

## Optical Tomography in Combustion

**Evseev, Vadim; Clausen, Sønnik; Fateev, Alexander**

*Publication date:*  
2013

*Document Version*  
Peer reviewed version

[Link back to DTU Orbit](#)

*Citation (APA):*

Evseev, V., Clausen, S., & Fateev, A. (2013). Optical Tomography in Combustion. Technical University of Denmark, Department of Chemical Engineering.

## DTU Library

Technical Information Center of Denmark

---

### 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
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# Optical Tomography in Combustion

## PhD Thesis



**Supervisor:** Sønnik Clausen, Senior Scientist

**Co-supervisor:** Alexander Fateev, Senior Scientist

**PhD Student:** Vadim Evseev

November 2012

## **Optical Tomography in Combustion**

PhD Thesis

2012

**Supervisor:** Sønnik Clausen, Senior Scientist

**Co-supervisor:** Alexander Fateev, Senior Scientist

**PhD Student:** Vadim Evseev

Copyright:      Reproduction of this publication in whole or in part must include the customary bibliographic citation, including author attribution, report title, etc.

Cover photo:    Flat flame burner

Published by:   Department of Chemical and Biochemical Engineering, Søtofts Plads, Building 229, DK-2800 Kgs. Lyngby, Denmark

Request thesis [www.kt.dtu.dk](http://www.kt.dtu.dk)

from:

ISSN:            [0000-0000] (electronic version)

ISBN:            13: 978-87-92481-84-9 (electronic version)

ISSN:            [0000-0000] (printed version)

ISBN:            13: 978-87-92481-84-9 (printed version)

# Abstract

The new methodology of optical infrared tomography of flames and hot gas flows was developed in the PhD project with a view to future industrial applications. In particular, the methodology for the tomographic reconstruction of an axisymmetric lab flame temperature profile was developed and tested in the lab using Fourier transform infrared spectroscopy techniques, including a new tomographic measurement scheme, sweeping scanning, having great potential for industrial applications with limited optical access. The results were compared to the reference point measurements on the same flame and the deviations are discussed. The methods are shown to have promising potential for future industrial applications.

The new multichannel infrared spectrometer system as a first prototype of the infrared spectroscopic tomography system was developed in the PhD project for simultaneous fast transient infrared spectral measurements at several line-of-sights with a view to applications for tomographic measurements on full-scale industrial combustion systems. The system was successfully applied on industrial scale for simultaneous fast exhaust gas temperature measurements in the three optical ports of the exhaust duct of the large Diesel engine. The results were compared to the measurements performed by another system employing spectral properties of nitric oxides in the ultraviolet range. A good agreement was observed between the results obtained using the two different systems.

In the context of the PhD project, it was also important to investigate the spectral properties of major combustion species such as carbon dioxide and carbon monoxide in the infrared range at high temperatures to provide the theoretical background for the development of the optical tomography methods. The new software was developed for the line-by-line calculations of the transmission spectra of a carbon dioxide/carbon monoxide mixture which is able to use within reasonable time the most recent but huge CDS-4000 database containing updated high-temperature spectroscopic line-by-line data. The software was used for the line-by-line calculations of the transmission spectra of the carbon dioxide/carbon monoxide mixture at high temperatures and the results were compared to the measurements in the high-temperature flow gas cell carried out before the PhD project. The results and discussion are presented in a journal article [Evseev *et al.* *JQSRT* 113 (2012) 2222, 10.1016/j.jqsrt.2012.07.015] included in the PhD thesis as an attachment.

The knowledge and experience gained in the PhD project is the first important step towards introducing the advanced optical tomography methods of combustion diagnostics developed in the project to future industrial applications.

# Preface

Mixing of cold and hot turbulent reacting gas flows, flame propagation and development are all fast transient phenomena appearing in the three dimensional space. Their temporal development can be described, e.g., by a stack of two dimensional slices of tomographic images of gas temperature and species concentrations profiles in the direction of the net mass propagation.

The knowledge on two dimensional gas temperature profiles is essential for the measurement of the other parameters characterizing combustion phenomena including two dimensional species concentrations profiles and hence helps in a better understanding of the phenomena in various combustion environments and facilitates further development of computational fluid dynamics codes, the optimization of burner/engine design and operation, etc.

The PhD project concerned the development of new optical emission/absorption tomography methods for combustion diagnostics to obtain the two dimensional information about the gas temperature. The project comprised combined theoretical, numerical and experimental approach, and was overlapping with ongoing projects at DTU Chemical Engineering (before 1 January 2012 Risø DTU, Optical Diagnostics Group).

The aim was to develop tomographic methods for lab scale measurements on a lab burner flame with a view to future full-scale applications, e.g., on diesel engines, industrial boilers, etc. The project was based on in situ optically-based spectroscopic methods in the infrared range developed at the department and, moreover, involved significant extensions and new development in the field of optical tomography.

The work was performed in close contact with the other members in the department and partners. The results contributed to the extension of department's capabilities for in situ optical diagnostics in various combustion environments. The project was centrally placed within department's strategy for emerging energy technologies.

Kgs. Lyngby (Denmark), November 2012

Vadim Evseev  
PhD Student

# Table of Contents

<b>Abstract</b> .....	<b>3</b>
<b>Preface</b> .....	<b>4</b>
<b>1. Introduction</b> .....	<b>8</b>
<b>1.1. Motivation</b> .....	<b>8</b>
<b>1.2. Purpose</b> .....	<b>9</b>
<b>1.3. Objectives</b> .....	<b>9</b>
1.3.1. Tomographic Reconstruction of a Lab Flame Temperature Profile.....	10
1.3.2. Simultaneous Fast Exhaust Gas Temperature Measurements on a Large Diesel Engine .....	11
1.3.3. Line-by-line Modeling of Gas Spectra.....	12
<b>1.4. Background</b> .....	<b>12</b>
1.4.1. Optical Tomography of Gases and Flames.....	12
1.4.2. Combustion Related Measurements on Engines .....	18
<b>2. Experimental Details</b> .....	<b>20</b>
<b>2.1. Multichannel IR Spectrometer System</b> .....	<b>20</b>
2.1.1. Introduction and Remarks.....	20
2.1.2. General Description of the System .....	20
2.1.3. Retrieval of IR Spectra from IR Images .....	21
2.1.4. Working Spectral Range .....	22
2.1.5. Number of Fibers Coupled onto the Entrance Slit and Time Resolution .....	23
2.1.6. Calibration of the x-axis (wavelength axis) .....	24
2.1.7. Instrument Line Shape Function and Spectral Resolution.....	26
2.1.8. Calibration of the y-axis (spectral radiance axis).....	28
2.1.9. Thermal Stability of the System .....	29
<b>2.2. Small-Scale Laboratory Burner</b> .....	<b>31</b>
2.2.1. Purpose.....	31
2.2.2. Description of the Burner .....	31
2.2.3. Setup of the Burner.....	32
2.2.4. Operation Parameters.....	34
2.2.5. Flame Fluctuations.....	35
2.2.6. Reference Temperature Profile Measurements in the Work of Ref. [74] .....	36
<b>2.3. FTIR Tomographic Measurements         on the Lab Burner</b> .....	<b>36</b>

2.3.1.	Introduction and Important Remarks.....	36
2.3.2.	FTIR Spectrometer.....	36
2.3.3.	Optical Setup for the FTIR Measurements .....	38
2.3.4.	Calibration of the System.....	42
2.3.5.	Thermal Stability of the System.....	43
2.3.6.	Parallel Scanning Scheme of Measurements.....	43
2.3.7.	Transmission and Emission Measurements .....	47
2.3.8.	Sweeping Scanning Scheme of Measurements .....	47
<b>2.4.</b>	<b>Simultaneous Fast Exhaust Gas Temperature Measurements on a Large Diesel Engine .....</b>	<b>51</b>
2.4.1.	Introduction and Remarks.....	51
2.4.2.	Large Diesel Engine.....	51
2.4.3.	Measurement Layout .....	52
<b>3.</b>	<b>Theory and Methods .....</b>	<b>54</b>
<b>3.1.</b>	<b>Line-by-line Modeling of Gas Spectra.....</b>	<b>54</b>
<b>3.2.</b>	<b>Gas Temperature Measurements on the Large Diesel Engine.....</b>	<b>58</b>
3.2.1.	Remarks.....	58
3.2.2.	The Spectral Emission-Absorption Method of Gas Temperature Measurement.....	58
3.2.3.	The Assumed Value of Spectral Absorptance .....	60
3.2.4.	Gas Temperature Calculation.....	63
<b>3.3.</b>	<b>Tomographic Algorithms for Gas Temperature Profile Reconstruction.....</b>	<b>63</b>
3.3.1.	Introduction and Remarks.....	63
3.3.2.	Problem Statement.....	64
3.3.3.	About Absorption .....	65
3.3.4.	Parallel Scanning .....	65
3.3.5.	The Differential Equation of Emission and Absorption.....	66
3.3.6.	A Method to Solve Integral Equation (40) .....	68
3.3.7.	Sweeping Scanning .....	71
3.3.8.	A Method for Solving Integral Equation (52).....	72
<b>4.</b>	<b>Results and Discussion.....</b>	<b>74</b>
<b>4.1.</b>	<b>Tomographic Reconstruction of the Lab Flame Temperature Profile.....</b>	<b>74</b>
4.1.1.	Introduction and Remarks.....	74
4.1.2.	Results for Stoichiometric Combustion ( $\varphi = 1$ ) .....	74
4.1.3.	Results for Lean Combustion ( $\varphi = 0.8$ ).....	77
4.1.4.	Discussion on the Deviation from the Results of Ref. [74].....	80
<b>4.2.</b>	<b>Simultaneous Fast Exhaust Gas Temperature Measurements on the Large Diesel Engine.....</b>	<b>82</b>
4.2.1.	Introduction and Remarks.....	82
4.2.2.	Emission and Transmission IR Spectra of the Exhaust Gas.....	82
4.2.3.	Exhaust Gas Temperature in the Three Optical Ports of the Exhaust Duct.....	83

<b>5.</b>	<b>Summary</b> .....	<b>86</b>
<b>5.1.</b>	<b>Tomographic Reconstruction of the Lab Flame Temperature Profile</b> .....	<b>86</b>
<b>5.2.</b>	<b>Simultaneous Fast Exhaust Gas Temperature Measurements on the Large Diesel Engine</b> .....	<b>90</b>
<b>5.3.</b>	<b>Line-by-line Modeling of Gas Spectra</b> .....	<b>92</b>
	<b>Acknowledgements</b> .....	<b>93</b>
	<b>References</b> .....	<b>93</b>
	<b>Appendix A.</b>	
	<b>The C++ Code of the Software     for the Line-by-line Calculations</b> .....	<b>97</b>
	<b>Appendix B.</b>	
	<b>High-resolution transmission measurements of CO<sub>2</sub>     at high temperatures     for industrial applications</b> .....	<b>168</b>



# 1. Introduction

## 1.1. Motivation

Mixing of cold and hot turbulent reacting gas flows, flame propagation and development are all fast transient phenomena appearing in the three dimensional space. Their temporal development can be described, e.g., by a stack of two-dimensional (2D) slices of tomographic images of gas temperature and species concentrations profiles in the direction of the net mass propagation.

The knowledge on 2D gas temperature profiles is essential for the measurement of the other parameters characterizing combustion phenomena including 2D species concentrations profiles and hence helps in better understanding of the phenomena in various combustion environments and facilitates further development of computational fluid dynamics (CFD) codes, the optimization of burner/engine design and operation, etc.

DTU Chemical Engineering (before 1 January 2012 Risø DTU, Optical Diagnostics Group) has successfully applied Fourier transform infrared (FTIR) spectroscopy with fiber optical probes for many years in various large scale combustion environments [1, 2, 3, 4, 5, 6, 7]. The FTIR data rate was limited typically to a few Hz because of a mechanical movement of a mirror. Therefore it was possible to follow only macro temporal fluctuations of the gas temperature.

The PhD project was extensively overlapping with the project “Fast optical measurements and imaging of flow mixing” (Energinet.dk ForskEL projektnr. 2008-1-0079) [11] in which a data rate in the kHz region was achieved using a fast infrared (IR) camera (CEDIP Infrared Systems, Titanium, detector: 560M, InSb, 640×512 pixels) coupled with an IR-optimized spectrometer (Sec. 2.1.2). The IR camera employs a new generation fast focal plane diode (InSb) array detector (FPA) and allows to utilize an exposure time of down to 7  $\mu$ s for spectroscopic measurements in the 1.5-5.1  $\mu$ m range making possible fast gas temperature measurements and spectral-resolved IR imaging of the flames and turbulent reacting flows [11].

However those measurements were averaged within some viewing angle and gave information only about “local” (i.e. 1D) gas temperature. Two-dimensional (2D) or even three-dimensional (3D) gas temperature profiles would be more helpful in understanding the combustion phenomena and thus would facilitate further development of CFD codes, the optimization of burner/engine design and operation, etc.

A 3D temperature profile can be assembled of horizontal tomographic slices (in the plane (x, y)) arranged in a vertical stack (along the z-axis). Each horizontal tomographic slice represents a 2D plot of the gas temperature. Because a flame or hot flue gas is a

highly dynamic object, temporal variations of temperature have to be taken into account. Hence a flame/hot flue gas can be well described by a stack of 2D tomographic slices being themselves a function of time  $t$ , i.e. 2Dt.

## 1.2. Purpose

The main purpose of the project was to start-up the development of a 2Dt IR spectroscopic tomography system and methods for in situ investigation of small-scale flames and hot flue gases by means of the 2Dt reconstruction of temperature and exploration of applicability of the developed technique to the full-scale combustion diagnostics on industrial combustion systems such as, e.g., a large Diesel engine.

In this context, it was also essential to investigate the spectral properties of major combustion species such as carbon dioxide ( $\text{CO}_2$ ) and carbon monoxide ( $\text{CO}$ ) in the IR spectral range at high temperatures with a view to gas temperature and species concentrations calculations to provide the theoretical background for the development of the optical tomography methods.

The knowledge and experience built up in the project is the first step to introduce advanced optical tomography methods to future industrial applications.

## 1.3. Objectives

The PhD project concerned the development of new optical emission/absorption tomography methods for combustion diagnostics to obtain the 2D information about the gas temperature.

The aim was to develop tomographic methods for lab scale measurements on a lab burner flame with a view to future full-scale applications, e.g., on diesel engines, industrial boilers, etc. The project was based on in situ optically-based spectroscopic methods in the IR range developed at DTU Chemical Engineering (before 1 January 2012 Risø DTU, Optical Diagnostics Group).

The development of the tomographic methods was conducted in the three main directions:

1. The development of tomographic algorithms for 2D gas temperature profile reconstruction and application of the developed algorithms using FTIR spectroscopy equipment (developed at the department) on a laboratory axisymmetric flame (see Sec. 1.3.1 for more details). This part of the PhD project was extensively overlapping with the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8].

2. The development of a **multichannel IR spectrometer system** (a first prototype of the 2Dt IR spectroscopic tomography system) for simultaneous spectral measurements at several line-of-sights (LOS) with a view to application for tomographic measurements on full-scale industrial combustion systems and the application of the developed system for simultaneous exhaust gas temperature measurements at the three optical ports of the exhaust duct of a large Diesel engine (Sec. 1.3.2). This part of the PhD project was closely related to the EU-funded HERCULES-B project “High-efficiency engine with ultra-low emissions for ships” carried out under 7<sup>th</sup> Framework Program “Sustainable Surface Transport” (grant agreement SCP7-GA-217878) presented in reports [9], to the project Energinet.dk ForskEL projektnr. 2008-1-0079 “Fast optical measurements and imaging of flow mixing” reported in Ref. [11], and to the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8].
3. The investigation of spectral properties of major combustion species such as CO<sub>2</sub> and CO in the IR spectral range at high temperatures to provide the theoretical background for the development of the optical tomography methods (Sec. 1.3.3), in particular, the line-by-line modeling of the transmission spectra of CO<sub>2</sub> in the 2.7, 4.3 and 15 μm regions and of CO in the 4.7 μm region at temperatures up to 1773 K (1500 °C), the volume fractions of CO<sub>2</sub> in the range 1-100% and at atmospheric pressure based on the HITEMP [12, 13] and CDSD [14, 15, 16] spectroscopic databases. The details on this part of the PhD project are published in a journal article [17] (*Evseev et al., High-resolution transmission measurements of CO<sub>2</sub> at high temperatures for industrial applications, Journal of Quantitative Spectroscopy and Radiative Transfer 113 (17) (November 2012) 2222-2233 <http://dx.doi.org/10.1016/j.jqsrt.2012.07.015>*) included in the PhD thesis (Appendix B).

### 1.3.1. Tomographic Reconstruction of a Lab Flame Temperature Profile

The objective of this part of the PhD project is the development of tomographic algorithms for 2D gas temperature profile reconstruction and application of the developed algorithms using FTIR spectroscopy equipment (developed at the department) on a laboratory axisymmetric flame. This part of the PhD project was extensively overlapping with the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8].

The tomographic algorithms has to be developed for the case of an axisymmetric temperature profile of a flame/hot flue gas in order to test the methods under more simple conditions such that possible deviations and errors could be identified more clearly.

The small scale laboratory burner producing a flame with known (reference) axisymmetric temperature profile is necessary to prove the developed tomographic algorithms for 2Dt reconstruction of gas temperature profiles.

The goal of the 2Dt measurements on the laboratory burner is to compare the temperature profiles obtained in this project to the reference profiles at different operation conditions of the burner.

The 2Dt measurements has to be performed in the 1.5-5.1 μm spectral range covering several rotational-vibrational emission/absorption bands of important combustion

species (e.g., CO<sub>2</sub>, CO, H<sub>2</sub>O, C<sub>x</sub>H<sub>y</sub>) using FTIR spectroscopy equipment developed at the department. The department has rich experience employing the FTIR spectroscopy techniques and equipment [1, 2, 3, 4, 5, 6, 7] which provide reliable spectral measurements hence using those techniques and equipment would provide the reconstruction algorithms with input data having minimal uncertainties. In that way possible deviations and errors in the results can be identified more clearly.

### 1.3.2. Simultaneous Fast Exhaust Gas Temperature Measurements on a Large Diesel Engine

Simultaneous spectral measurements at several line-of-sights are essential for on-line time-resolved 2D tomographic measurements of gas temperature and species concentrations in a cross-section of a flame or hot flue gas.

The objective for this part of the PhD project was to develop a **multichannel IR spectrometer system** (a first prototype of the 2Dt IR spectroscopic tomography system) for simultaneous fast time-resolved transient IR spectral measurements at several line-of-sights and to apply the system for fast time-resolved simultaneous IR measurements of exhaust gas temperature in the three optical ports of the exhaust duct of a large Diesel Engine (2-stroke 4-cylinder test marine Diesel engine 4T50ME-X at MAN Diesel & Turbo, Copenhagen [9]). The exhaust duct of one of the cylinders was slightly modified (by MAN Diesel & Turbo) by making three optical ports in it to allow three line-of-sight optical measurements across the exhaust gas stream coming from the exhaust valve of the cylinder.

The development and the application of the **multichannel IR spectrometer system** was also relevant to the EU-funded HERCULES-B project “High-efficiency engine with ultra-low emissions for ships” carried out under 7<sup>th</sup> Framework Program “Sustainable Surface Transport” (grant agreement SCP7-GA-217878) presented in reports [9], to the project Energinet.dk ForskEL projektnr. 2008-1-0079 “Fast optical measurements and imaging of flow mixing” reported in Ref. [11], and to the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8].

It should be noted that the simultaneous fast time-resolved transient IR spectral measurements in the three optical ports of the exhaust duct of a large Diesel engine were performed for the first time. The gas temperature was obtained simultaneously at the three ports from the simultaneous IR measurements as the brightness temperature of the intensity of the 4.3 μm CO<sub>2</sub> band.

The gas temperature must be known to measure, e.g., NO concentration by optical spectroscopy as well as to measure other pollutant species concentrations or other quantities characterizing combustion phenomena. The gas temperature in the three ports of the exhaust duct was also obtained from the structure of the ultraviolet (UV) NO 226 nm absorption band using a different system [9, 10]. A reasonable agreement between the IR CO<sub>2</sub> measurements of the exhaust gas temperature (performed using the **multichannel IR spectrometer system**) and UV NO measurements was observed [9, 10] (Sec. 4.2).

### 1.3.3. Line-by-line Modeling of Gas Spectra

The objective of this part of the PhD project was the investigation of spectral properties of major combustion species such as CO<sub>2</sub> and CO in the IR spectral range at high temperatures with a view to gas temperature and species concentrations calculations to provide the theoretical background for the development of the optical tomography methods, in particular, the line-by-line modeling of the transmission spectra of CO<sub>2</sub> in the 2.7, 4.3 and 15 μm regions and of CO in the 4.7 μm region at temperatures up to 1773 K (1500 °C), the volume fractions of CO<sub>2</sub> in the range 1-100% and at atmospheric pressure based on the HITEMP [12, 13] and CDSD [14, 15, 16] spectroscopic databases.

The details on this part of the PhD project are published in a journal article [17] (*Evseev et al., High-resolution transmission measurements of CO<sub>2</sub> at high temperatures for industrial applications, Journal of Quantitative Spectroscopy and Radiative Transfer 113 (17) (November 2012) 2222-2233 <http://dx.doi.org/10.1016/j.jqsrt.2012.07.015>*) included in the PhD thesis (Appendix B).

The transmission spectra of CO<sub>2</sub> in the 2.7, 4.3 and 15 μm regions at temperatures up to 1773 K (1500 °C), the volume fractions of 1, 10 and 100% and at atmospheric pressure (1.00±0.01 atm) were measured at the department (before the PhD project). The spectra were recorded in a high-temperature flow gas cell developed at the department [2, 17, 18, 19, 20] and using an FTIR spectrometer at a nominal resolution of 0.125 cm<sup>-1</sup>.

The goal of this part of the PhD project was to perform the line-by-line modeling of the transmission spectra of the CO<sub>2</sub>/CO mixture at the above conditions based on the HITEMP-1995 [12], HITEMP-2010 [13], CDSD-HITEMP [14] and CDSD-4000 [16] databases and to compare the modeling results with the measurements.

The line-by-line modeling procedure is presented in detail in this PhD thesis in Sec. 3.1 and implemented in software developed using the C++ Programming Language (Microsoft Visual Studio 2008). The software is developed in this PhD project and the code is given in Appendix A. It should be noted that the detailed description of the line-by-line procedure as well as the software code are not given in Ref. [17] (which is included in the PhD thesis, Appendix B).

## 1.4. Background

### 1.4.1. Optical Tomography of Gases and Flames

Wright et al. [21] reported high-speed continuous imaging of hydrocarbon fuel distribution and mixing in the combustion chamber of an automotive engine (chemical species tomography). The measurement grid consisted of 27 dual-wavelength optical paths implemented in one cylinder of an otherwise standard four-cylinder port-injected gasoline engine, using special optical access layer carrying embedded optical fibers and collimators. Dual-wavelength measurements were recorded on each channel at 100 kilo-samples per second, prior to off-line processing that reduced the effective frame rate to 3000–4000 frames per second. The performance of the system was assessed, using run-

ning conditions chosen to provide a qualitatively known (homogeneous) fuel distribution for validation purposes.

In order to develop the chemical species tomography system described in Ref. [21], Pal et al. [22] carried out various computational steps to address the problem of measuring minor species concentrations using single-pass, short path-length absorption techniques in the mid-IR. The focus was primary on the imaging of CO in a combustion exhaust as a case study, with an average concentration of 10 ppm over a 50 mm diameter cross-section, taking account of the presence of other absorbing species. A feasible beam arrangement for tomographic imaging providing 48 measurements of path concentration integral was suggested. Representative phantom reconstructions were obtained with results for application to such dynamic gaseous subjects.

Ma et al. [23, 24, 25, 26] developed a technique for obtaining simultaneous tomographic images of temperature and species concentrations based on hyperspectral absorption spectroscopy. The hyperspectral information is reported to enable several key advantages when compared to traditional tomography techniques based on limited spectral information. Those advantages included a significant reduction in the number of required projection measurements, and an enhanced insensitivity to measurements/inversion uncertainties. Those advantages greatly facilitated the practical implementation and application of the tomography technique. The developed technique and a prototype sensor were tested using a near-adiabatic, atmospheric-pressure laboratory Hencken burner. The spatial and temporal resolution enabled by that new sensing technique was expected to resolve several key issues in practical combustion devices.

An et al. [27] demonstrated and validated temperature imaging using hyperspectral H<sub>2</sub>O absorption tomography in controlled experiments. Fifteen wavelengths were monitored on each of 30 laser beams to reconstruct the temperature image in a 381 mm × 381 mm square room temperature plane that contained a 102 mm × 102 mm square zone of lower or higher temperature. The temperature of the 102 mm × 102 mm block zone was changed from 293 to 353 K (20 to 80 °C) in increments of 20 K. The room temperature was constant at 297 K (24 °C). Each of the 225 temperatures in the entire 15-by-15 domain was treated as unknown and independent during the solution. No regularization or a priori information was used to determine the results. The experimental reconstruction results generally reproduced the temperature distribution faithfully, and the locations and sizes of the block zone were also well distinguished. As regards the sensitivity, it is reported in Ref. [27] that even when the internal zone was only 4 K cooler than the surroundings, its presence was still detectable.

Gillet et al. [28] used an IR sensor for measurements of spatial distribution of hydrocarbon concentration in a model of a gas turbine combustor, using absorption tomography along multiple line-of-sights. The sensor is reported to have the potential for monitoring the degree of premixedness of reacting fuel and air in stationary gas turbine combustors, where operation with lean premixed mixtures is important for reduction of NO<sub>x</sub> emissions. A numerical simulation, using a stochastic representation of a turbulent hydrocarbon distribution in the cross section of a duct was generated, in order to evaluate the ability of computed tomography to reconstruct the initial distribution. The algebraic reconstruction tomography algorithm for planar parallel geometry of line-of-sights was employed.

Ravichandran and Gouldin [29, 30] reported that established methods for obtaining the inverse of the Radon transform, the underlying reconstruction problem in the determination of asymmetric temperature and species concentrations profiles from absorp-

tion measurements, are not well-suited for limited data problems which are the case in the optical tomography of gases and flames. A new method that minimized the number of measurements required by making use of a priori information such as the smoothness of the absorption coefficient profiles and constraints on its domain and range, was applied to this inversion problem [30]. Test results obtained by using simulated absorption measurements generated from phantom temperature and NO concentration distributions are presented in Ref. [30]. Comparison with the results obtained using the convolution backprojection method, a widely used method for solving the discrete data inverse Radon transform problem is also presented in Ref. [30]. It was found that the developed method is superior to the convolution backprojection method.

Best et al. [31] applied tomographic reconstruction techniques to line-of-sight FTIR emission and transmission measurements to derive spectra corresponding to small volumes within an axisymmetric ethylene diffusion flame. The diameter of the burner was 1 cm. From those spectra, point values for species temperatures and relative concentrations was determined for CO<sub>2</sub>, H<sub>2</sub>O, alkanes, alkenes, alkynes, and soot. The reconstruction program published by Shepp and Logan [32] was adapted for the work of Ref. [31]. It was corrected for a minor indexing error which showed up only in cases where the number of projections is relatively small ( $\ll 100$ ). The results of the reconstruction were consistent with the experimental data in that transmittance across the flame diameter reconstructed from the image properties agreed with the measured transmittance [31].

Similar research was also performed on coal flames [33]. Tomographic reconstruction techniques were applied to line-of-sight FTIR emission/transmission measurements to derive spectra that correspond to small volumes within a coal flame. From those spectra, spatially resolved point values for species temperature and relative concentrations can be determined. The technique was used to study the combustion of Montana Rosebud subbituminous coal burned in a transparent wall reactor. The coal was injected into the center of an up-flowing preheated air stream to create a stable flame. The visible diameter of the flame was about 1 cm. Values for particle temperature, relative particle density, relative soot concentration, the fraction of ignited particles, the relative radiance intensity, the relative CO<sub>2</sub> concentration and the CO<sub>2</sub> temperature were obtained as functions of distance from the flame axis and height above the coal injector nozzle. The spectroscopic data are reported to be in good agreement with visual observations and thermocouple measurements. The highest CO<sub>2</sub> temperatures are reported to be 2200 to 2600 K (1927 to 2327 °C) and the highest particle temperatures 1900 to 2000 K (1627 to 1727 °C), with occasional temperatures up to 2400 K (2127 °C). Standard Fourier image reconstruction technique which is capable of handling data from systems of arbitrary shape was employed. The flame in the work of Ref. [33] was cylindrically symmetric. The computer program published by Shepp and Logan [32] was used by applying the reconstruction one wavelength at a time to determine spatially resolved spectra.

The tomographic reconstruction process enhances noise in the spatial resolved data, relative to that in the line-of-sight data. In both works of Ref. [31] and Ref. [33] the data were smoothed by co-adding data from eight adjacent wave number bands. That resulted in degraded resolution from the 8 cm<sup>-1</sup> used, although still sufficient to quantitatively measure the gas species. The results of the reconstruction were consistent in that the summed absorbance and radiance across the flame diameter agreed with the measured line-of-sight absorbance and radiance.

A self-absorption correction as described in [31] was employed in the works of Ref. [31] and Ref. [33]. The self-absorption was, in fact, taken into account as an approximation in the equation of radiative transfer to simplify the latter. However, the radiative transfer equation could be used directly in the equations for tomographic reconstruction without approximating the self-absorption part as it was done in this PhD project.

In both works of Ref. [31] and Ref. [33] the emission and transmission measurements were made along the same 1 mm wide by 4 mm high optical path defined with apertures. With this optical geometry, parallel line-of-sight emission and transmission spectra were collected across the flame at 1 mm increments along the radius. The vertical spatial resolution was chosen to enhance the IR signal to system noise. The spatially resolved "point" values correspond to an average within 1 mm  $\times$  1 mm  $\times$  4 mm high volumes. The instruments used were a Nicolet models 7199 and 20 SX, equipped with a globar source for transmission measurements, and utilizing mercury cadmium-telluride detectors for both emission and transmission measurements [34, 35]. Mirrors were used in the optical setup (as shown in Fig. 1 in Ref. [35]) to arrange the beams for the emission and transmission measurements. Emission measurements were made with the movable mirror in place (see Fig. 1 in Ref. [35]). Transmission measurements were made with the movable mirror removed.

Bates et al. [36] described an experimental technique that combined tomography, Hadamard signal encodement, and a patented FTIR emission/transmission technique to perform simultaneous spatially resolved gas species and soot measurements during combustion. Tomographic analysis of line-of-sight FTIR data allowed spatially resolved measurements to be made. Hadamard encodement of the tomographic sections increased the overall signal throughput, improving the signal-to-noise ratio for each measurement. The Hadamard technique led to a major simplification in the tomographic apparatus in that the scanning apparatus that would normally be required was eliminated, and focusing of the IR light was much easier. The FTIR Hadamard tomography was performed to measure soot in a fuel-rich diffusion flame. Spatially resolved concentration measurements agreed well with previous data, and clearly showed striking three-dimensional features that could not normally be measured by simple line-of-sight techniques.

Yousefian et al. [37] retrieved the temperature and species distributions in semi-transparent gaseous axisymmetric objects by carrying out the inversion of their directional low-resolution spectral transmission and/or spectral emission measurements in the IR range by using methods based on the solution of the radiative transfer equation. The validity domains of the hypothesis on which the low resolution inversion methods are based were discussed by comparison with the corresponding exact radiative properties generated by line-by-line descriptions. A propane-air laminar premixed flame was experimentally studied. The diameter of the burner was 4 cm. The IR data were collected for the 4.3  $\mu\text{m}$  band of  $\text{CO}_2$ , either by a directional scanning (at a fixed frequency) or by a frequency scanning (in a fixed direction). Depending on the data acquisition methods, two different reconstruction techniques were used. For spatial scanning, a generalized Abel transformation or the regularized Murio's method, was utilized. For spectral scanning the recovered results were carried out by the Chahine-Smith method. For both techniques, profile reconstructions were worked out accounting for a spectral data bank based on the narrow statistical band model.



In the optical system used in the work of Ref. [37] several pin holes (of diameters 2 and 3 mm) determined a fixed line-of-sight. By translating repeatedly the flame in a horizontal plane, a set of emission or transmission measurements was acquired. The vertical displacement allowed the study of the flame for different heights. The spatial resolution was defined by a 1 mm slit placed in front of the source. Radiation from the blackbody or from the flame was directed to an FTIR spectrometer by means of a parabolic off-axis mirror, followed by a plane mirror close to the spectrometer entrance. The FTIR spectrometer (Perkin Elmer 1760X) was equipped with a triglycide sulfate detector. Before the mathematical inversions, the experimental data sets were preconditioned in two steps, (1) by symmetrization with respect to the position of the line-of-sight and (2) by filtering out noisy data by a Fourier analysis.

The absorption coefficient profiles of CO<sub>2</sub> and propane were retrieved from the set of experimental directional transmittivity results by the preconditioned Abel transform, the Abel-Simonneau method, and the conjugate gradient method [37]. Once the absorption profiles are known for each altitude they can be introduced in the emission, then by means of a second Abel inversion, the unknown temperature profiles can be obtained [37]. The reconstructed temperature profiles were also compared to thermocouple measurements (carried out with a 50- $\mu$ m diameter 6-30 Pt/PtRh, corrected for radiative exchanges and conductive losses) [37]. Concentration profiles of CO<sub>2</sub> were also obtained in work [37]. The results were compared with those predicted by the propane-air combustion thermodynamic model for the experimental equivalence ratio of 0.92.

In work [37], the errors in the emission and transmission measurements were evaluated to be  $\pm 1.5\%$ . They involved an error of  $\pm 2\%$  on the physical model. Accordingly, by a simulation the uncertainties introduced in the temperature reconstruction process were evaluated in the extreme unfavourable case to be  $\pm 7\%$  near the symmetry axis and  $\pm 11\%$  toward the flame edge. As for the CO<sub>2</sub> distribution the error was  $\pm 9\%$  on the axis and  $\pm 14\%$  for the edge.

As reported in Ref. [37], the Chahine-Smith method of inversion used for emission data to reconstruct the CO<sub>2</sub> temperature and concentration profiles gave only poor results when associated with the statistical narrow band model mean parameters. Thus its handling advantages, requiring only spectral scanning in a single line-of-sight, were offset by less accurate results and less adaptability when high temperature and stiff concentration gradients were present, as was the case near the flame front [37].

Correia et al. [38, 39, 40] reported the development, implementation and test of a laboratory flame 3D temperature sensor which allowed for the tomographic reconstruction of the temperature field, based on the flame radiation emitted at 800 nm and 900 nm. The work of Refs. [38, 39, 40] aimed to contribute to overcome the difficulties on the application of tomographic techniques to large scale combustion systems by adopting new strategies for flame data collection and processing.

In works [38, 39, 40], the image acquisition system comprised several pairs of IR cameras that were placed on the platform, obtaining radial views of the flame. The platform allowed for changing of the position of the cameras, namely their distance to the flame and the angle between the sets of cameras. IR interference filters (Melles Griot 800 and 900 nm filters) were used for the acquisition of the monochromatic flame images. The resolution of the flame images was  $560 \times 760$  pixel. Each pixel corresponded to a  $0.6 \times 0.6$  mm square, at the flame location. A blackbody furnace was used to calibrate the cameras on the selected wavelengths [38, 39, 40]. The local soot particle tem-

peratures were quantified using the system and the results for both the tomography and the pyrometry calculations were compared with the thermocouple results [38, 39, 40].

Barrag and Lawton [41] used computer optical tomography as a diagnostic technique in the study of a premixed methane-air diffusion turbulent flame of equivalence air/fuel ratio of 0.5. Visible light from a tungsten filament lamp at a wavelength of  $0.65\ \mu\text{m}$  was used. The test plane, of 12 mm diameter, of a flame was scanned using eleven equi-spaced parallel optical paths which were rotated through 180 deg and a complete set of readings were taken at 30 deg intervals. The optical thickness and emission ratio were measured along each path and at each 30 deg interval. A computer program based on the convolution method combined with the Shepp-Logan filter [32], was used to reconstruct a two-dimensional image of the absorption coefficient, emission function, temperature distribution and soot concentration at a chosen instant.

In the optical tomography setup of Ref. [41], the light was coming from a backlight source (which was a laser or tungsten ribbon lamp). The light was switched on and off with a chopper plate which was a rotating plate with suitable openings to provide emission/transmission measurements. The chopped light passed through the flame and a  $0.65\ \mu\text{m}$  optical filter and then it was detected by a photomultiplier tube which converted light intensity into voltages. The system was calibrated using a laminar flame in which the temperature was measured and compared using a thermocouple method as well as a tomographic method. The two methods were coincident within the range of 3% [41].

The developed in work [41] tomography techniques were also applied for tomographic reconstruction of the 2D images of gas temperature, optical thickness and soot concentration from 8 line-of-sight emission/transmission measurements on a one-cylinder internal combustion engine by employing an optical plate with 16 quartz windows mounted between engine cylinder head and engine cylinder block [42].

A simulation work is reported by Liu and Man [43] where the multi-wavelength inversion method was extended to reconstruct the time-averaged temperature distribution in a non-axisymmetric turbulent unconfined sooting flame by the multi-wavelength measured data of low time-resolution outgoing emission and transmission radiation intensities. Gaussian,  $\beta$  and uniform distribution probability density functions were used to simulate the turbulent fluctuation of temperature, respectively. The reconstruction of time-averaged temperature consisted of three steps. First, the time-averaged spectral absorption coefficient was retrieved from the time-averaged transmissivity data by an algebraic reconstruction technique. Then, the time-averaged blackbody spectral radiation intensity was estimated from the outgoing spectral emission radiation intensities. Finally, the time-averaged temperature was approximately reconstructed from the multi-wavelength time-averaged spectral emission radiation data by the least-squares method. Noisy input data were used to test the performance of the proposed inversion method. The results showed that the time-averaged temperature distribution can be estimated with good accuracy, even with noisy input data. As reported in Ref. [43], the accuracy of the estimation decreases with the increