of inner cylinder [-]
% e2 - Emission coefficient of outer cylinder [-]
% diffspec - State if the reflection of the outer cylinder is diffuse
%           or
%           specular. Only accepts 'Diff' or 'Spec'

function [Q12] = ConCylSim(r1,r2,L,T1,T2,e1,e2,diffspec)
sigma = 5.670374419e-8;      % Stefan-Boltzmann constant [W m^-2 K^-4]
A1 = 2*pi*r1*L;            % Inner cylinder area [m^2]
A2 = 2*pi*r2*L;            % Outer cylinder area [m^2]

if diffspec == 'Diff'      % Calculation for diffuse reflection of
    outer cylinder
    Q12 = A1*sigma*(T2^4-T1^4)/(1/e1+A1/A2*(1/e2-1)); % Heat flux
        from inner to outer cylinder [W]
elseif diffspec == 'Spec' % Calculation for specular reflection of
    outer cylinder
    Q12 = A1*sigma*(T2^4-T1^4)/(1/e1+1/e2-1); % Heat flux
        from inner to outer cylinder [W]
else
    disp('Fill in Diff or Spec for diffspec') % Error for
        wrong input for diffspec
    Q12 = 'error'
end
```

A.2.3 Numerical Calculation

The numerical calculation of the concentric cylinder benchmark is performed using the *Surface-to-Surface Radiation* module in COMSOL. The figures in this section describe how the model is developed.

Name	Expression	Value	Description
r1	0.1[m]	0.1 m	Radius of Inner Cylinder
r2	0.3[m]	0.3 m	Radius of Outer Cylinder
T1	500[K]	500 K	Temperature of Inner Cylinder
T2	300[K]	300 K	Temperature of Outer Cylinder
e1	0.5	0.5	Emissivity of Inner Cylinder
e2	0.1	0.1	Emissivity of Outer Cylinder
rhos1	0	0	Specularity of Inner Cylinder
rhos2	1 - e2	0.9	Specularity of Outer Cylinder
M1	120	120	Number of elements on 1/4th of Inner Cylinder
M2	200	200	Number of elements on 1/4th of Outer Cylinder

Figure A.9: The parameter list used to describe the geometry and the physics

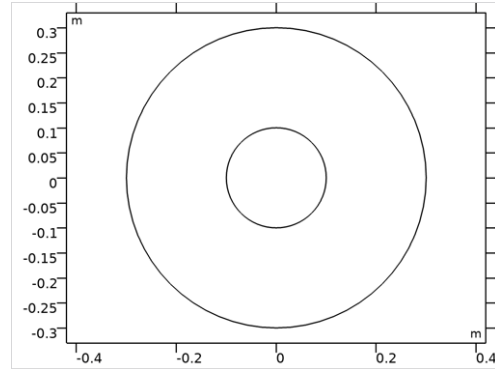


Figure A.10: The geometry of the model

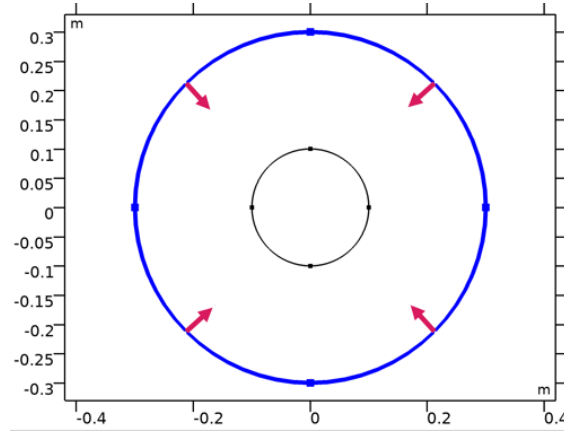
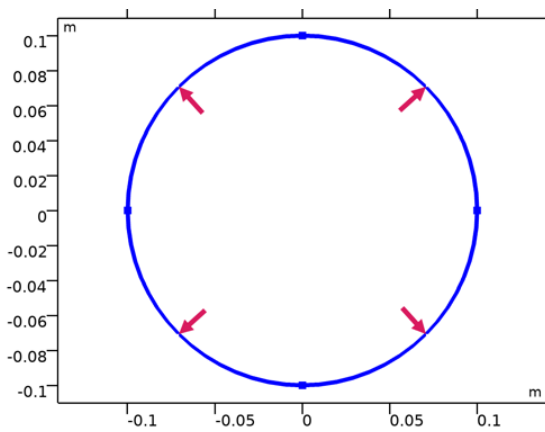


Figure A.11: The opaque surface boundary condition applied at the inner and outer cylinder. The applied values are shown in the parameter list

Description	Value
Use radiation groups	Off
Wavelength dependence of surface properties	Constant
Surface-to-surface radiation method	Ray shooting
Radiation resolution	512
Maximum number of adaptations	5
Transparent media refractive index	1
Tolerance	1e-5

Figure A.12: Radiation Settings

Description	Value
Minimum element quality	1.0
Average element quality	1.0
Edge element	1280
Vertex element	8

Figure A.13: Mesh Settings

A.3 Isothermal Parallel Plates

A.3.1 Refinement and Parameter Studies

For the mesh and setting refinement studies the parameters from the table below are used. When a setting is not changed it has the value as stated in the table.

Setting	Value
γ	0.5
ϵ	0.1
Radiation Resolution	1024
Tolerance	1e-6
Maximum number of adaptations	6
Number of elements per plate	800

The total and local heat flux are evaluated over one of the plates. The average and the maximum error of the local heat flux is evaluated.

Mesh Study

From figures A.14 and A.15 can be derived that the model converges to a solution for the heat flux. The accuracy is at the cost of the computational time. At 400 elements per plate the difference in error is below 10^{-4} .

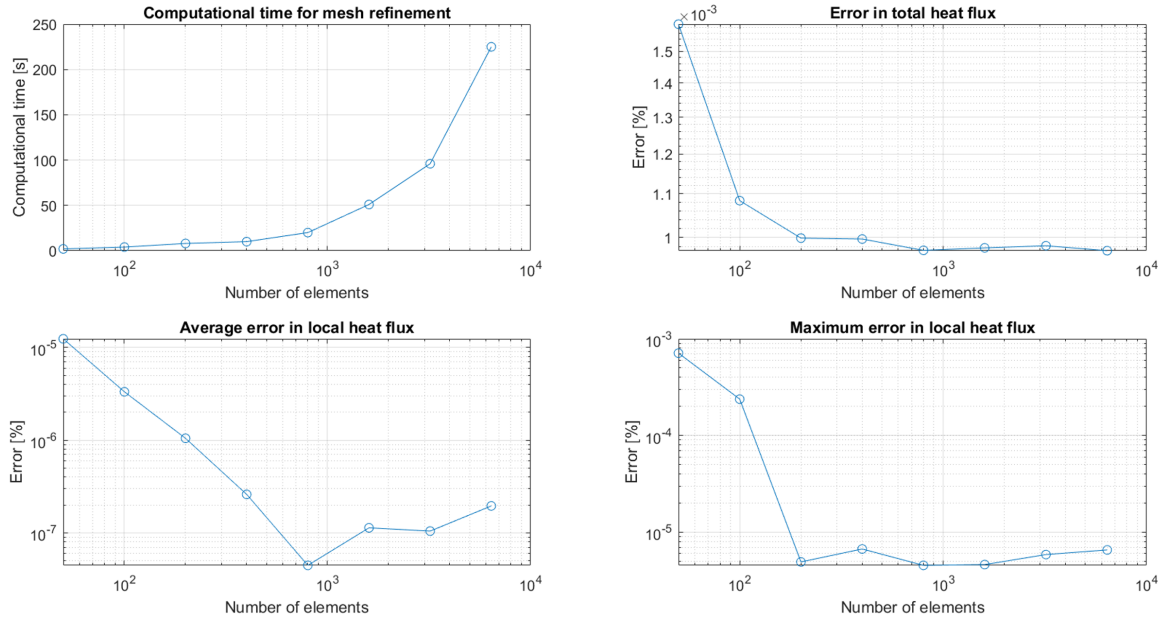


Figure A.14: Computation time and errors in heat flux for mesh refinement.

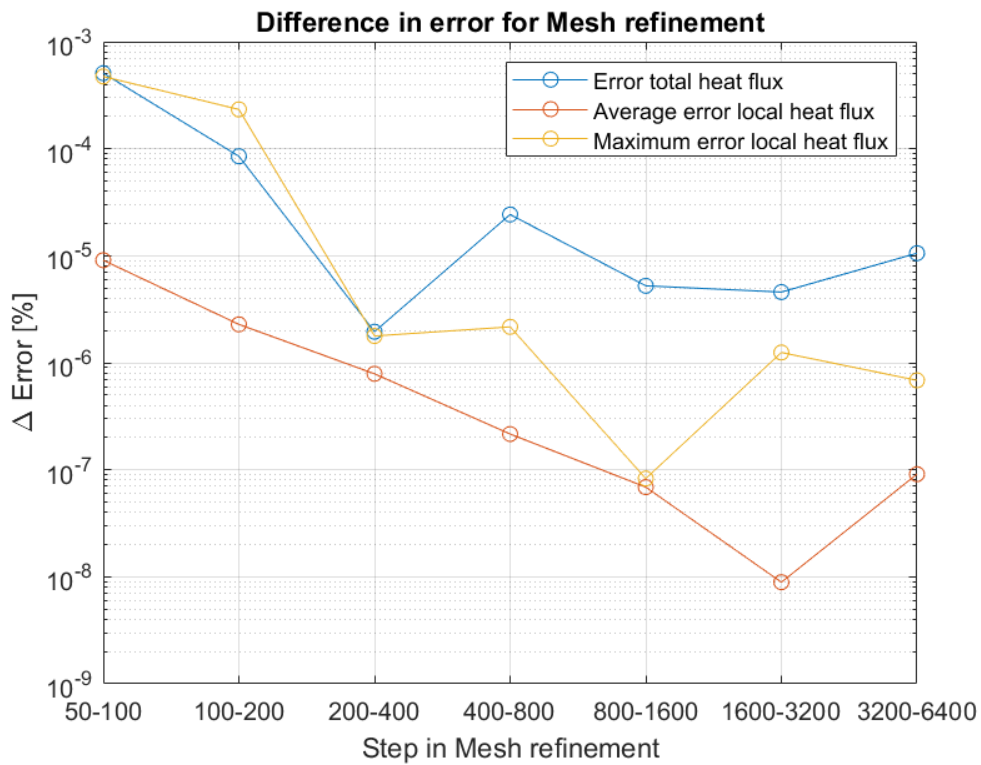


Figure A.15: Change in error for every refinement step.

Ray Resolution

From figures A.16 and A.17 can be derived that the model converges to a solution for the heat flux. The accuracy is at the cost of the computational time. At a ray resolution of 512, the difference in error stays below 10^{-3} .

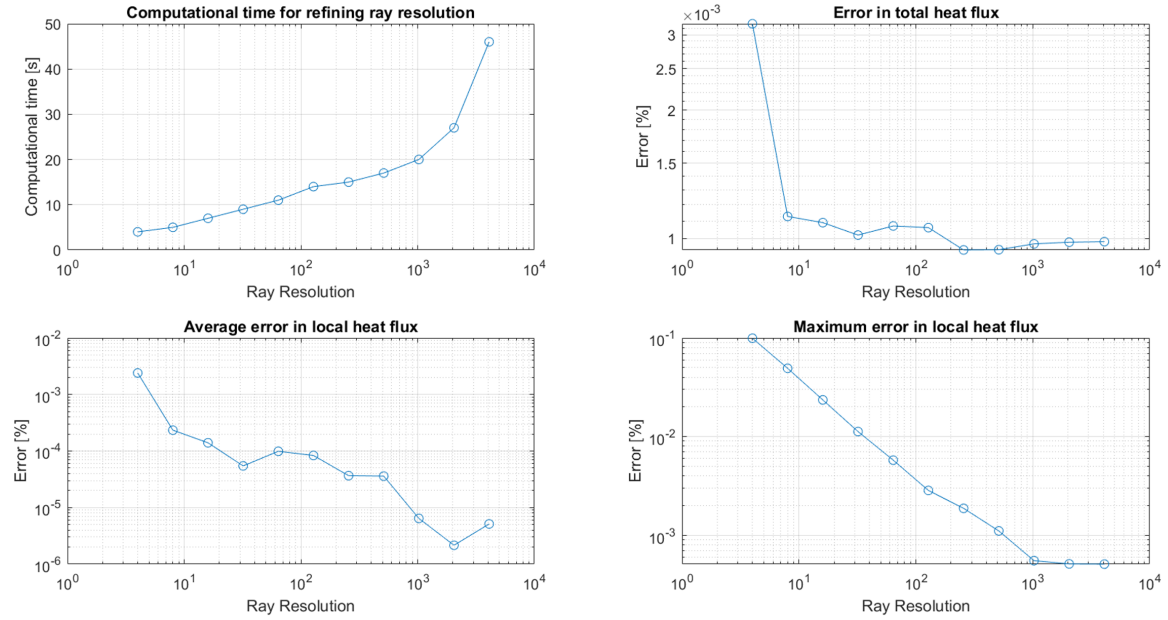


Figure A.16: Computation time and errors in heat flux for ray resolution refinement.

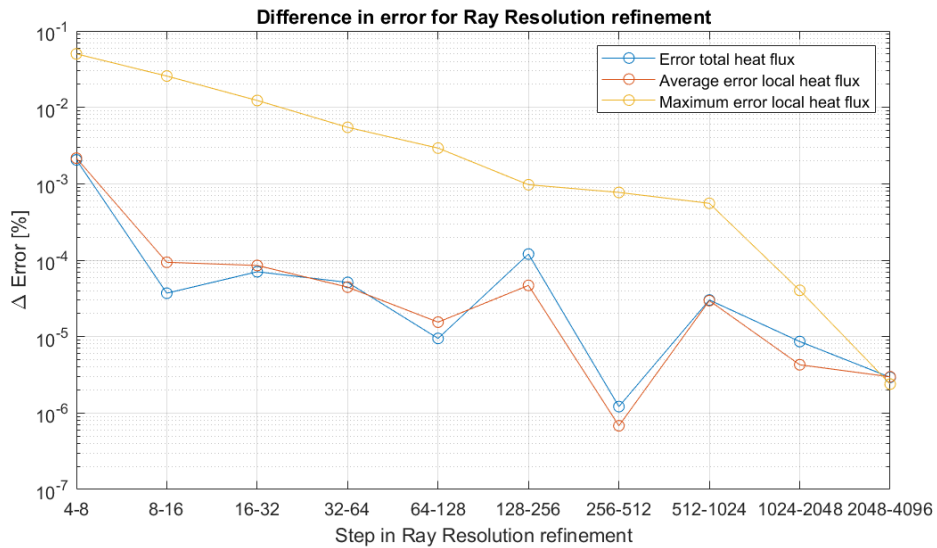


Figure A.17: Change in error for every refinement step.

Tolerance

From figures A.18 and A.19 can be derived that the model converges to a solution for the heat flux. The computational time is not significantly influenced by decreasing the tolerance. At a tolerance of 10^{-7} , the difference in error stays below 10^{-4} .

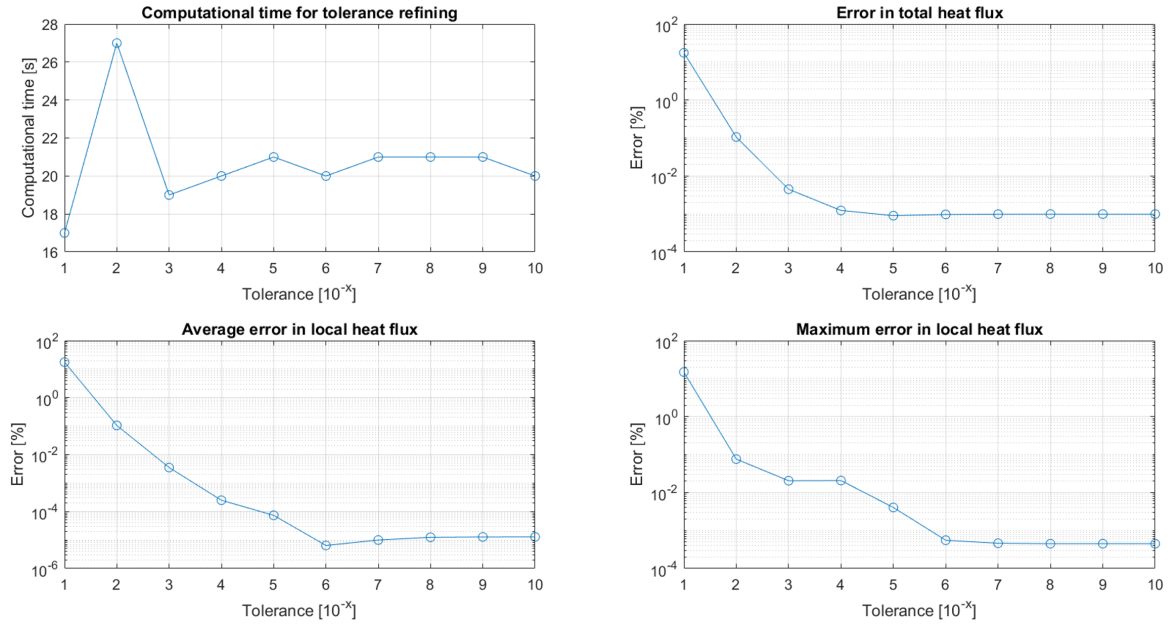


Figure A.18: Computation time and errors in heat flux for tolerance refinement.

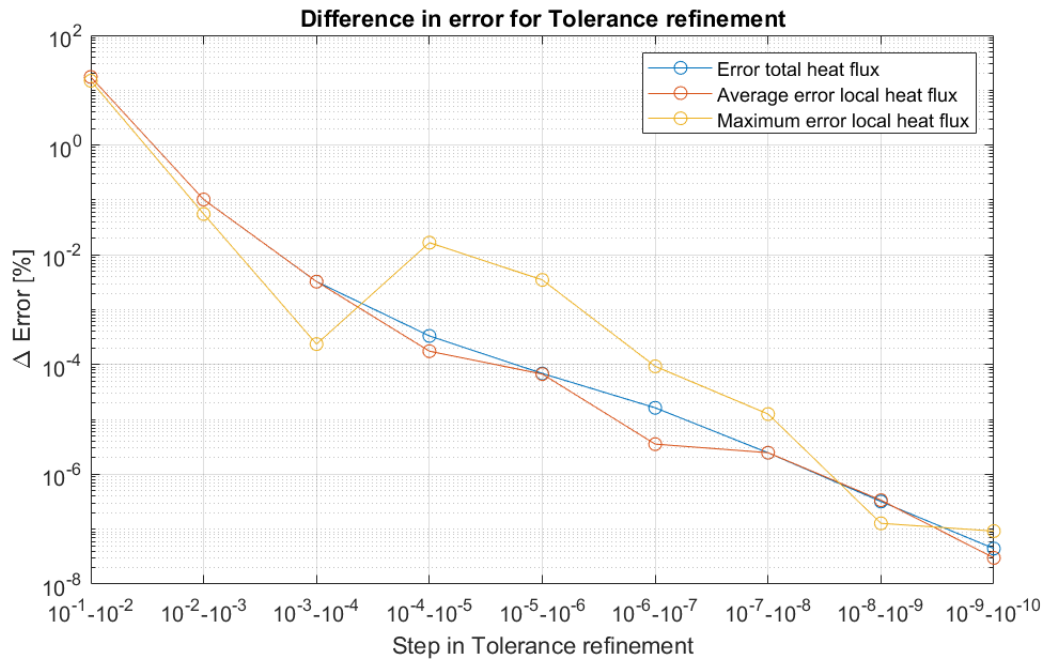


Figure A.19: Change in error for every refinement step.

Maximum number of adaptations

From figures A.20 and A.21 can be derived that the model converges to a solution for the heat flux. The computational time is not significantly influenced by decreasing the maximum number of adaptations. For a maximum number of adaptations of 5, the difference in error stays below 10^{-3} .

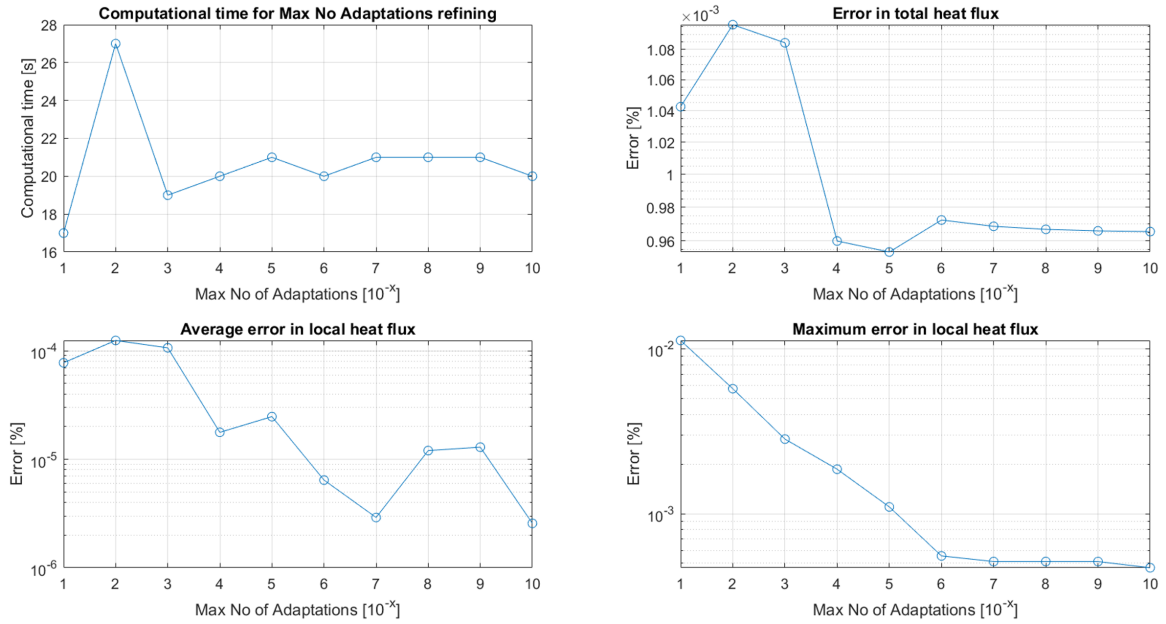


Figure A.20: Computation time and errors in heat flux for maximum number of adaptations refinement.

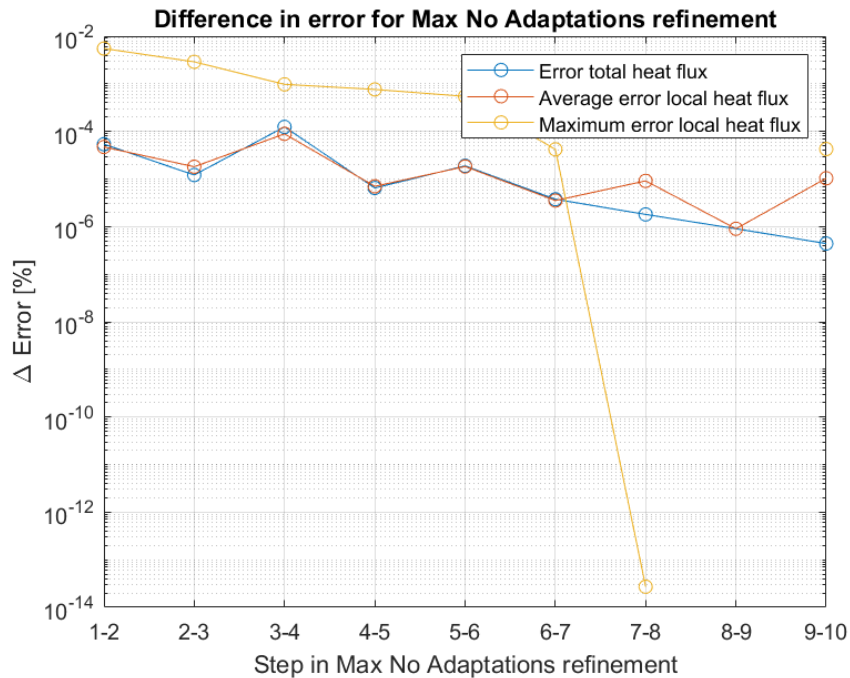


Figure A.21: Change in error for every refinement step.

Parameter Study

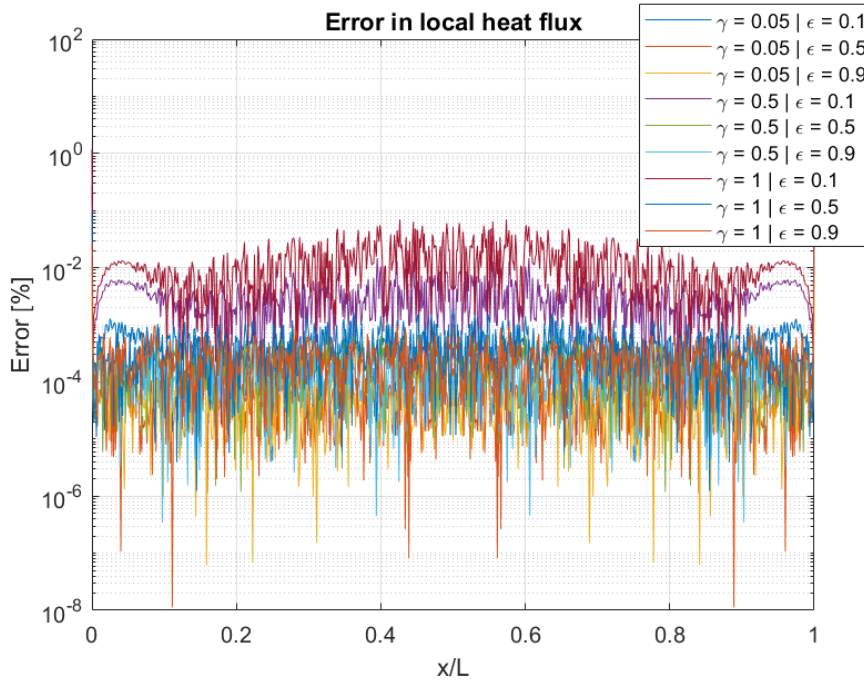
The parameter study is performed with the following settings:

Setting	Value
Radiation Resolution	2048
Tolerance	1e-6
Maximum number of adaptations	6
Number of elements per plate	800

The table below shows the error made in the total heat flux, the maximum and average error in the local heat flux, all in percentages.

γ	ϵ	Q_{an} [W/m]	Q_{COM} [W/m]	Q_{err}	$avg(q_{err}(x))$	$max(q_{err}(x))$	t_{comp}
0.05	0.1	14.787	14.787	8.0066e-07	0.0010555	0.27824	107
0.05	0.5	21.268	21.267	2.2702e-06	8.2623e-05	0.010833	45
0.05	0.9	22.265	22.264	1.9584e-06	5.6612e-05	0.0028777	23
0.5	0.1	36.503	36.502	7.8296e-07	0.0053083	0.86004	32
0.5	0.5	120.41	120.41	1.877e-06	0.00034715	0.055134	17
0.5	0.9	166.81	166.81	1.0847e-07	0.00018971	0.01523	13
1	0.1	40.536	40.536	7.0846e-07	0.015922	1.1841	30
1	0.5	160.87	160.87	1.9618e-06	0.00066786	0.093367	13
1	0.9	249.83	249.83	4.8075e-07	0.00033217	0.027702	15

The error in the local heat flux on a loglog-scale:



A.3.2 Analytical Calculation

The MatLab function shown on this pages shows the calculation of the total heat loss and the distribution of the heat loss for diffuse reflecting plates.

```

% This script calculates the total and local heat loss over the
% length of two diffuse isothermal parallel plates. The inputs are:
% epsilon: Emission coefficient of plates [-]
% gamma: Height over width ratio of the geometry [-]
% N: Number of evaluation points along the plates [-]
function [q,Q] = Diff_fun(epsilon,gamma,N)
%% Input parameters
rho = 1-epsilon; % [-] Reflectivity (diffuse)
%% Geometry parameters
X = linspace(0,1,N);% [-] Lower plate x/L
Y = linspace(0,1,N);% [-] Upper plate x/L
%% Initial values while loop
beta_y = 9*ones(1,N);
difference = 10;
count = 0;
%% While loop to calculate beta for every X
while difference > 0.000001 % Tolerance
    for i = 1:length(X)
        beta_int(i,:) = beta_y./((Y-X(i)).^2 + gamma^2).^(3/2); %
            The vector that needs to be integrated
        beta_x(i,:) = 1 + rho*gamma^2/2*trapz(Y,beta_int(i,:)); %
            Calculation of beta
    end
    difference = max(abs(beta_x'-beta_y)); % Difference
        between the new and previous beta
    beta_y = beta_x'; % The new beta is
        the input for the next calculation
    count = count + 1; % Counter to
        check the amount of iterations
end
%% Calculate the fluxes
H = (beta_x'-1)/rho; % Total flux arriving at a certain point
on the plate
q = (1-beta_x'*epsilon)/rho; % Heat loss at a certain point on the
plate
B = H+q; % Flux leaving a certain point on the
plate
Q_dl = trapz(X,q)*epsilon; % Total dimensionless heat loss
sigma = 5.670367e-8; % Stefan-Boltzmann constant
T = 300; % Plate temperature
Q = Q_dl*sigma*T^4; % Total heat loss
end

```

The MatLab function shown on this pages shows the calculation of the total heat loss and the distribution of the heat loss for specular reflecting plates.

```

% This script calculates the heat loss over the length of two
% specular
% isothermal parallel plates. Input values are:
% epsilon: Emissivity coefficient of the plates
% gamma: Height over width ratio of the plates
% mesh: Number of points along the plates
function [hl_spec,HL_dl] = ParPlat_spec_fun(epsilon,gamma,mesh)
% Constants
T = 300; % Isothermal plates temperature [K]
L = 1; % Plate length [m]
Tamb = 0; % Ambient temperature [K]
sigma = 5.670367e-8; % Stefan-Boltzmann constant [W?m?2?K?4]
alpha = epsilon; % Absorptivity [-]
h = gamma*L; % Height between plates [m]
eb = sigma*T^4; % Black body plate emission [W.m-2]
% View Factor calculation for No reflections
x = linspace(0,L,mesh); % x-coordinate along plate [m]
X = x/L; % normalized x-coordinate [-]
% View Factor calculation for k reflections
j = 1:length(X); % Vector along plate coordinates
for i = 1:length(epsilon) % Loop over multiple emissivities
    rho(i) = 1-epsilon(i); % Reflectivity [-]
    k = 0; % Starting value for no. of
        reflections
        hl_k(i,j) = ones(1,length(X)); % Initial vector value
        hl_sum(i,j) = zeros(1,length(X)); % Initial vector values
        while hl_k(i,j) > 1e-8 % Set accuracy of calculation
            hl_k(i,j) = rho(i)^k*(X./sqrt(X.^2 + (k+1)^2*gamma^2) + (1-X)
                ./sqrt((1-X).^2 + (k+1)^2*gamma^2)); % Local heat loss
                for k reflections
                    hl_sum(i,j) = hl_sum(i,j) + hl_k(i,j); % Summation of
                    local heat loss over all reflections
                k = k + 1; % Number of reflections [-]
            end
            hl_spec(i,j) = 1 - alpha(i)/2*hl_sum(i,j); % Final value for
            local heat loss
            q_spec(i,j) = hl_spec(i,j) .* epsilon(i) * sigma * T^4; % Local
            heat loss [W]
            HL_sum(i) = 0; % Initial value
            HL_k(i) = 1; % Initial value
            k=0; % Initial value
            while HL_k(i) > 0.00001 % Set accuracy for total heat loss
                HL_k(i) = rho(i)^k*(sqrt(1+(k+1)^2*gamma^2) - (k+1)*gamma); %
                Total heat loss for k reflections
                HL_sum(i) = HL_sum(i) + HL_k(i); % Summation of total heat
                loss over all reflections
                k = k + 1; % Number of reflections [-]
            end
            HL_dl(i) = epsilon(i)*(1 - alpha(i)*HL_sum(i)); % Final value for
            total heat loss
            HL(i) = HL_dl(i)*sigma*T^4; % Total heat loss [W]
        end
    end
end
end

```

A.3.3 Numerical Calculation

The numerical calculation of the isothermal parallel plates benchmark is performed using the *Surface-to-Surface Radiation* module in COMSOL. The figures in this section describe how the model is developed.

Name	Expression	Value	Description
gamma	0.05	0.05	Ratio between the height and width of the plates
L	1[m]	1 m	Width of the plates
height	L*gamma	0.05 m	Height between the plates
T_iso	300[K]	300 K	Temperature of the plates
T_vac	0[K]	0 K	Temperature of the surroundings
e_plates	0.1	0.1	Emission coefficient of the plates
e_sur	1	1	Emission coefficient of the surroundings
rhos_plates	1 - e_plates	0.9	Specularity of the plates
rhos_sur	0	0	Specularity of the surroundings
M	800	800	Number of elements along each plate

Figure A.22: The parameter list used to describe the geometry and the physics

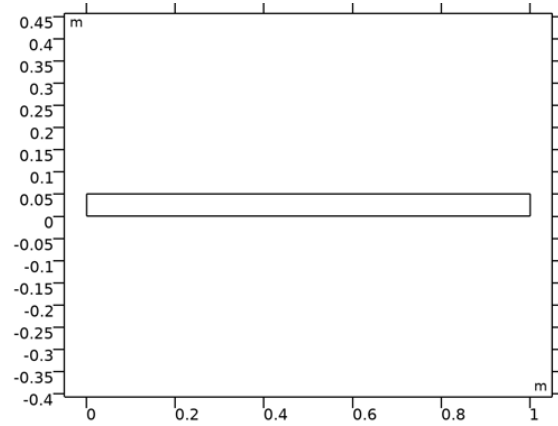


Figure A.23: The geometry of the model

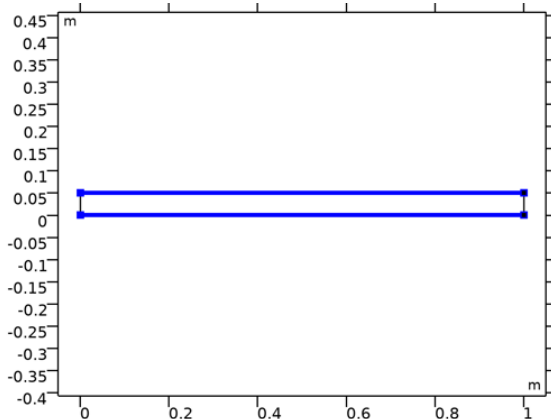


Figure A.24: The opaque surface boundary condition describing the reflecting plates

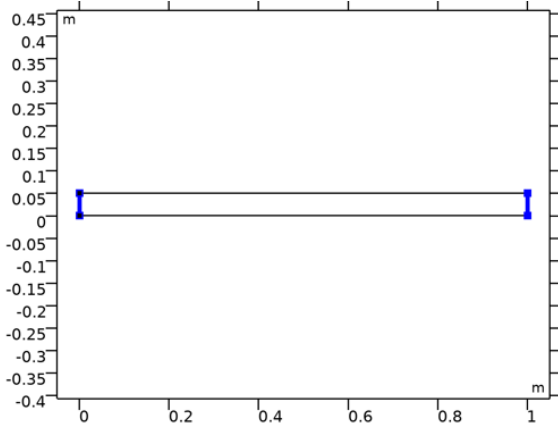


Figure A.25: Diffuse surface that has a temperature of 0K and an emission coefficient of 1, which means all irradiation is absorbed and there is no emission.

Description	Value
Use radiation groups	Off
Wavelength dependence of surface properties	Constant
Surface-to-surface radiation method	Ray shooting
Radiation resolution	2048
Maximum number of adaptations	6
Transparent media refractive index	1
Tolerance	1e-6

Figure A.26: Radiation Settings

Description	Value
Minimum element quality	1.0
Average element quality	1.0
Edge element	1626
Vertex element	4

Figure A.27: Mesh Settings

A.4 Wedge with Conducting Plates

A.4.1 Analytical solution accuracy

The analytical solution of the *Wedge* benchmark is used to analyze the quality of the COMSOL models. It is important to know the accuracy of this solution. The analytical calculation is modeled in MatLab. In order to create the heat flux and temperature distributions, the domain consists of discrete measuring points. The dimensionless temperature distribution is derived by the following equation:

$$\theta(\xi) = 1 + \epsilon N_c \left(\int_0^\xi \theta^4(\zeta)(\xi - \zeta)d\zeta - \int_0^1 \theta^4(\zeta)K_\gamma(\xi, \zeta)d\zeta \right) \quad (\text{A.2})$$

In this equation the integral is taken from 0 to ξ , this means that for the first few points there is an error in the solution. This error can be minimized by taking as many measuring points as possible. However this takes a long time to compute, since the solution needs to converge.

The dimensionless local heat flux is a function of the dimensionless temperature distribution. Described by the equation below. Since the temperature distribution is to the power of 4, the error made in the dimensionless temperature is enlarged in the dimensionless local heat flux.

$$\frac{q(\xi)}{\epsilon \sigma T_b^4} = \theta(\xi) - \epsilon \int_0^1 \theta^4(\zeta)G_\gamma(\xi, \zeta)d\zeta \quad (\text{A.3})$$

A study is performed where the amount of measuring points is increased by doubling the amount of points for every step. Figure A.28 shows the analytical solution for the local heat flux. In the zoomed in figure can clearly be seen that there is an error in the first measuring points of the analytical solution. For increasing the measuring points, the location of the peak moves towards $\xi = 0$.

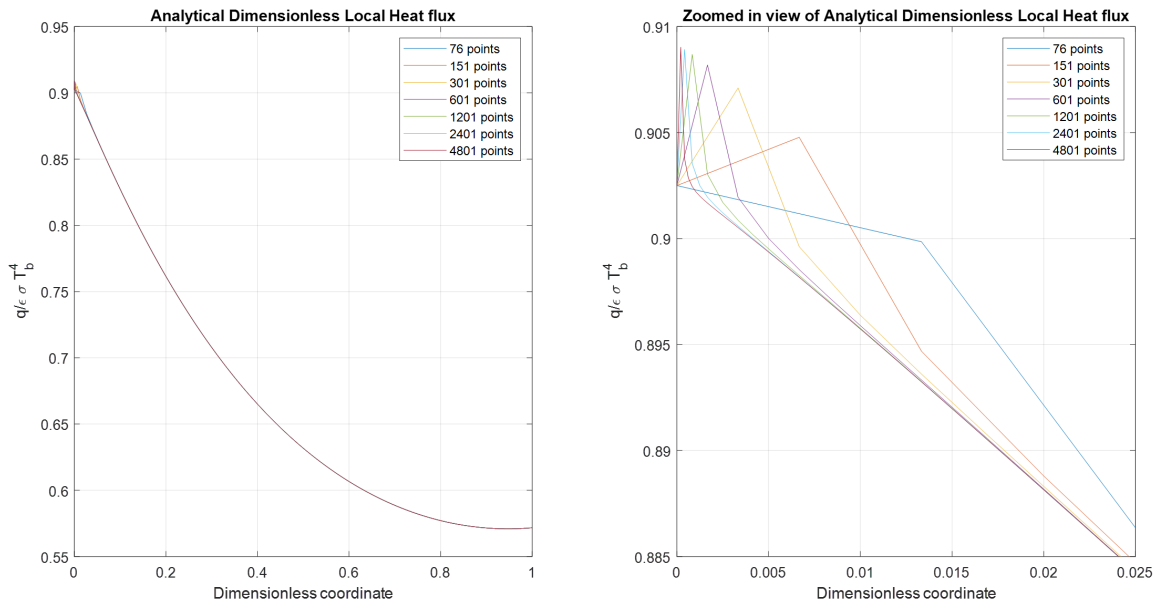


Figure A.28: Analytical solution for the dimensionless local heat flux for different amount of measuring points. The figure on the right is a zoomed in image of the left figure.

Figure A.29 shows the percentage difference in local heat flux and temperature for every step where the amount of measuring points is doubled. For every step the percentage difference gets smaller over the total domain. However for values very close to $\xi = 0$ the error is still relatively large for the local heat flux.

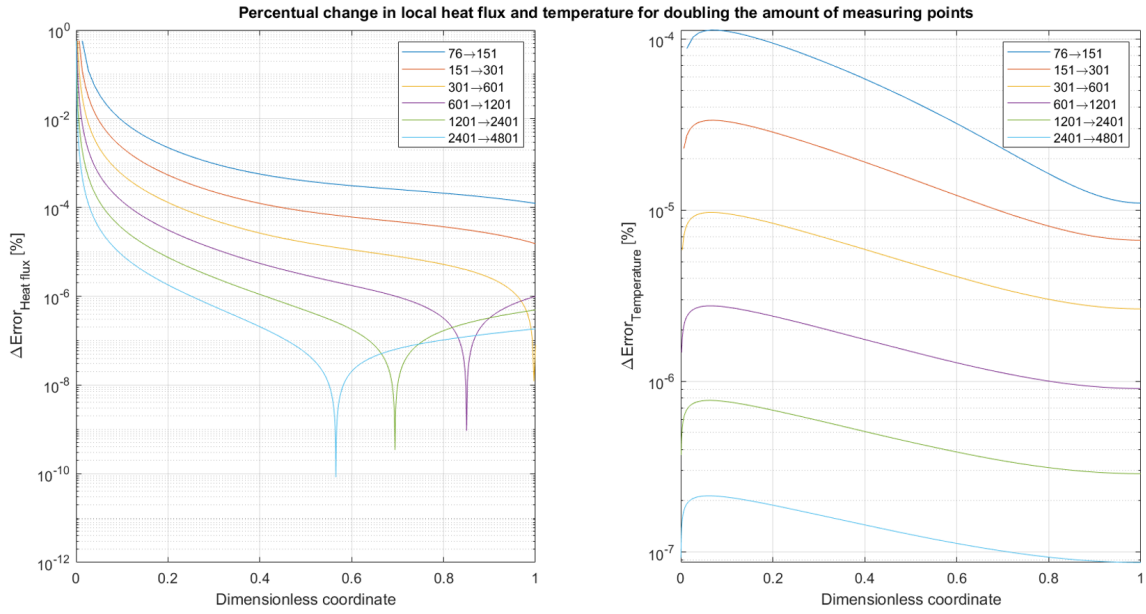


Figure A.29: Percentage change for the local heat flux (left) and temperature (right), for doubling the amount of measuring points in the analytical model.

The analytical solution that consists of 4801 measuring points is used as a reference for the mesh, setting and parameter study. From figure A.29, it can be concluded that the accuracy of the analytical temperature over the whole domain is at least 10^{-6} . For the local heat flux the accuracy is at least 10^{-3} for the domain $\xi > 0.01$. In the other studies described in this report, when deciding the maximum error of the heat flux, measuring points of the local heat flux for $\xi < 0.01$ are omitted. This is done, because the analytical solution is not accurate in this domain.

A.4.2 Refinement and Parameter Studies

For the mesh and setting refinement studies the parameters from the table below are used. When the setting is not changed it has the value as stated in the table below.

Parameter / Setting	Value
γ	60°
ϵ	0.1
N_c	4
Radiation Resolution (Mesh Study)	512 (1024)
Tolerance	$1e-5$
Maximum number of adaptations	5
Number of elements per plate	400

The heat flux depends on the temperature to the power of 4, it is thus expected that the error is largest for the local heat flux.

Mesh Study

From figures A.30 and A.31 can be derived that the model converges to a solution for both heat flux and temperature. The accuracy is at the cost of the computational time. At 400 elements per plate the error does not change significantly for any of the results.

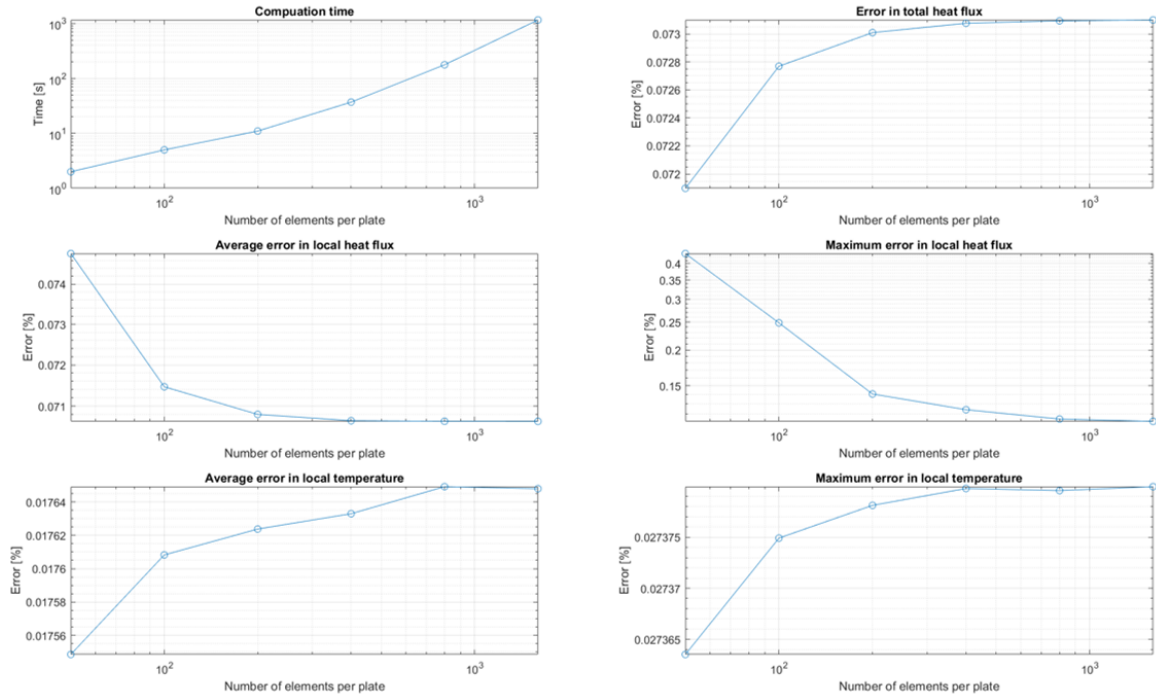


Figure A.30: Computation time and errors in heat flux and temperature for mesh refinement.

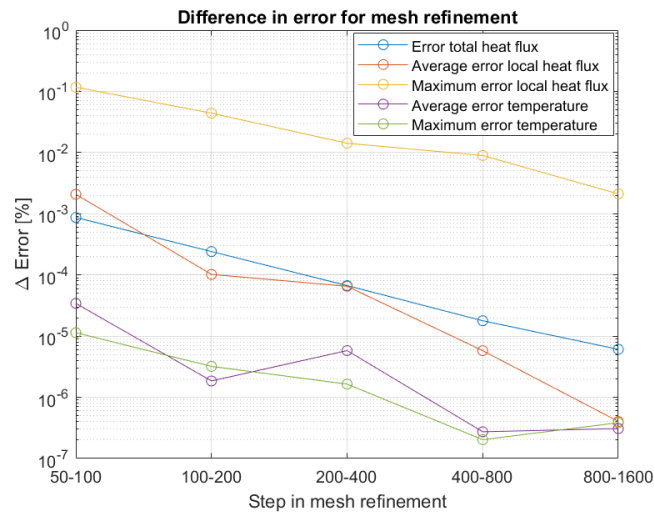


Figure A.31: Change in error for every refinement step.

Ray Resolution

From figures A.32 and A.33 can be derived that the model converges to a solution for both heat flux and temperature. The accuracy is at the cost of the computational time. At a ray resolution of 512 the difference in error is below 10^{-3} .

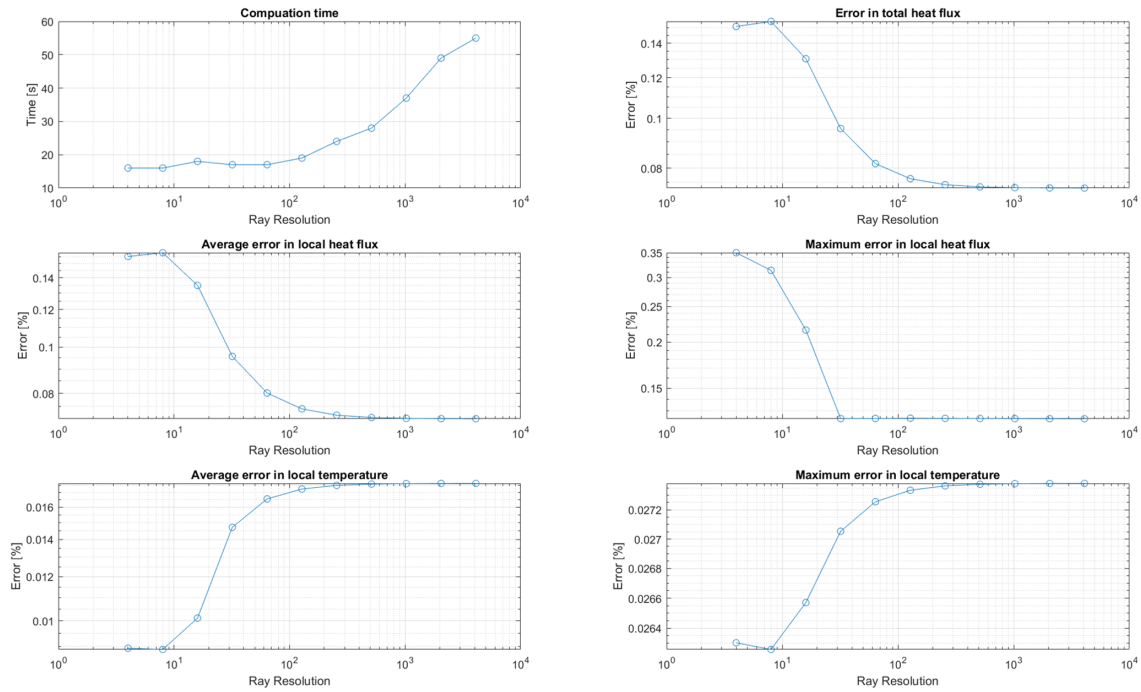


Figure A.32: Computation time and errors in heat flux and temperature for ray resolution refinement.

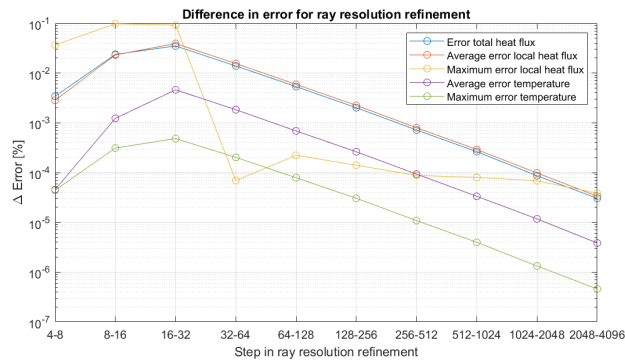


Figure A.33: Change in error for every refinement step.

Tolerance

From figures A.34 and A.35 can be derived that the model converges to a solution for both heat flux and temperature. There is no clear relation between the tolerance and the computational time. The result changes significantly when decreasing the tolerance until a value of 10^{-4} . Decreasing the tolerance further does not seem to improve the result much.

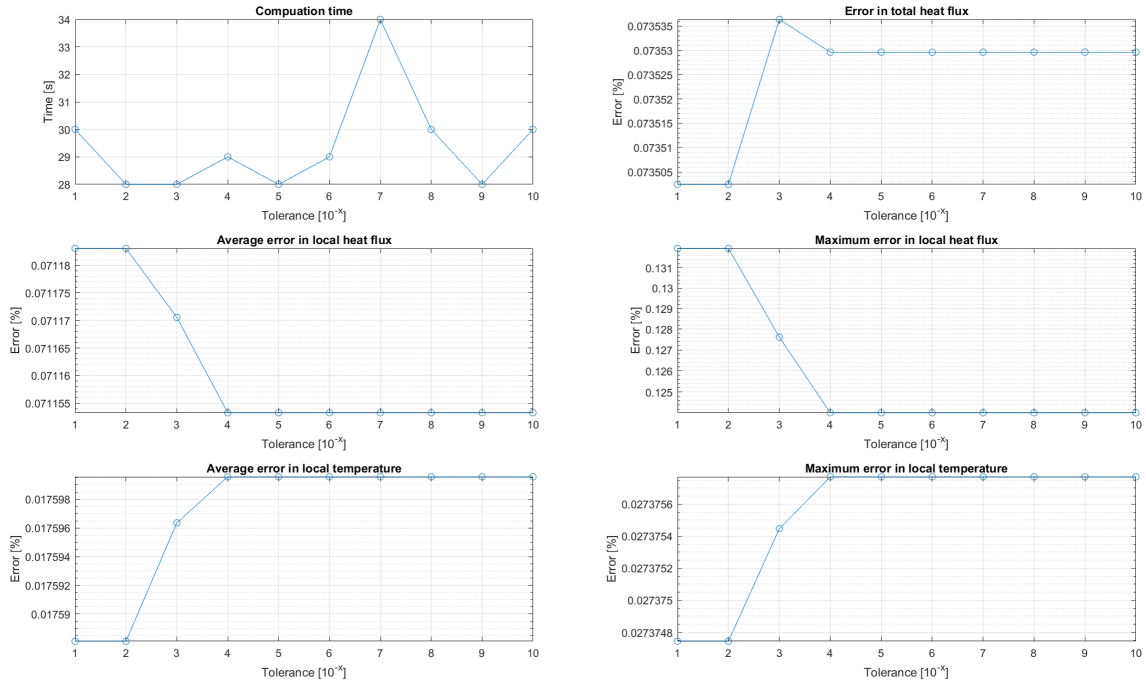


Figure A.34: Computation time and errors in heat flux and temperature for tolerance refinement.

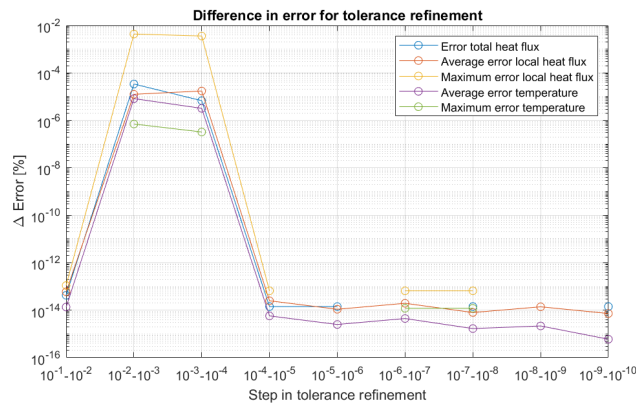


Figure A.35: Change in error for every refinement step.

Max No Adaptations

From figures A.36 and A.37 can be derived that the model converges to a solution for both heat flux and temperature. There is no clear relation between the maximum number of adaptations and the computational time. The influence of the maximum number of adaptations does not seem to have a very large influence on the result. Above a value of 4, the difference in error stays below 10^{-4} .

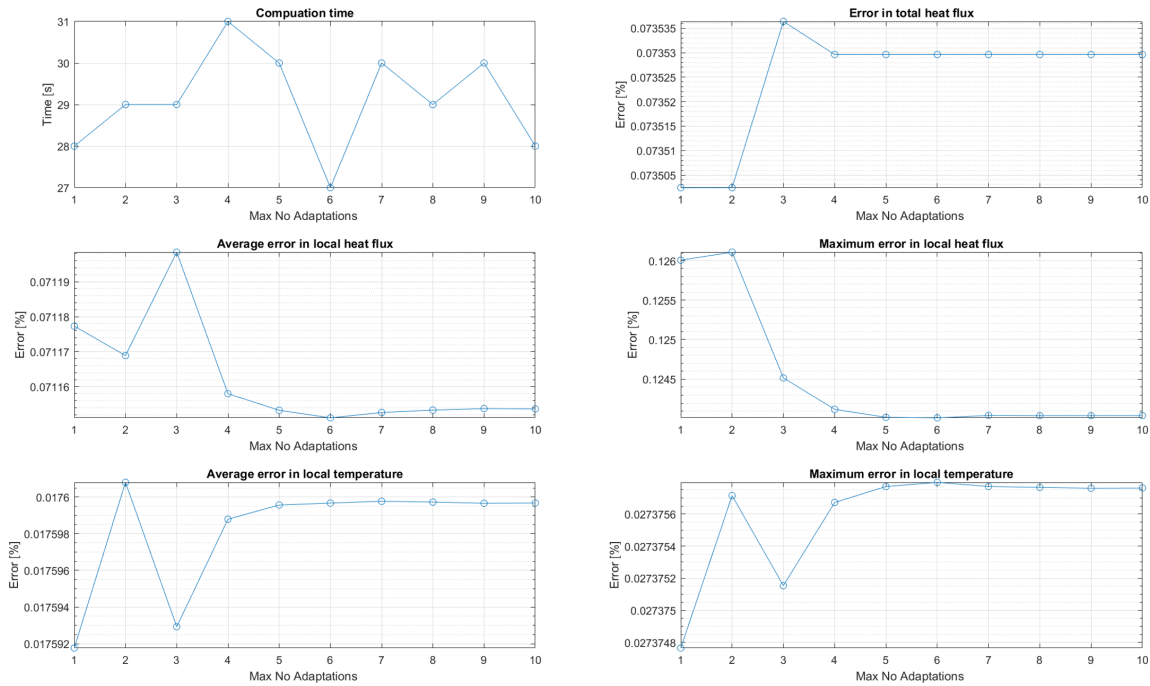


Figure A.36: Computation time and errors in heat flux and temperature for increasing the maximum number of adaptations.

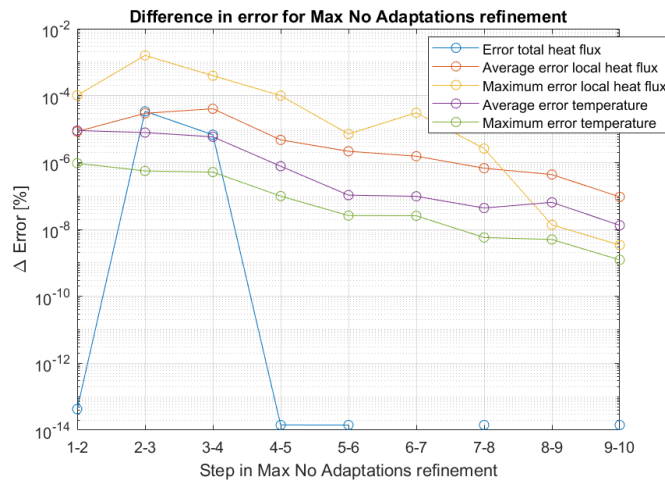


Figure A.37: Change in error for every refinement step.

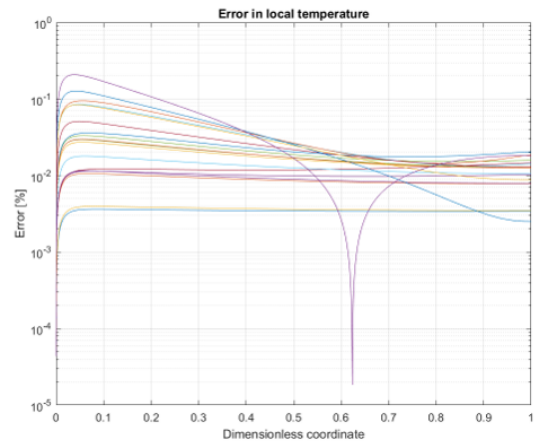
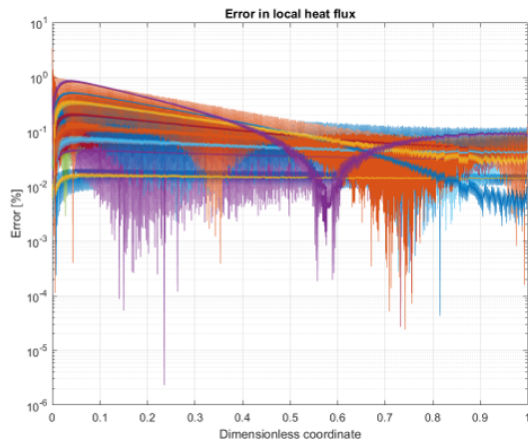
Parameter Study

The parameter study is performed for the following settings:

Setting	Value
Radiation Resolution	512
Tolerance	1e-5
Maximum number of adaptations	5
Number of elements per plate	400

The table below shows the error made in the total heat flux, the maximum and average error in the local heat flux and temperature, all in percentages. The error in the heat flux is larger than in the temperature. The error is larger for a larger value of the emissivity and a larger conduction parameter.

γ	ϵ	N_c	Q_{err}	$avg(q_{err})$	$max(q_{err})$	$avg(T_{err})$	$max(T_{err})$	t_{comp}
60	0.1	0.5	0.0139	0.0140	0.0212	0.0034	0.0036	22
60	0.1	1.5	0.0354	0.0354	0.0519	0.0088	0.0107	37
60	0.1	4	0.0730	0.0730	0.124	0.0176	0.0274	30
60	0.5	0.5	0.0410	0.0410	0.123	0.0102	0.0115	24
60	0.5	1.5	0.0856	0.0856	0.258	0.0209	0.0333	28
60	0.5	4	0.1631	0.163	0.531	0.0350	0.0854	28
60	0.9	0.5	0.0501	0.0696	0.315	0.0123	0.0147	20
60	0.9	1.5	0.0934	0.105	0.578	0.0234	0.0361	24
60	0.9	4	0.1696	0.147	0.981	0.0384	0.0950	27
135	0.1	0.5	0.0147	0.0371	0.0177	0.0036	0.0040	13
135	0.1	1.5	0.0372	0.0371	0.0485	0.0092	0.0116	17
135	0.1	4	0.0778	0.0748	0.121	0.0186	0.0296	19
135	0.5	0.5	0.0536	0.0530	0.0850	0.0132	0.0180	12
135	0.5	1.5	0.1135	0.102	0.223	0.0256	0.0511	15
135	0.5	4	0.2250	0.155	0.539	0.0397	0.1275	16
135	0.9	0.5	0.0773	0.0744	0.150	0.0186	0.0299	12
135	0.9	1.5	0.1613	0.129	0.383	0.0328	0.0844	15
135	0.9	4	0.218	0.220	0.901	0.0538	0.2086	15



A.4.3 Analytical Calculation

The MatLab script shown below is used to calculate the total heat loss and the distributions for the heat loss and temperature analytically for the wedge with conducting plates benchmark.

```
% This function calculates the heat loss and temperature distribution
    along
% the length of the plates of the wedge with conducting plates. The
    inputs:
% gammad:   Opening angle of the wedge [45-180^o]
% epsilon:  Emission coefficient of the plates [-]
% N_c:      Conduction parameter [-]
% mesh:     Number of points along the plate
function [thetax,qnor,Qnor] = wedge_fun(gammad,epsilon,N_c,mesh)
%% Solver parameters
countdiv = 4;                % Amount of iterations before taking
    the middle [-]
accuracy = 0.00001;         % Difference for which the while loop
    will end [-]
%% Create the plots for the N_c values

gamma = gammad/180*pi;      % Angle between the plates [rad]

zeta = 0:1/(mesh-1):1;      % Dimensionless coordinate along
    plate [-] x/L
thetaz = ones(1,length(zeta))*0.9; % Initial values for
    dimensionless temperature along plate to initiate the iteration
    [-]
difference = 1;             % Initial value for difference [-]
count = 1;                  % First count for iteration [-]
counttot = 0;               % Start of total counter (only for
    postprocessing to check if iteration converges) [-]
%% Calculation of the temperature distribution theta
while difference > accuracy % While loop to calculate
    dimensionless temperature distribution
    belowzero = 0;          % Initial state
    aboveone = 0;           % Initial state
    for i = 1:length(zeta) % Loop over the length of the plate
        xi = (i-1)/(length(zeta)-1); % Dimensionless coordinate
            over the plate (0-1)
        if xi == 0          % Prescribed temperature at xi = 0
            thetax(i) = 1;
        else                % This section calculates the values for K(J)
            which is used to calculate theta
            J(i-1,1) = 0;    % Prescribed because calculation
                results in NaN
            J(i-1,2:length(zeta)) = 0.5.*((xi.*(zeta(2:length(zeta))-
                cos(gamma)))./sqrt(1-2.*zeta(2:length(zeta)).*cos(
                gamma) + zeta(2:length(zeta)).^2) + cos(gamma)).*(sqrt(
                xi^2-2*xi.*zeta(2:length(zeta))*cos(gamma) + zeta(2:
                length(zeta)).^2) - zeta(2:length(zeta))) - zeta(2:
                length(zeta)).*sin(gamma)^2.*(log(xi-zeta(2:length(
                zeta)).*cos(gamma) + sqrt(xi^2 - 2*xi.*zeta(2:length(
                zeta)).*cos(gamma) + zeta(2:length(zeta)).^2)) - log(
                zeta(2:length(zeta)).*(1-cos(gamma))))); % Equation
                (17)
```

```

if gammad < 60
    J2(i-1,1) = 0; % Prescribed because calculation
    results in NaN
    J2(i-1,2:length(zeta)) = 0.5.*((xi.*(zeta(2:length(
        zeta))-cos(2*gamma))./sqrt(1-2.*zeta(2:length(
        zeta)).*cos(2*gamma) + zeta(2:length(zeta)).^2) +
        cos(2*gamma).*(sqrt(xi^2-2*xi.*zeta(2:length(zeta))
        )*cos(2*gamma) + zeta(2:length(zeta)).^2) - zeta
        (2:length(zeta))) - zeta(2:length(zeta)).*sin(2*
        gamma)^2.*(log(xi-zeta(2:length(zeta)).*cos(2*
        gamma) + sqrt(xi^2 - 2*xi.*zeta(2:length(zeta)).*
        cos(2*gamma) + zeta(2:length(zeta)).^2)) - log(
        zeta(2:length(zeta)).*(1-cos(2*gamma))))); %
    Equation (17)
    J3(i-1,1) = 0; % Prescribed because calculation
    results in NaN
    J3(i-1,2:length(zeta)) = 0.5.*((xi.*(zeta(2:length(
        zeta))-cos(3*gamma))./sqrt(1-2.*zeta(2:length(
        zeta)).*cos(3*gamma) + zeta(2:length(zeta)).^2) +
        cos(3*gamma).*(sqrt(xi^2-2*xi.*zeta(2:length(zeta))
        )*cos(3*gamma) + zeta(2:length(zeta)).^2) - zeta
        (2:length(zeta))) - zeta(2:length(zeta)).*sin(3*
        gamma)^2.*(log(xi-zeta(2:length(zeta)).*cos(3*
        gamma) + sqrt(xi^2 - 2*xi.*zeta(2:length(zeta)).*
        cos(3*gamma) + zeta(2:length(zeta)).^2)) - log(
        zeta(2:length(zeta)).*(1-cos(3*gamma))))); %
    Equation (17)
    K(i-1,:) = xi + epsilon.*(J(i-1,:) + (1-epsilon).*J2(
        i-1,:) + (1-epsilon)^2.*J3(i-1,:)); % Equation
    (16)
elseif gammad < 90
    J2(i-1,1) = 0; % Prescribed because calculation
    results in NaN
    J2(i-1,2:length(zeta)) = 0.5.*((xi.*(zeta(2:length(
        zeta))-cos(2*gamma))./sqrt(1-2.*zeta(2:length(
        zeta)).*cos(2*gamma) + zeta(2:length(zeta)).^2) +
        cos(2*gamma).*(sqrt(xi^2-2*xi.*zeta(2:length(zeta))
        )*cos(2*gamma) + zeta(2:length(zeta)).^2) - zeta
        (2:length(zeta))) - zeta(2:length(zeta)).*sin(2*
        gamma)^2.*(log(xi-zeta(2:length(zeta)).*cos(2*
        gamma) + sqrt(xi^2 - 2*xi.*zeta(2:length(zeta)).*
        cos(2*gamma) + zeta(2:length(zeta)).^2)) - log(
        zeta(2:length(zeta)).*(1-cos(2*gamma))))); %
    Equation (17)
    K(i-1,:) = xi + epsilon.*(J(i-1,:) + (1-epsilon).*J2(
        i-1,:)); % Equation (16)
elseif gammad < 180
    K(i-1,:) = xi + epsilon.*J(i-1,:); % Equation (16)
end
thetax(i) = 1 + epsilon*N_c*(trapz(zeta(1:i),thetaz(1:i)
.^4.*(xi-zeta(1:i))) - trapz(zeta,thetaz.^4.*K(i-1,:))
); % Dimensionless temperature distribution;
equation (15)
if thetax(i) < 0 % Changes the state in order to
make sure that the dimensionless temperature is

```



```

        between 0 and 1 and does not converge to values which
        are impossible
        belowzero = 1;
    elseif thetax(i) > 1
        aboveone = 1;
    end
end
end
if belowzero == 1 % Puts the value of theta between
    0 and 1, a value below 0 or above 1 is physically impossible
    and the model could converge to a physically infeasible
    solution.
    thetax = (thetax + 1)./2;
elseif aboveone == 1
    thetax = 1 - (thetax - 1);
end
count = count + 1; % Counter for converging
counttot = counttot + 1 % Total counter
difference = max(abs(thetax - thetax)); % Maximal difference
    between the current and previous dimensionless temperature
    distribution
if count == countdiv % After countdiv iterations theta is
    chosen in the middle of the previous two distributions for
    faster convergence
    thetax = (thetax+thetax)./2;
    count = 1; % Reset counter
else
    thetax = thetax; % Current distribution is kept for
        calculating difference in next iteration
end
end
%% Calculation of the normalized heat flux qnor
for i = 1:length(zeta) % For-loop which loops over
    the length of the plate
    xi = (i-1)/(length(zeta)-1); % Dimensionless coordinate of
        the plate x/L
    if xi == 0 % At xi = 0 the equation (21)
        is used
        if gammad < 60
            L = 0.5*((1 + cos(gamma)) + (1-epsilon)*(1+cos(2*gamma))
                + (1-epsilon)^2*(1+cos(3*gamma)));
        elseif gammad < 90
            L = 0.5*((1 + cos(gamma)) + (1-epsilon)*(1+cos(2*gamma)))
                ;
        elseif gammad < 180
            L = 0.5*(1+cos(gamma));
        end
        qnor(i) = 1-epsilon*L; % Normalized heat flux at xi
            = 0; equation (20)
    else
        if gammad < 60
            f(i,:) = sin(gamma)^2/2*(xi.*zeta)./(xi^2 + zeta.^2 - 2*
                xi.*zeta.*cos(gamma)).^1.5; % Equation (13)
            f2(i,:) = sin(2*gamma)^2/2*(xi.*zeta)./(xi^2 + zeta.^2 -
                2*xi.*zeta.*cos(2*gamma)).^1.5; % Equation (13)
        end
    end
end

```

```

f3(i,:) = sin(3*gamma)^2/2*(xi.*zeta)./(xi^2 + zeta.^2 -
2*xi.*zeta.*cos(3*gamma)).^1.5; % Equation (13)
G(i,:) = f(i,:) + (1-epsilon).*f2(i,:) + (1-epsilon)^2.*
f3(i,:); % Equation (12)
P = 0.5*(1 + (cos(gamma) - zeta)./(1 + zeta.^2 - 2.*zeta
.*cos(gamma)).^0.5); % Equation (26)
P2 = 0.5*(1 + (cos(2*gamma) - zeta)./(1 + zeta.^2 - 2.*
zeta.*cos(2*gamma)).^0.5); % Equation (26)
P3 = 0.5*(1 + (cos(3*gamma) - zeta)./(1 + zeta.^2 - 2.*
zeta.*cos(3*gamma)).^0.5); % Equation (26)
M = 1 - epsilon.*(P + (1-epsilon).*P2 + (1-epsilon)^2.*P3
); % Equation (25)
elseif gammad < 90
f(i,:) = sin(gamma)^2/2*(xi.*zeta)./(xi^2 + zeta.^2 - 2*
xi.*zeta.*cos(gamma)).^1.5; % Equation (13)
f2(i,:) = sin(2*gamma)^2/2*(xi.*zeta)./(xi^2 + zeta.^2 -
2*xi.*zeta.*cos(2*gamma)).^1.5; % Equation (13)
G(i,:) = f(i,:) + (1-epsilon).*f2(i,:); %
Equation (12)
P = 0.5*(1 + (cos(gamma) - zeta)./(1 + zeta.^2 - 2.*zeta
.*cos(gamma)).^0.5); % Equation (26)
P2 = 0.5*(1 + (cos(2*gamma) - zeta)./(1 + zeta.^2 - 2.*
zeta.*cos(2*gamma)).^0.5); % Equation (26)
M = 1 - epsilon.*(P + (1-epsilon).*P2); %
Equation (25)
elseif gammad < 180
f(i,:) = sin(gamma)^2/2*(xi.*zeta)./(xi^2 + zeta.^2 - 2*
xi.*zeta.*cos(gamma)).^1.5; % Equation (13)
G(i,:) = f(i,:);
% Equation (12)
P = 0.5*(1 + (cos(gamma) - zeta)./(1 + zeta.^2 - 2.*zeta
.*cos(gamma)).^0.5); % Equation (26)
M = 1 - epsilon.*P;
% Equation (25)
end
qnor(i) = thetax(i)^4 - epsilon*trapz(zeta,thetax.^4.*G(i,:))
; % Normalized heat flux; Equation (19)
Qnor = epsilon*trapz(zeta,thetax.^4.*M);
% Normalized total heat flux;
Equation (24)
end
end
end

```

A.4.4 Numerical Calculation

The numerical calculation of the wedge with conducting plates benchmark is performed using the *Surface-to-Surface Radiation* and the *Heat transfer in Solids* modules in COMSOL. The figures in this section describe how the model is developed.

Name	Expression	Value	Description
W	1[m]	1 m	Width of the plates
t	0.05[m]	0.05 m	Thickness of the plates
gamma	60[deg]	1.0472 rad	Wedge opening
N_c	4	4	Conduction parameter
k_mat	250[W/(m*K)]	250 W/(m-K)	Thermal conductivity
T_b	$(k_mat * t * N_c / (W^2 * \sigma))^{1/3}$	958.93 K	Base temperature
epsilon	0.9	0.9	Emission coefficient
M	400	400	Number of elements per plate

Figure A.38: The parameter list used to describe the geometry and the physics

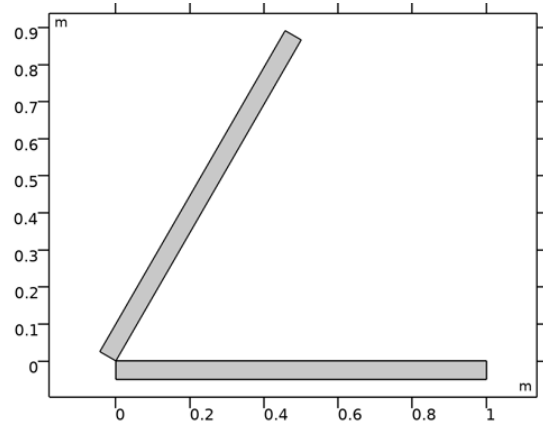


Figure A.39: The geometry of the model

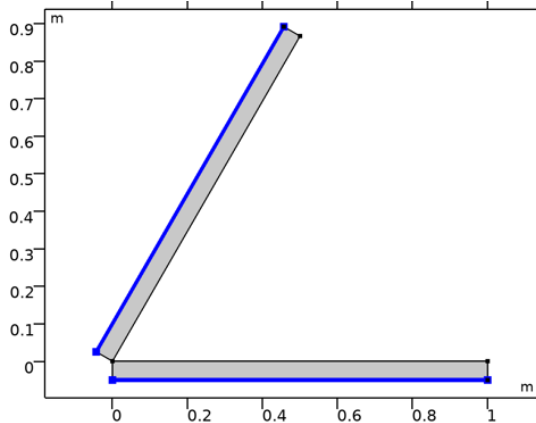


Figure A.40: The adiabatic boundary condition for the Heat Transfer in Solids module, prescribing a heat flux of 0W through these surfaces.

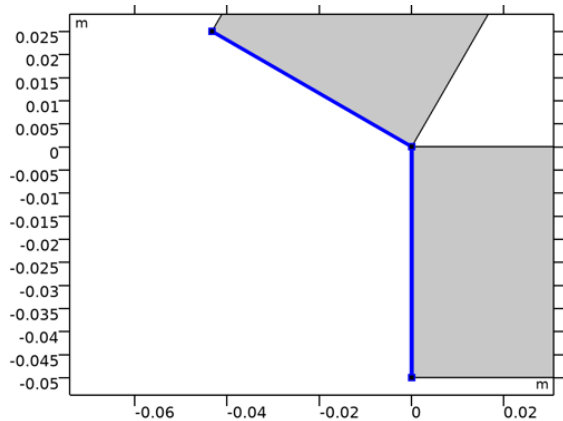


Figure A.41: The temperature boundary condition for the Heat Transfer in Solids module, prescribing the temperature of T_b of these surfaces. This temperature is defined in the parameter list as $T_b = \left(\frac{\kappa t N_c}{L \sigma}\right)^{1/3}$

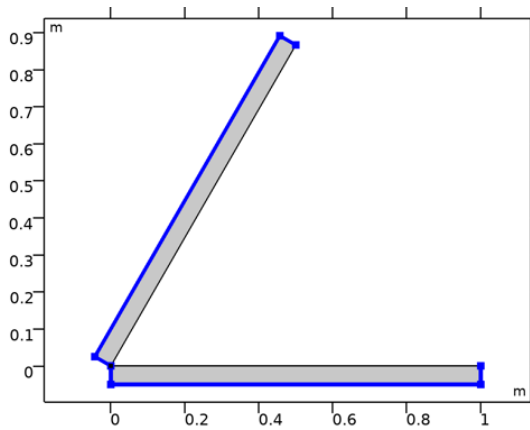


Figure A.42: Boundary condition for the Surface-to-Surface Radiation module prescribing radiation emitted or received on these surfaces.

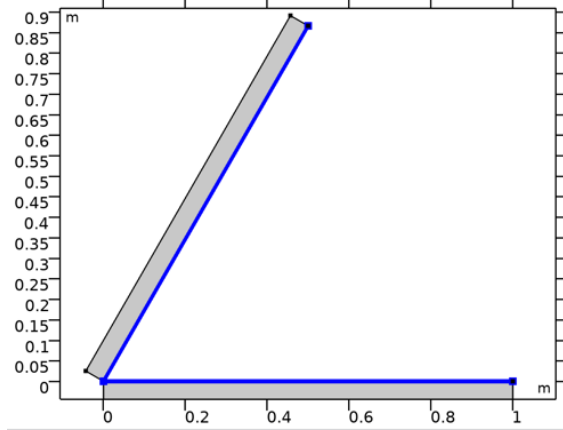


Figure A.43: Boundary condition for the Surface-to-Surface Radiation module describing opaques surfaces with specular reflection.

Description	Value
Use radiation groups	Off
Wavelength dependence of surface properties	Constant
Surface-to-surface radiation method	Ray shooting
Radiation resolution	512
Maximum number of adaptations	5
Transparent media refractive index	1
Tolerance	1e-5

Figure A.44: Radiation Settings

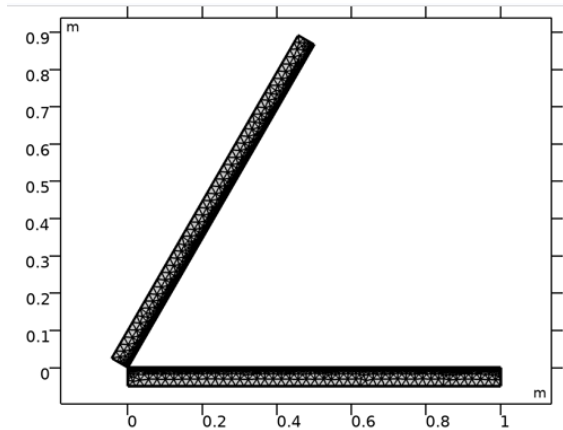


Figure A.45: The mesh of the geometry

Description	Value
Minimum element quality	1.0
Average element quality	1.0
Edge element	1626
Vertex element	4

Figure A.46: Mesh settings

B Square Passage including Angle Dependent Reflection and Polarization

This appendix contains details on the square passage benchmark. Section B.1 describes several studies conducted on the analytical and numerical models. The fact that the results for both models converge is shown in this section. In section B.2 the MatLab model for the analytical approximation is shown. The way the numerical model is constructed is displayed in section B.3.

B.1 Refinement studies

Analytical approximation convergence

In order to derive the converged value of the analytical approximation function of the square passage benchmark, the number of reflections and number of subdivisions was increased. To prove that the function converges, three contour plots are shown in figure B.1, where the influence of increasing the reflections and subdivisions is shown.

The value for the transmission at 10 subdivisions and 10 reflections is taken as the reference: $\tau = 1$. From the contour plots can be derived that the approximation becomes more accurate when increasing both parameters. Which of the two parameters is restricting the convergence depends on the length to width ratio. For a short passage, the number of subdivisions is restricting, where the number of reflections is the restricting parameter for long passages.

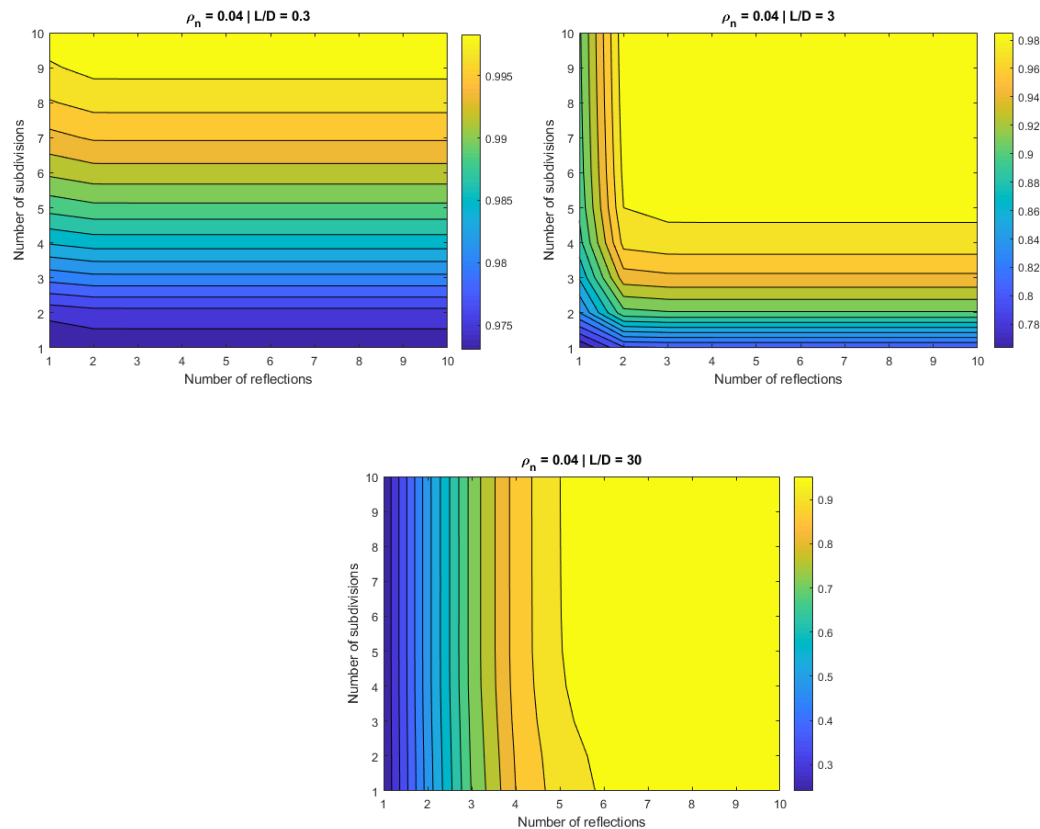


Figure B.1: These three contour plots show the influence of increasing the number of reflections and subdivisions on the results of the analytical approximation. The different plots represent different length to width ratios. When increasing both the number of reflections and subdivisions, the result will converge. However the length to width ratio strongly influences which parameter is the restricting factor. For short passages the number of subdivisions is of a larger influence and for long passages the number of reflections is the main restriction.

Numerical Model

The quality of the numerical result is strongly dependent on the settings used. Three settings are varied for the numerical model to evaluate the influence on the results: (1) the ratio of the number of bundles and the amount of rays per bundle, (2) the mesh and (3) the number of rays released at the inlet.

In the ray tracing model it is possible to influence in what way the rays are released. On the surface a number of ray bundles (N) is released. Each bundle contains a number of rays (N_w) and the direction of these rays is randomly distributed.

The influence of the ratio between the amount of bundles and the amount of rays per bundle on the transmission is evaluated by calculating the relative error between the analytical and numerical result for the transmission. This is shown in figure B.2 for a normal reflection coefficient $\rho_n = 0.8$.

The graph on the left shows the relative error for each length over width ratio. The error is higher for larger lengths of the passage. In order to determine which N/N_w -ratio is most favorable the minimal error for each L/D -ratio is evaluated and subtracted from the relative error. The result is shown in the graph on the right.

For every L/D -ratio an N/N_w -ratio of 1.5 results in the smallest error. This ratio is used for the rest of the models.

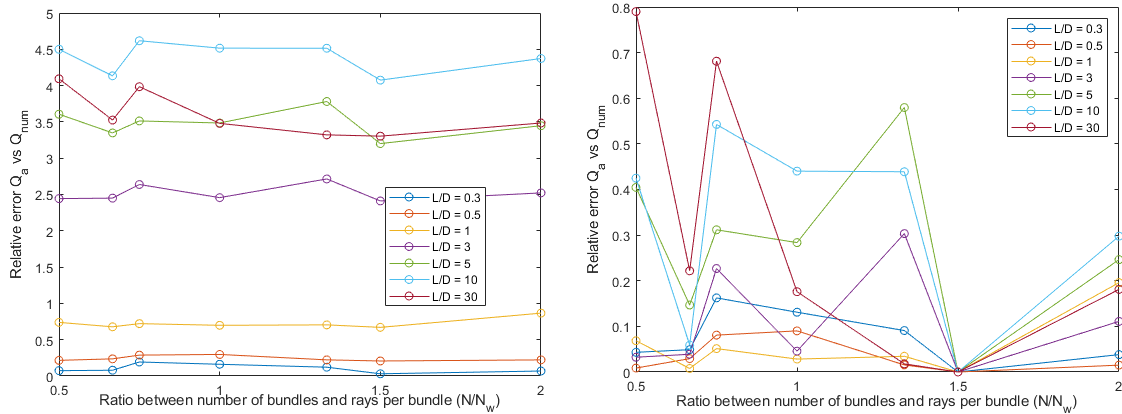


Figure B.2: These graphs show the influence of the ratio between the amount of bundles released (N) and the amount of rays per bundle (N_w), by evaluating the relative error between the numerical and analytical results for $\rho_n = 0.8$. The graph on the right is obtained by subtracting the minimal error for every L/D -ratio from the error in the graph on the left. For $N/N_w = 1.5$ the error is the smallest for all L/D -ratios.

The second model parameter that has been evaluated is the amount of mesh elements. In general it holds for Finite Element Models that the result is more accurate for a higher number of elements. However this model is an exception.

The amount of energy lost to the side walls depends on the angle of incidence and the polarization state of the radiation. These are determined by calculating the dot product between the ray direction vector and the normal vector of the wall. Since the walls are flat, the normal vector does not become more accurate for a finer mesh, this would be the case for curved walls.

Figure B.3 shows the conducted study, where the influence of the mesh size is evaluated for $\rho_n = 0.8$ and different length to width ratios. The mean value for each length to width ratio is evaluated and subtracted from each evaluation (equation B.1). This results in a certain deviation of the transmission for different mesh sizes.

From figure B.3 can be derived that the influence of different mesh sizes is less than 0.5%.

$$Dev(L/D, ref)[\%] = \frac{\tau(L/D, ref) - \tau_{avg}(L/D)}{\tau_{avg}(L/D)} \cdot 100[\%] \quad (\text{B.1})$$

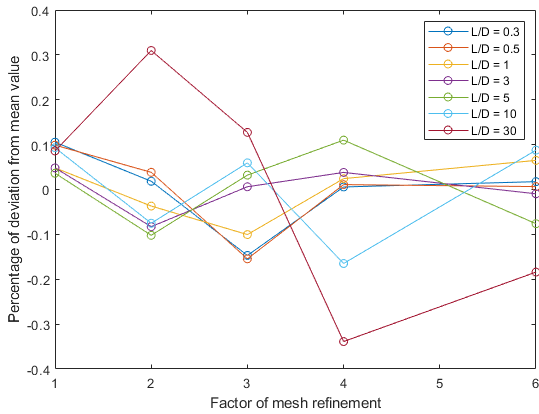


Figure B.3: This graph shows the percentage of deviation of the transmission through the square passage for changing the mesh size for $\rho_n = 0.8$ and various L/D -ratios. The deviation is less than 0.5%, from this can be concluded that the mesh has a very small influence on the transmission.

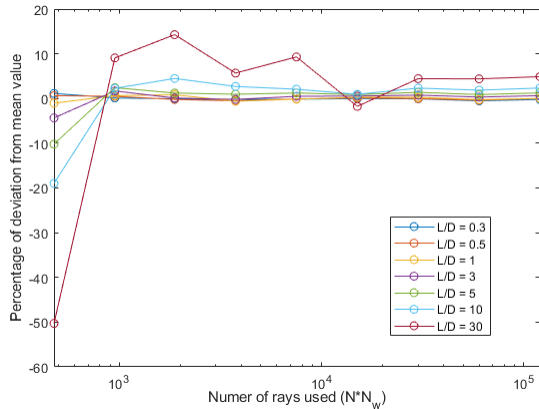


Figure B.4: This graph shows the percentage of deviation of the transmission through the square passage for increasing the number of released rays for $\rho_n = 0.8$ and various L/D -ratios. The resulting transmission converges to a value when increasing the number of rays.

The third model parameter that has been evaluated is the amount of rays released at the inlet ($N \cdot N_w$). Increasing the amount of rays released will increase the accuracy of the calculated transmission. This is shown in figure B.4 where the deviation calculated by equation B.1 is plotted for increasing number of rays. The normal reflection coefficient is equal to $\rho_n = 0.8$ and various length to width ratios are evaluated. The resulting transmission stabilizes when increasing the number of rays, which means that the result converges.

B.2 Analytical Approximation

The function in the following script calculates the transmission through a square passage. The type of reflection taken into account can be changed by choosing which expression for ρ_{Dmat} is used.

```
% This function evaluates the transmission through the square passage
. The
% inputs are:
% nreal:    Refractive index of the wall [-]
% k:       Absorptive index of the wall [-]
% x:       Length over width ratio of the passage [-]
% maxN:    Maximum number of reflection taken into account [-]
% Idisc:   Number of subdivisions of the end surfaces for the first
% reflection = Idisc^2 [-]
function [Qtot] = Polarization_fun(nreal,k,x,maxN,Idisc)
%% Input variables
D = 0.2;                                     % Width of the
    passage [m]
rhon = ((nreal-1)^2+k^2)/((nreal+1)^2+k^2);   % Normal reflectivity
    of the wall [-]
L = x*D;                                     % Length of the
    passage [m]
%% Define end plane and number of reflections
surf = 0;                                    % Initial value
QF = 0;                                       % Initial value
QKer = 0;                                     % Initial value
for N = 0:maxN                                % Loop over the number of reflections
for M = 0:N                                    % Loop over the end planes per reflection
    J = N-M;                                   % Determine plane J for chosen plane M
    %% Discretization grid
    if N == 0 || N == 1                       % Create the grid for 0 and 1 reflections
        Im1 = Idisc;
        Ij1 = Idisc;
        Im2 = Idisc;
        Ij2 = Idisc;
    else
        Im2 = ceil(2*Idisc/N); % Function defined to decrease the
            number of subdivisions for a higher number of reflections
        Ij2 = Im2;
        Im1 = ceil(2*Idisc/N);
        Ij1 = Im1;
    end
    %% Assign begin and end surf aces
    for j1 = 1:Ij1                            % Loop over all subdivisions of the end plane
    for m1 = 1:Im1
    for j2 = 1:Ij2
    for m2 = 1:Im2
        surf = surf + 1; % Count surface subdivision
    %% Kernel calculation
    x1 = [0;D/Ij1*j1-D/(Ij1*2);D/Im1*m1-D/(Im1*2)]; % Starting
        coordinate of ray vector
    [x2,Fxi1,Fxi2,Feta1,Feta2] = x2_fun(D,L,M,J,Im2,Ij2,m2,j2); % End
        coordinate and help variables for ray vector
    A2 = (Fxi2-Fxi1)*(Feta2-Feta1); % Surface area of end
        subdivision
    r = x2 - x1; % Ray direction vector
```



```

    rlength = norm(r); % Length traveled by ray
    theta_release = acos(dot(r/norm(r), [1;0;0])); % Angle of
        incidence
    Ker = cos(theta_release)*cos(theta_release)/(pi*rlength^2)*A2; %
        Radiation Kernel
    %% Heat flux calculation
    E = [100/(Im1*Ij1) 100/(Im1*Ij1) 0 0]; % Energy matrix
    I = [1/2;1/2;0;0]; % Intensity matrix
    % rhoDmat = diag([1 1 1 1])*rhon^N; % Constant
    reflectivity
    % rhoDmat = rho_adeq(nreal,k,D,N,x1,r); % Only angle
    dependent reflection - no polarization
    rhoDmat = rhoD_matrix(nreal,k,D,N,x1,r); % Multiplication of
        the rotation and reflection matrix
    dQKer(surf) = E*rhoDmat*I*Ker; % Heat flux from one
        subdivision of the start plane to one subdivision of the end
        plane
    QKer = QKer + dQKer(surf); % Summation for all
        surface combinations
    end
    end
    end
    end
end
end
Qtot = real(QKer*4); % Heat flux multiplied by 4 due to
    symmetry

```

This function is used in the *Polarization_fun* to determine the coordinates of the end point of the ray vector for a given subdivision of the end plane.

```

% This function derives the coordinate of the end point of the ray
    vector
% on the end plane. It also derives help variables to calculate the
    surface
% area of the subdivision
function [x2,Fxi1,Fxi2,Feta1,Feta2] = x2_fun(D,L,M,J,Im2,Ij2,m2,j2)
    if J == 0 % Due to the smaller end planes at 0 reflections a
        slightly different calculation is required
        x2_2 = D/2 + D/(Ij2*2)*j2 - D/(Ij2*4); % y-component of the
            end point of the ray vector
        Fxi1 = x2_2 - D/(Ij2*4); % Help variable for area
            calculation
        Fxi2 = x2_2 + D/(Ij2*4);
    else
        x2_2 = D*J + D/Ij2*j2 - D/(Ij2*2);
        Fxi1 = x2_2 - D/(Ij2*2);
        Fxi2 = x2_2 + D/(Ij2*2);
    end
    if M == 0 % Due to the smaller end planes at 0 reflections a
        slightly different calculation is required
        x2_3 = D/2 + D/(Im2*2)*m2 - D/(Im2*4); % z-component of the
            end point of the ray vector
        Feta1 = x2_3 - D/(Im2*4);
        Feta2 = x2_3 + D/(Im2*4);
    else
        x2_3 = D*M + D/Im2*m2 - D/(Im2*2);
    end

```

```
Feta1 = x2_3 - D/(Im2*2);
Feta2 = x2_3 + D/(Im2*2);
end
x2 = [L;x2_2;x2_3];      % Coordinates of the end of the ray
vector
```

This function is used in the *Polarization_fun* to determine the reflection and rotation matrix for a ray vector. This includes the influence of angle-dependency, polarization and the rotation of the polarization planes.

```
%% Description
% This function derives two 4x4 matrices: the rho-matrix and the D-
matrix.
% The result of this function is the multiplication of these two
matrices.
% And is used in the Polarization_fun. The rho-matrix describes the
% reflectivity at a certain angle of incidence for polarized
radiation. The
% D-matrix describes the rotation of the polarization planes after
one
% reflection.
%% Input
% nreal - the refractive index (>0) of the material of the duct
% k - the extinction coefficient (>=0) of the material of the duct
% D - the width and height of the square duct
% N - number of reflections with the duct wall
% x1 - center of the surface at the entrance of the duct
% r - vector from the entrance to the end of the duct

function [rhoDmat] = rhoD_matrix(nreal,k,D,N,x1,r)
Dmat(:,:,1) = diag([1 1 1 1]);      % Initial rotation matrix (
unity)
miter = 1;                          % Initial iteration for left/
right (set to 1)
jiter = 1;                          % Initial iteration for top/
bottom (set to 1)
rn_loc(:,1) = r/norm(r);            % Initial normalized local
direction vector
if N == 0                            % No reflections returns the
unity matrix
rhoDmat = Dmat(:,:,1);
else                                  % If there are reflections the
calculation loops over all reflections
for i = 1:N
Ay(i) = abs((jiter*D - x1(2))/r(2)); % Multiplication
constant for the relative distance towards the side of the
duct
Az(i) = abs((miter*D - x1(3))/r(3)); % Multiplication
constant for the relative distance towards the top/bottom
of the duct
if Ay(i) < Az(i)                    % The smallest multiplication
constant is chosen since that is where the ray will bounce
first left/right or top/bottom
A(i) = Ay(i);                      % Multiplication constant
left/right
if mod(jiter,2) == 0                % If the iteration is even
the ray bounces with the left wall and otherwise with
```

```

        the right wall
        n(:,i) = [0;1;0];           % Normal vector
    else
        n(:,i) = [0;-1;0];        % Normal vector
    end
    jiter = jiter+1;              % Add 1 to change direction
    of next bounce
else
    A(i) = Az(i);                 % Multiplication constant top
    /bottom
    if mod(miter,2) == 0          % If the iteration is even
        the ray bounces with the bottom wall and otherwise
        with the top wall
        n(:,i) = [0;0;1];        % Normal vector
    else
        n(:,i) = [0;0;-1];      % Normal vector
    end
    miter = miter+1;             % Add 1 to change direction
    of next bounce
end
s_in(:,i) = cross(rn_loc(:,i),n(:,i))./norm(cross(rn_loc(:,i)
, n(:,i))); % Unit vector perpendicular to plane of
incidence for incoming ray
p_in(:,i) = cross(s_in(:,i),rn_loc(:,i));
% Unit vector parallel
to plane of incidence and perpendicular to ray for
incoming ray
thetai(i) = acos(abs(dot(n(:,i),rn_loc(:,i))));
% Angle of incidence
rn_loc(:,i+1) = -dot(rn_loc(:,i),n(:,i)).*n(:,i) + dot(rn_loc
(:,i),cross(n(:,i),s_in(:,i))).*cross(n(:,i),s_in(:,i));
% Local ray direction vector
s_em(:,i) = s_in(:,i);
%
Unit vector perpendicular to plane of incidence for
emitted ray
p_em(:,i) = cross(s_em(:,i),rn_loc(:,i+1));
% Unit vector parallel to
plane of incidence and perpendicular to ray for emitted
ray
a2 = 0.5*(sqrt((nreal^2-k^2 - sin(thetai(i)).^2).^2 + 4*nreal
^2*k^2) + (nreal^2-k^2)-sin(thetai(i)).^2); % Help
variable for rho calculation
b2 = 0.5*(sqrt((nreal^2-k^2 - sin(thetai(i)).^2).^2 + 4*nreal
^2*k^2) - (nreal^2-k^2)+sin(thetai(i)).^2); % Help
variable for rho calculation
a = sqrt(a2);
% Help variable for rho calculation
b = sqrt(b2);
% Help variable for rho calculation
deltap(i) = atan((2*cos(thetai(i))*((nreal^2-k^2)*b - (2*
nreal*k)*a))/((nreal^2+k^2)^2*cos(thetai(i))^2 - (a^2+b^2)
)); % Phase change for p-polarized radiation

```

```

deltas(i) = atan(2*b*cos(thetai(i))/(cos(thetai(i)).^2 - a^2
- b^2));
                                                    % Phase
    change for s-polarized radiation
delta(i) = 2*pi + (deltap(i) - deltas(i));

    % Difference in phase change upon reflection
rhoss(i) = ((cos(thetai(i)) - a).^2+b.^2)./((cos(thetai(i)) +
a).^2+b.^2); % Perpendicular component of reflectivity
matrix
rhopp(i) = (((nreal^2-k^2).*cos(thetai(i))-a).^2 + (2*nreal*k
.*cos(thetai(i)) - b).^2)./(((nreal^2-k^2).*cos(thetai(i))
+a).^2 + (2*nreal*k.*cos(thetai(i)) + b).^2); %
Parallel component of reflectivity matrix
rhovu(i) = sqrt(rhoss(i)*rhopp(i))*cos(delta(i));
                                                    % Matrix element reflectivity
matrix
rhovv(i) = rhovu(i);
                                                    %
Matrix element reflectivity matrix
rhovu(i) = sqrt(rhoss(i)*rhopp(i))*sin(delta(i));
                                                    % Matrix element reflectivity
matrix
rhovu(i) = -rhovu(i);
                                                    %
Matrix element reflectivity matrix
rhomat(:, :, i) = [rhoss(i) 0 0 0; 0 rhopp(i) 0 0; 0 0 rhovu(i)
rhovv(i)]; % Reflectivity
matrix
if i == 1 % If this is
the first reflection rhoDmat does not exist yet so needs
to be created from rhomat and Dmat
rhoDmat(:, :, i) = rhomat(:, :, i)*Dmat(:, :, i);
else
beta_poss = [acos(dot(p_em(:, i-1), p_in(:, i))), 2*pi-acos(
dot(p_em(:, i-1), p_in(:, i))), asin(-dot(p_em(:, i-1),
s_in(:, i))), asin(-dot(p_em(:, i-1), s_in(:, i)))+pi];
% Calculate beta with two equations which give 4
values
beta_diff = [beta_poss(1) - beta_poss(3), beta_poss(2) -
beta_poss(4), beta_poss(3) - beta_poss(2), beta_poss(4)
- beta_poss(1)]; % A vector with the difference
between the four values is created
for search = 1:length(beta_diff) % By looking
which two of the four values are equal the value of
beta is determined
if beta_diff(search) < 1e-10
beta(i) = beta_poss(search);
end
end
Dmat(:, :, i) = [cos(beta(i))^2 sin(beta(i))^2 0.5*sin(2*
beta(i)) 0; sin(beta(i))^2 cos(beta(i))^2 -0.5*sin(2*
beta(i)) 0; -sin(2*beta(i)) sin(2*beta(i)) cos(2*beta(
i)) 0; 0 0 0 1]; % Rotation matrix
rhoDmat(:, :, i) = rhomat(:, :, i)*Dmat(:, :, i)*rhoDmat(:, :, i)

```

```

        -1);          % rho-matrix multiplied by the rotation
                    matrix and then multiplied by the previous rhoDmatrix
    end
end
rhoDmat = rhoDmat(:,:,N);          % Resulting matrix is the
rhoDmatrix after the last reflection.
end

```

This function is used in the *Polarization_fun* to determine the reflection when only angle dependent reflection is taken into account.

```

%% Description
% This function derives one 4x4 matrix: the rho-matrix. The matrix is
  used
% in the Polarization_fun. The rho-matrix describes the reflectivity
  at a
% certain angle of incidence without taking polarization state into
  account
%% Input
% nreal - the refractive index (>0) of the material of the duct
% k - the extinction coefficient (>=0) of the material of the duct
% D - the width and height of the square duct
% N - number of reflections with the duct wall
% x1 - center of the surface at the entrance of the duct
% r - vector from the entrance to the end of the duct
function [rho_adep,thetai,n] = rho_adep(nreal,k,D,N,x1,r)
miter = 1;          % Initial iteration for left/
  right (set to 1)
jiter = 1;          % Initial iteration for top/
  bottom (set to 1)
rn_loc(:,1) = r/norm(r);          % Initial normalized local
  direction vector
if N == 0          % No reflections returns the
  unity matrix
  rho_adep = 1;
else          % If there are reflections the
  calculation loops over all reflections
  for i = 1:N
    Ay(i) = abs((miter*D - x1(2))/r(2)); % Multiplication
      constant for the relative distance towards the side of the
      duct
    Az(i) = abs((jiter*D - x1(3))/r(3)); % Multiplication
      constant for the relative distance towards the top/bottom
      of the duct
    if Ay(i) < Az(i)          % The smallest multiplication
      constant is chosen since that is where the ray will bounce
      first left/right or top/bottom
      A(i) = Ay(i);          % Multiplication constant
      left/right
      if mod(miter,2) == 0          % If the iteration is even
        the ray bounces with the left wall and otherwise with
        the right wall
        n(:,i) = [0;1;0];          % Normal vector
      else
        n(:,i) = [0;-1;0];          % Normal vector
      end
      miter = miter+1;          % Add 1 to change direction
    end
  end
end

```

```

        of next bounce
    else
        A(i) = Az(i);           % Multiplication constant top
        /bottom
        if mod(jiter,2) == 0    % If the iteration is even
            the ray bounces with the bottom wall and otherwise
            with the top wall
            n(:,i) = [0;0;1];   % Normal vector
        else
            n(:,i) = [0;0;-1]; % Normal vector
        end
        jiter = jiter+1;       % Add 1 to change direction
        of next bounce
    end
    s_in(:,i) = cross(rn_loc(:,i),n(:,i))./norm(cross(rn_loc(:,i)
        ,n(:,i)));           % Unit vector perpendicular to plane of
        incidence for incoming ray
    thetai(i) = acos(abs(dot(n(:,i),rn_loc(:,i)))) ;
        % Angle of incidence
    rn_loc(:,i+1) = -dot(rn_loc(:,i),n(:,i)).*n(:,i) + dot(rn_loc
        (:,i),cross(n(:,i),s_in(:,i))).*cross(n(:,i),s_in(:,i));
        % Local ray direction vector
    a2 = 0.5*(sqrt((nreal^2-k^2 - sin(thetai(i)).^2).^2 + 4*nreal
        ^2*k^2) + (nreal^2-k^2)-sin(thetai(i)).^2); % Help
        variable for rho calculation
    b2 = 0.5*(sqrt((nreal^2-k^2 - sin(thetai(i)).^2).^2 + 4*nreal
        ^2*k^2) - (nreal^2-k^2)+sin(thetai(i)).^2); % Help
        variable for rho calculation
    a = sqrt(a2);

        % Help variable for rho calculation
    b = sqrt(b2);

        % Help variable for rho calculation
    rhoss(i) = ((cos(thetai(i)) - a).^2+b.^2)./((cos(thetai(i)) +
        a).^2+b.^2); % Perpendicular component of reflectivity
        matrix
    rhopp(i) = (((nreal^2-k^2).*cos(thetai(i))-a).^2 + (2*nreal*k
        .*cos(thetai(i)) - b).^2)./(((nreal^2-k^2).*cos(thetai(i))
        +a).^2 + (2*nreal*k.*cos(thetai(i)) + b).^2); %
        Parallel component of reflectivity matrix
    if i == 1 % If this is
        the first reflection rhoDmat does not exist yet so needs
        to be created from rhomat and Dmat
        rho_adeq(i) = (rhoss(i)+rhopp(i))/2;
    else
        rho_adeq(i) = (rhoss(i)+rhopp(i))/2*rho_adeq(i-1);
        % rho-matrix multiplied by the rotation matrix
        and then multiplied by the previous rhoDmatrix
    end
end
rho_adeq = rho_adeq(N); % Resulting matrix is the
rhoDmatrix after the last reflection.
end

```

This function is used to evaluate *Polarization_fun* by gradually increasing the number of reflections and subdivisions. The accuracy of the converged answer is given as an input. The amount of reflections and subdivisions are increased separately, to determine if the result still changes for one of the two.

```
%% Description
% This function helps in evaluating the Polarization_fun, by
  automatically
% choosing the appropriate values for the maximum number of
  reflections (N)
% and the discretization to evaluate, to achieve a prescribed
  accuracy of
% the solution.
%% Output
% Qfin - The final value for the transmission through the tube
% dQ - The resulting relative accuracy [%]
% N - The amount of reflections that are taken into account
% Idisc - The amount of discretizations the model undergoes
%% Input
% nreal - the refractive index (>0) of the material of the duct
% k - the extinction coefficient (>=0) of the material of the duct
% x - The relative length of the duct: x = L/D
% acc - The requested relative accuracy for the solution

function [Qfin,dQ,N,Idisc,Q] = Eval_Polfun(nreal,k,x,acc)

maxcount = 30; % The maximum
  number of evaluations - to prevent the script from running forever

N(1) = 3; % Initial
  number of reflections taken into account
Idisc(1) = 3; % Initial discretizations the model undergoes

%% The function is evaluated 3 times to have initial values of the
  result
count = 1 % Initiate
  counter
Q(count) = Polarization_fun(nreal,k,x,N(count),Idisc(count)); %
  Transmission for initial values
count = count + 1 % Counter
N(count) = N(count-1) + 1; %
  Increase the number of reflections by 1
Idisc(count) = Idisc(count-1); %
  These discretizations does not change
Q(count) = Polarization_fun(nreal,k,x,N(count),Idisc(count)); %
  Transmission
DQN = abs(Q(count)-Q(count-1)); % Relative difference for
  increasing the number of reflections by 1
count = count + 1
N(count) = N(count-1); % The
  number of reflections does not change
Idisc(count) = Idisc(count-1) + 1; %
  Increase the discretizations by 1
Q(count) = Polarization_fun(nreal,k,x,N(count),Idisc(count)); %
  Transmission
DQI = abs(Q(count)-Q(count-1)); % Relative difference for
```

```
    increasing the discretizations by 1

%% If the difference for both the reflections and discretizations is
    above acc both N and Idisc are increased
while DQN > acc && DQI > acc && count < maxcount
    count = count + 1
    N(count) = N(count-1) + 1;
    Idisc(count) = Idisc(count-1);
    Q(count) = Polarization_fun(nreal,k,x,N(count),Idisc(count));
    DQN = abs(Q(count)-Q(count-1));
    count = count + 1
    N(count) = N(count-1);
    Idisc(count) = Idisc(count-1) + 1;
    Q(count) = Polarization_fun(nreal,k,x,N(count),Idisc(count));
    DQI = abs(Q(count)-Q(count-1));
end

%% If the difference for the discretizations is met only the number
    of reflections is increased
while DQN > acc && count < maxcount
    count = count + 1
    N(count) = N(count-1) + 1;
    Idisc(count) = Idisc(count-1);
    Q(count) = Polarization_fun(nreal,k,x,N(count),Idisc(count));
    DQN = abs(Q(count)-Q(count-1));
end

%% If the difference for the number of reflections is met only the
    discretizations is increased
while DQI > acc && count < maxcount
    count = count + 1
    N(count) = N(count-1);
    Idisc(count) = Idisc(count-1) + 1;
    Q(count) = Polarization_fun(nreal,k,x,N(count),Idisc(count));
    DQI = abs(Q(count)-Q(count-1));
end

%% If both conditions are met the function returns the last evaluated
    values of Q and DQ
N = N(count);
Idisc = Idisc(count);
Qfin = Q(count);
dQ = max(DQI,DQN);
```


B.3 Numerical Model

The numerical calculation of the square passage benchmark is performed using the *Ray Optics* module in COMSOL. The figures in this section describe how the model is developed.

Name	Expression	Value	Description
D	20[cm]	0.2 m	Width of the passage
L	D*30	6 m	Length of the passage
Tb	300[K]	300 K	Temperature at the entrance
nreal	1.5	1.5	Refractive index of the walls
kimag	0	0	Absorptive index of the walls
N	200	200	Number of bundles released
Nw	300	300	Number of rays per bundle

Figure B.5: The parameter list used to describe the geometry and the physics

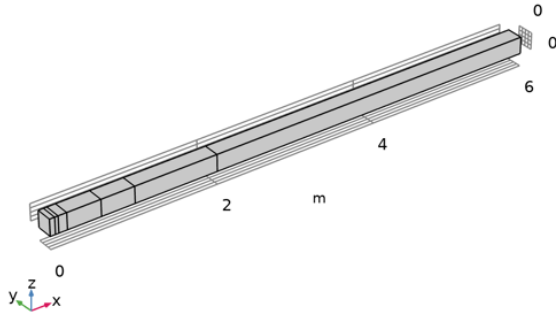


Figure B.6: Geometry of the model

Name	Expression	Unit	Description
n	nreal + kimag*i		Complex refractive index
theta_t	asin(1/n*sin(gop.wall1.thetai))	rad	Angle of transmittance
r_s	(cos(gop.wall1.thetai) - n*cos(theta_t))/(cos(gop.wall1.thetai) + n*cos(theta_t))		Reflection coefficient perpendicular polarized radiation
r_p	(n*cos(gop.wall1.thetai) - cos(theta_t))/(n*cos(gop.wall1.thetai) + cos(theta_t))		Reflection coefficient parallel polarized radiation

Figure B.7: Definitions used to define the reflection coefficients of the side walls of the geometry using the Fresnel equations

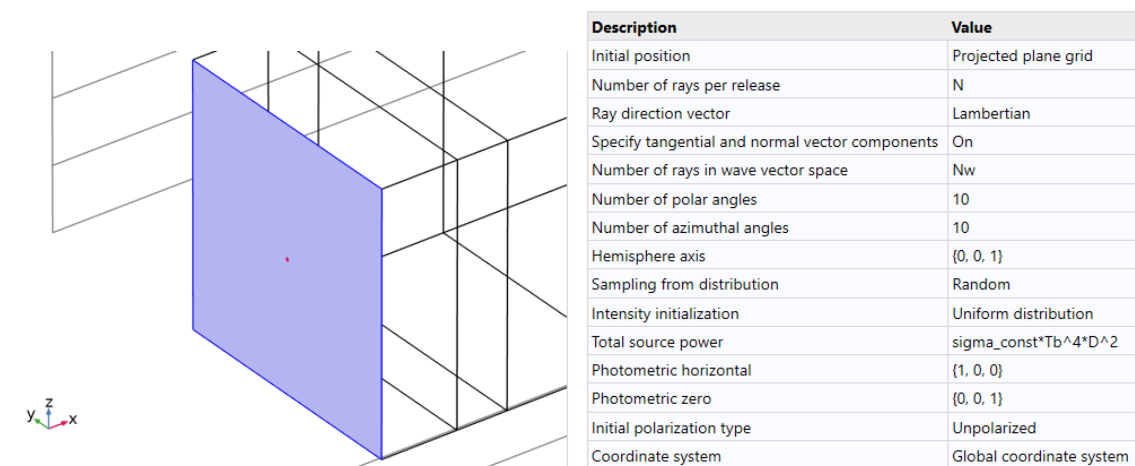


Figure B.8: The inlet boundary condition prescribing the emission of the source unpolarized power in a Lambertian distribution

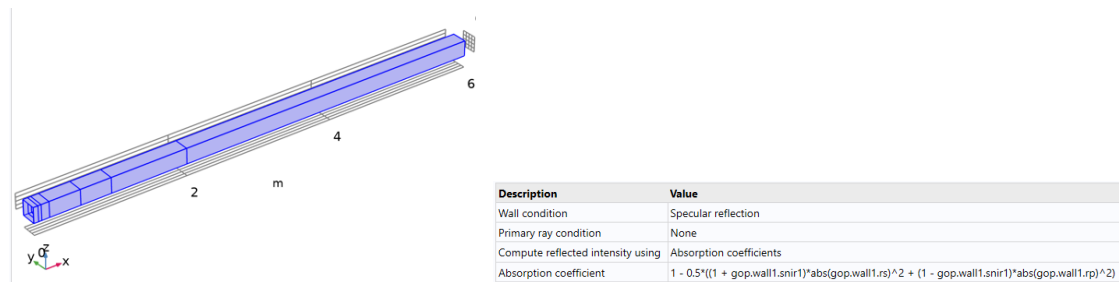


Figure B.9: The boundary condition of the specular side walls, with the absorption coefficient defined as 1 minus the reflection coefficient. The deposited ray power is evaluated from the side walls for the different lengths, which is the purpose of the partitions.

gop.wall1.rp	r_p	1	Reflection coefficient, p-polarized
gop.wall1.rs	r_s	1	Reflection coefficient, s-polarized
gop.wall1.ddelta	$\arg(\text{gop.wall1.rs}) - \arg(\text{gop.wall1.rp})$	rad	Phase shift
gop.wall1.snir1	$\text{gop.sn1} \cdot \cos(2 \cdot \text{gop.wall1.thetaesi}) + \text{gop.sn2} \cdot \sin(2 \cdot \text{gop.wall1.thetaesi})$	1	Incident normalized stokes parameter 1, rotated
gop.wall1.snir2	$-\text{gop.sn1} \cdot \sin(2 \cdot \text{gop.wall1.thetaesi}) + \text{gop.sn2} \cdot \cos(2 \cdot \text{gop.wall1.thetaesi})$	1	Incident normalized stokes parameter 2, rotated
gop.wall1.snir3	gop.sn3	1	Incident normalized stokes parameter 3, rotated
gop.wall1.Q0r	$0.5 \cdot \text{gop.Q} \cdot ((1 + \text{gop.wall1.snir1})^{\text{abs}(\text{gop.wall1.rs})^2} + (1 - \text{gop.wall1.snir1})^{\text{abs}(\text{gop.wall1.rp})^2})$	W	Total power, reflected ray

Figure B.10: In the variables of the wall boundary condition the reflection coefficients are overruled by the Fresnel reflection defined in the definitions

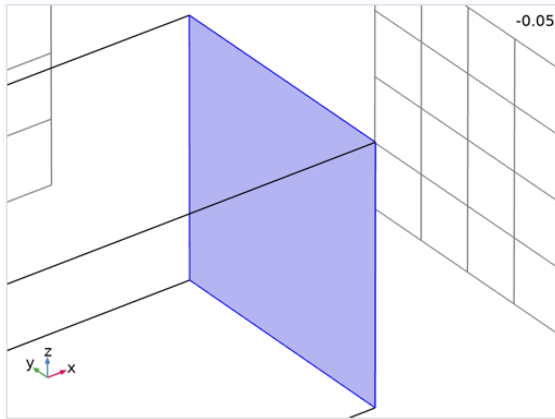


Figure B.11: At the end surface all rays are frozen

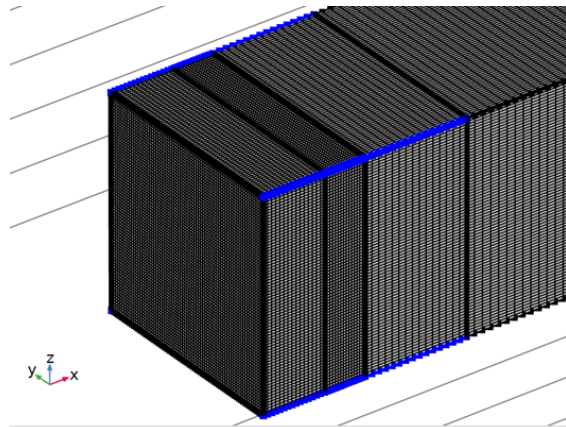


Figure B.12: The mesh assigned to the geometry using Quads for the side walls

Description	Value
Minimum element quality	0.02279
Average element quality	0.2914
Hexahedron	1659375
Quad	99750
Edge element	3580
Vertex element	32

▼ Study Settings

Time-step specification: Specify maximum path length

Length unit: m

Lengths: range(0,0.1,45) m

Characteristic group velocity: c_const m/s

Tolerance: Physics controlled

Stop condition: Active rays have intensity below threshold

Threshold ray intensity: 1e-9[W/m^2] W/m^2

Figure B.13: Mesh settings

Figure B.14: Study settings

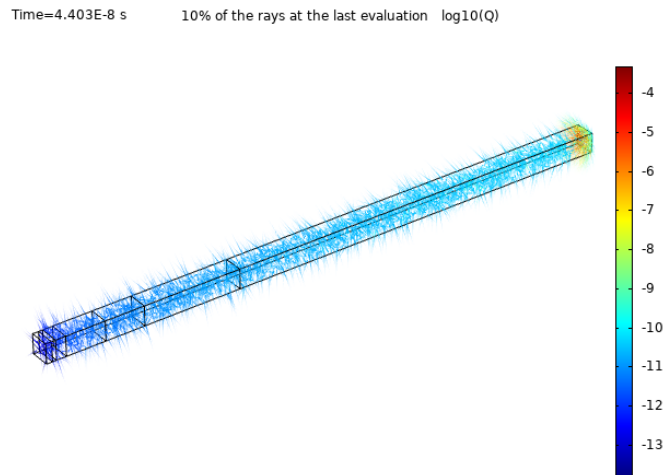


Figure B.15: After the last evaluation there are still rays that did not reach the end of the passage. The study is stopped when all rays are below the threshold value. The amount of energy that is not yet transferred to the side walls is such a small fraction that it can be neglected.

C Experimental Validation

This section of the appendix describes the experiment model in more detail. Section C.1 describes the analytical model of Miranda. Section C.2 shows the MatLab model that is used to solve this analytical model and then derive the values for the correction factor γ which is used to calculate the original raw data from the experiment. The final section C.3 shows the ray tracing model developed in COMSOL.

C.1 Analytical Model Transmission Metal Tube

The analytical model described by Miranda in *A Calculation and Experimental Verification of the Infrared Transmission Coefficient of Straight Cylindrical Metal Tubes (1996)* is solved using the MatLab model described in C.2. This chapter describes which equations are used.

The model derives the transmission through the tube by calculating the fraction of transmission of one single ray and then integrating over all the possible release conditions.

The fraction of transmission of one ray (t) is calculated by the following equation:

$$t = \frac{\rho_{\perp}^N + \rho_{\parallel}^N}{2} \quad (\text{C.1})$$

In this equation ρ_{\perp} and ρ_{\parallel} represent the reflection coefficients for perpendicular and parallel polarized radiation. The number of reflections of the ray with the side walls is indicated by N . Since the perpendicular and parallel component are considered separately it can be concluded that polarization is taken into account. However the rotation of the polarization planes is not taken into account and this is an important factor in calculating the fraction of transmission.

The reflection coefficients are calculated by the following equations, which are derived from the Fresnel equations:

$$\rho_{\parallel}(\Phi) = \frac{(n \cos(\Phi) - 1)^2 + (\kappa \cos(\Phi))^2}{(n \cos(\Phi) + 1)^2 + (\kappa \cos(\Phi))^2} \quad (\text{C.2})$$

$$\rho_{\perp}(\Phi) = \frac{(n - \cos(\Phi))^2 + \kappa^2}{(n + \cos(\Phi))^2 + \kappa^2} \quad (\text{C.3})$$

In this equation Φ indicates the angle of incidence of the radiation with respect to the metal wall. Due to the cylindrical geometry, this is the same for all bounces of a single ray. The variable n represents the refraction index and κ the absorption index of the metal wall. These are calculated using the Hagen-Ruebens equation:

$$\kappa = n = \sqrt{\frac{\lambda_0 \mu_0 c_0}{4\pi r_e}} = \sqrt{\frac{0.003 \lambda_0}{r_e}} \quad (\text{C.4})$$

In this equation λ_0 is the wavelength of the radiation, μ_0 is the permeability of vacuum, c_0 is the speed of light in vacuum and r_e is the electrical resistivity of the metal wall.

In order to derive equation C.1 the number of reflections of the ray needs to be calculated. This is done by evaluating the following equation:

$$N = \text{ceil} \left(\frac{L - d_0}{d} \right) \quad (\text{C.5})$$

Here L is the length of the tube, d_0 is the distance the ray travels along the z -axis before the first reflection and d is the distance between reflections. The *ceil*-function takes the first integer greater than the value of the fraction.

The values of d , d_0 and Φ depend on the entrance conditions of the ray and the radius of the metal tube, R . The entrance conditions are: r the distance from the origin at which the ray crosses the entrance plane, θ the angle that the ray's projection makes onto this plane with \mathbf{r} , and α is the angle

the ray makes with the z -axis. Figure C.1 shows a schematic representation of the tube, including the parameters stated below.

$$d = 2 \frac{\sqrt{R^2 - r^2 \sin^2(\theta)}}{\tan(\alpha)} \quad (\text{C.6})$$

$$d_0 = \frac{\sqrt{R^2 - r^2 \sin^2(\theta)} - r \cos(\theta)}{\tan(\alpha)} \quad (\text{C.7})$$

$$\cos(\Phi) = \frac{\sqrt{R^2 - r^2 \sin^2(\theta)}}{R} \sin(\alpha) \quad (\text{C.8})$$

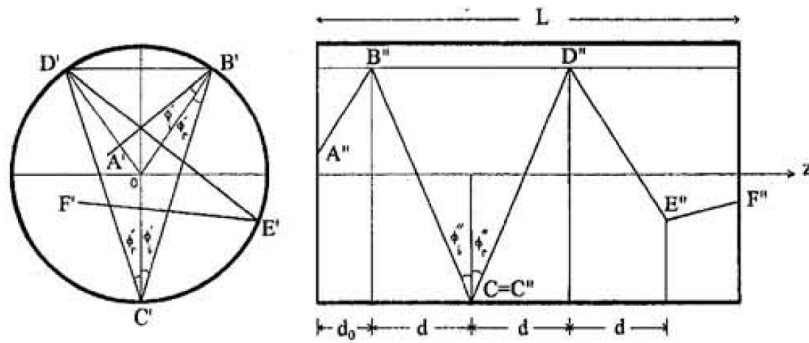


Figure C.1: Schematic representation of the metal tube of the analytical model of Miranda (1996). The angle of incidence with the wall, Φ , the distance between reflections, d and the distance before first reflection d_0 are visualized.

The total transmission through the tube is calculated with the following expression:

$$t_{total} = \frac{\int \int \int \int t(r, \theta, \alpha, \lambda) w(r, \alpha, \lambda) dr d\theta d\alpha d\lambda}{\int \int \int w(r, \alpha, \lambda) dr d\alpha d\lambda} \quad (\text{C.9})$$

Here w is a weighting function:

$$w = r \sin(2\alpha) E_b(\lambda, T) \quad (\text{C.10})$$

The black body emission is approximated by the *Wien's Formula*:

$$E_b(\lambda, T) = \frac{2\pi C_1}{n^2 \lambda^5 \left[\exp\left(\frac{C_2}{n\lambda T}\right) - 1 \right]} \quad (\text{C.11})$$

C.2 Calculation of Correction Factor γ

This function evaluates the analytical calculation formulated by Miranda in *A Calculation and Experimental Verification of the Infrared Transmission Coefficient of Straight Cylindrical Metal Tubes (1996)*, to calculate the transmission through a metal tube. The resulting transmission is used in the calculation of the correction factor γ .

```

% Analytical model as described by Miranda.
clear all; close all; clc;
%% Optical constants
sigma = 5.6704e-8; % Stefan Boltzmann constant
C1 = 0.59552138e8; % Constant [W um^4 / (m^2 sr)]
C2 = 14387.752; % Constant [um K]
%% Geometry parameters
Lvec = [0.4 1.05 9.8 25.4 50.3 79.8 114.8 150]; % Length of tube
R = 3.35; % Tube radius
re = 5e-5; % Electrical resistivity tube wall
T = 300; % Temperature source
nvac = 1; % Refractive index vacuum
%% Range of the variables
rmin = 0; % Radius of the disk
rmax = 3.15;
alphamin = 0; % Angle with z-axis at entrance
alphamax = pi/2;
lambdamin = 3; % Wavelength of radiation
lambdamax = 250;
thetamin = 0; % Angle of projection onto entrance plane
thetamax = pi;
%% Solution of the bottom of the calculation
% Does not depend on the length of the pipe
Eyb = @(lambda) (2*pi*C1)/(nvac.^2.*lambda.^5.*(exp(C2./(nvac.*
    lambda.*T))-1)); % Black body emission as function of
    wavelength
w = @(r,alpha,lambda) r.*sin(2.*alpha).*Eyb(lambda); % Weighting
    function
tbot = integral3(w,rmin,rmax,alphamin,alphamax,lambdamin,lambdamax)*
    thetamax; % Denominator of the total transmission integral
%% Solution of the top part of the fraction
% For different pipe lengths
for count = 1:length(Lvec) % Loop over all tube lengths
    L(count) = Lvec(count); % Length for this iteration
    n = @(lambda) sqrt(0.003*lambda/re); % Refractive index as
        function of wavelength
    k = @(lambda) n(lambda); % Absorptive index
    d = @(r,theta,alpha) 2.*sqrt(R^2 - r.^2.*sin(theta).^2)./tan(
        alpha); % Distance in z-direction between reflections
    d0 = @(r,theta,alpha) (sqrt(R^2 - r.^2.*sin(theta).^2)-r.*cos(
        theta))./tan(alpha); % Distance in z-direction before first
        reflection
    cosphi = @(r,theta,alpha) sqrt(R^2 - r.^2.*sin(theta).^2)./R.*sin
        (alpha); % Angle of incidence at tube wall
    N = @(r,theta,alpha) ceil((L(count)-d0(r,theta,alpha))./d(r,theta
        ,alpha)); % Number of reflections
    rhopar = @(r,theta,alpha,lambda) ((n(lambda).*cosphi(r,theta,
        alpha) - 1).^2 + (k(lambda).*cosphi(r,theta,alpha)).^2) ./ ((n
        (lambda).*cosphi(r,theta,alpha) + 1).^2 + (k(lambda).*cosphi(r

```

```

    ,theta,alpha)).^2); % Reflection coefficient for parallel
    polarized radiation
    rhoper = @(r,theta,alpha,lambda) ((n(lambda) - cosphi(r,theta,
    alpha)).^2 + k(lambda).^2) ./ ((n(lambda) + cosphi(r,theta,
    alpha)).^2 + k(lambda).^2); % Reflection coefficient for
    perpendicular polarized radiation
    t = @(r,theta,alpha,lambda) (rhopar(r,theta,alpha,lambda).^N(r,
    theta,alpha) + rhoper(r,theta,alpha,lambda).^N(r,theta,alpha))
    ./2; % Function for the transmission of one ray
    ttopvar = @(r,theta,alpha,lambda) t(r,theta,alpha,lambda).*w(r,
    alpha,lambda); % Function inside the numerator of the total
    transmission integral
    ATol = 4e-3; % Absolute tolerance for the integral
    RTol = ATol; % Relative tolerance for the integral
    ttop = integral4(ttopvar,rmin,rmax,thetamin,thetamax,alphamin,
    alphamax,lambdamin,lambdamax,'AbsTol',ATol,'RelTol',ATol);
    % Numerator of the total transmission integral
    tcoef(count) = ttop/tbot*100 % Percentage of transmission
    through the tube
end

```

This script calculates the correction factor γ which is used to convert the experimental results from the paper. The results from the previous script are used in the calculation.

```

clear all; close all; clc;
% Radii of the disk and tube
r = 3.15;
R = 3.35;
% Function for the view factor calculation
X = @(L) 1 + (1 + (r./L).^2)./(r./L).^2;
F = @(L) 0.5.*(X(L) - sqrt(X(L).^2 - 4));
% Length to diameter ratios
x = [0.4 1.05 9.8 25.4 50.3 79.8 114.8];
% Result of the analytical model
t_mod = [99.445001122990600,98.063652035308440,83.246120719938970,...
        66.049240776618300,49.309597422073340,37.505443289180120,...
        28.964379307464778];
% Calculation of gamma
for i = 1:length(x)
    gamma(i) = F(x(i))./(t_mod(i).*0.01) + (1 - F(x(i)))./(t_mod(i)
    .*0.01)).*(r/R)^2;
end

```


C.3 Numerical Model

The numerical calculation of the square passage benchmark is performed using the *Ray Optics* module in COMSOL. The figures in this section describe how the model is developed.

Name	Expression	Value	Description
r	3.15 [cm]	0.0315 m	Radius copper disks
R	3.35 [cm]	0.0335 m	Radius metal tube
L	114.8[cm]	1.148 m	Length metal tube
Tb	300[K]	300 K	Temperature copper disks

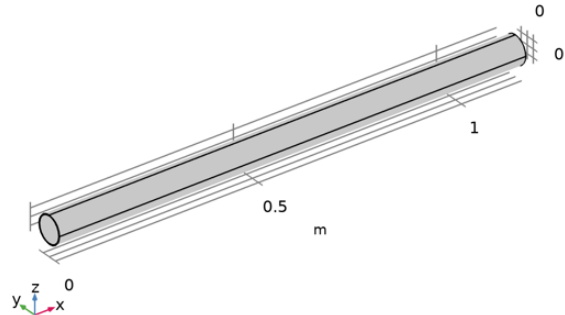
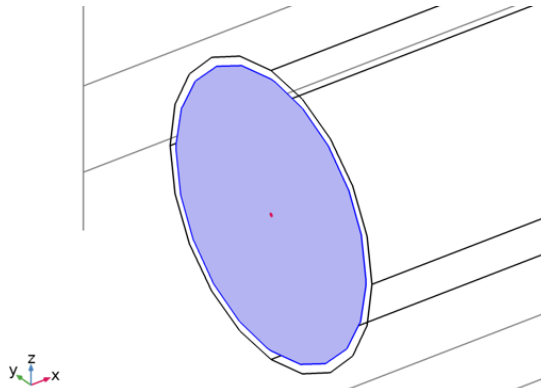


Figure C.2: The parameter list used to describe the geometry and the physics

Figure C.3: Geometry of the model

Name	Expression	Unit	Description
nreal	$\sqrt{0.003 * gop.lambd0 * 1e6[m] / 5e-5} [1/m]$		Refractive index
kimag	nreal		Absorptive index
n	nreal + kimag*i		Complex refractive index
theta_t	$\text{asin}(1/n * \sin(gop.phii))$	rad	Angle of transmittance
r_s	$(\cos(gop.phii) - n * \cos(theta_t)) / (\cos(gop.phii) + n * \cos(theta_t))$		Reflection coefficient perpendicular polarized radiation
r_p	$(n * \cos(gop.phii) - \cos(theta_t)) / (n * \cos(gop.phii) + \cos(theta_t))$		Reflection coefficient parallel polarized radiation

Figure C.4: Definitions used to define the reflection coefficients of the side walls of the geometry using the Fresnel equations. The dependency of the reflection on the wavelength is also defined.



Description	Value
Initial position	Projected plane grid
Number of rays per release	100
Ray direction vector	Lambertian
Specify tangential and normal vector components	On
Number of rays in wave vector space	50
Number of polar angles	10
Number of azimuthal angles	10
Hemisphere axis	{0, 0, 1}
Sampling from distribution	Random
Distribution function	List of values
Values	1.25887036174068e-05 1.16488997326348e-05 2.11820053793385e-05 2.26256878858298e-05 8.72749449164817e-06 1.35013188706393e-05 1.27068150691412e-05 1.700887449408059e-05 1.89455284366308e-05 2.07047380545033e-05 1.00417526803492e-05 1.79841064498275e-05 1.72549705569458e-05 8.35614978739454e-06 7.64948423646270e-06 1.36630725264194e-05 4.52442721743407e-05 1.10063809558014e-05 1.54714050917786e-05 9.27631913150957e-06
Intensity initialization	Uniform distribution
Total source power	$\text{sigma_const} * T_b^4 * r^{\wedge} 2 * \text{pi}$
Photometric horizontal	{1, 0, 0}
Photometric zero	{0, 0, 1}
Initial polarization type	Unpolarized
Coordinate system	Global coordinate system

Figure C.5: The inlet boundary condition prescribing the emission of the source unpolarized power in a Lambertian distribution and 20 values for wavelengths

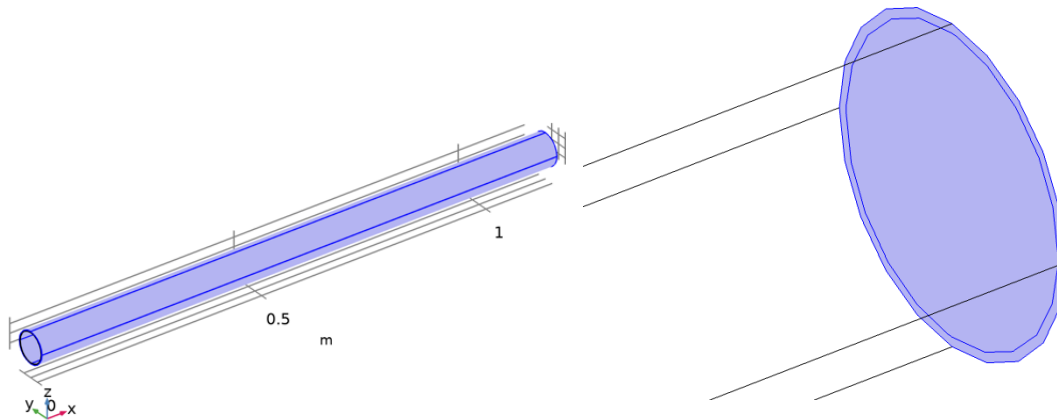


Figure C.6: The boundary condition of the specular side walls

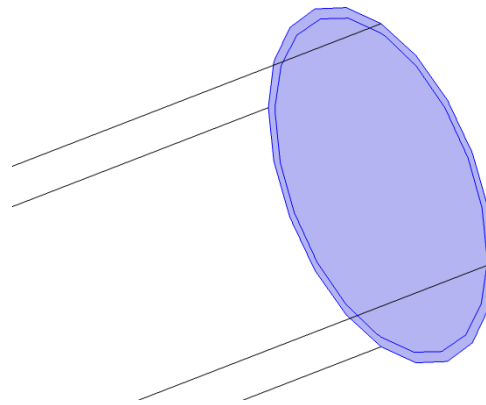


Figure C.7: All rays are frozen at the end of the tube. The deposited ray power is evaluated on the copper disk

Description	Value
Minimum element quality	0.0
Average element quality	0.6994
Tetrahedron	4924948
Pyramid	287000
Triangle	4448
Quad	287000
Edge element	6540
Vertex element	16

Figure C.8: Mesh settings

Study Settings	
Time-step specification:	Specify maximum path length
Length unit:	m
Lengths:	range(0,0.75,21) m
Characteristic group velocity:	c_const m/s
Tolerance:	Physics controlled
Stop condition:	Active rays have intensity below threshold
Threshold ray intensity:	1e-7[W/m ²] W/m ²

Figure C.9: Study settings

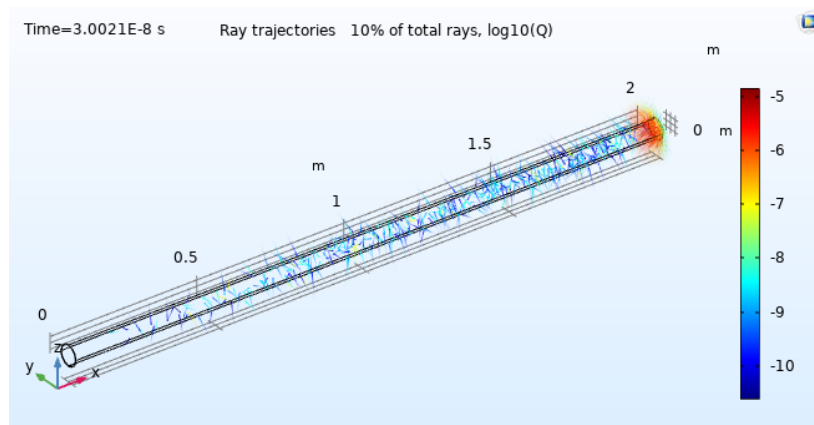


Figure C.10: After the last evaluation there are still rays that did not reach the end of the passage. The study is stopped when all rays are below the threshold value. The amount of energy that is not yet transferred to the side walls is such a small fraction that it can be neglected.

D Proof of Concept Experiment

A proof of concept experiment is conducted to evaluate if it is possible to measure thermal radiation using a simple experiment set up. The reflected radiation from different surfaces is measured using an infrared camera. A black radiation source is used, which is able to reach a maximum temperature of 150°C . The infrared camera used in this experiment is a FLIR T425. The reflected radiation is measured for different angles of incidence, as shown in the schematic figure below.

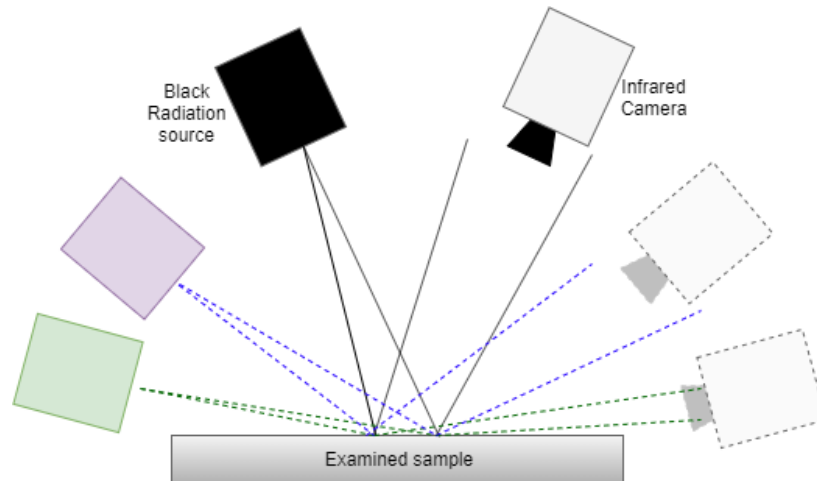


Figure D.1: Schematic figure of the experiment, which measures the reflective properties of the examined sample using a black radiation source and an infrared camera.

In order to effectively assess the reflected radiation it is important to know the emitted radiation from the black radiation source. This is also used to calibrate the camera, since the emission coefficient of the source is $\epsilon = 1$. The figure below shows the measured temperatures of the source with the infrared camera at different angles. The results from the measurements are displayed in the graph below that shows the emission of the source as a function of the angle. It can be concluded that the radiation is emitted in a conical shape.

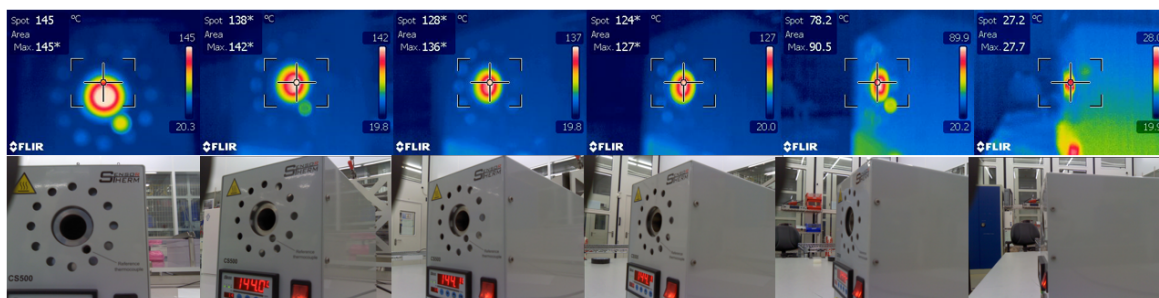


Figure D.2: Measurements of the black radiation source for different angles

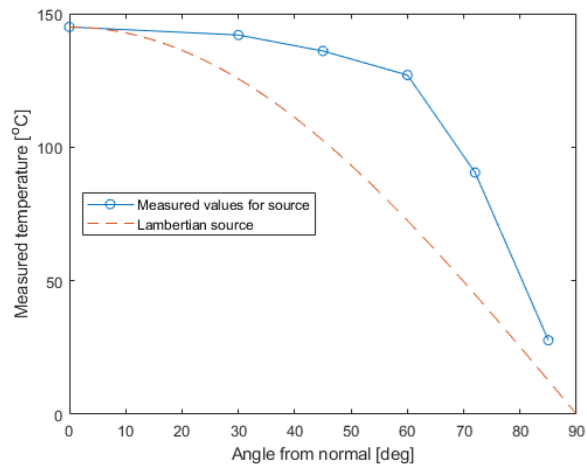


Figure D.3: The emission of the source as a function of the angle.

For the proof of concept two samples are examined. Both samples are metallic surfaces, but one reflects radiation diffusely and the other reflects specularly.

For the diffuse reflecting surface, a piece of aluminum foil is wrinkled which makes sure the irradiation is reflected in all directions. This can be seen from the measurements, where the measured temperature is between 22°C and 24.5°C.

For the second measurement a flat metal surface is examined. The radiation is clearly reflected specularly. The shape of the radiation source can be seen clearly in the infrared images. The measured intensity is also higher than for the diffuse reflection. There seems to be angle dependency in the reflection. However to be able to compare this angle dependency with the theory more accurate measurements are required.

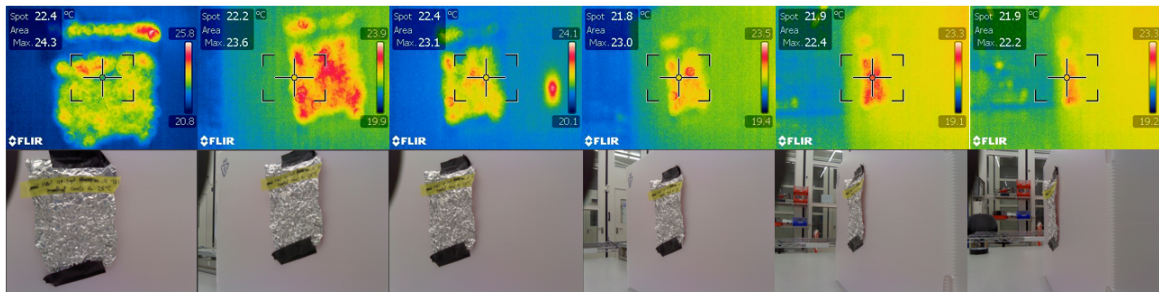


Figure D.4: Measurements of the reflected radiation for wrinkled aluminum foil that shows diffuse reflection.

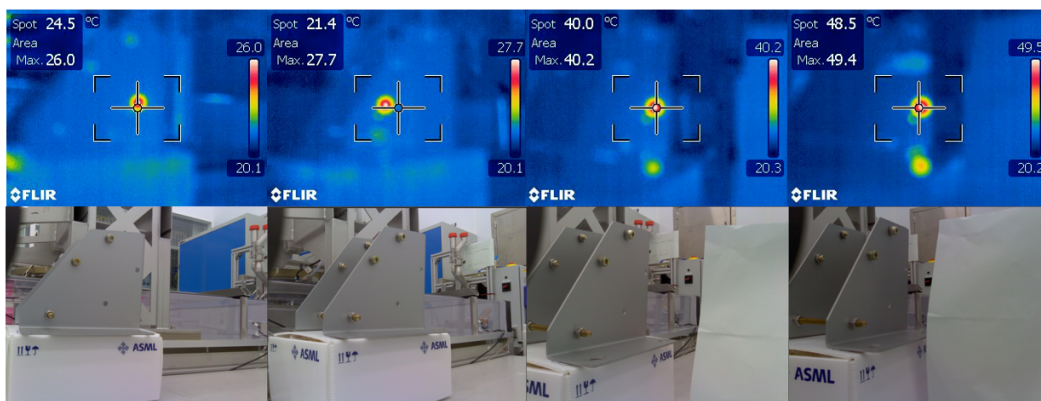


Figure D.5: Measurements of the reflected radiation for a flat metal surface that shows specular reflection.

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conduct¹.

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

12-02-2020

Name

Mathieu Wijnen

ID-number

0888043

Signature

Mathieu Wijnen

Submit the signed declaration to the student administration of your department.

¹ See: <http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/>

The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU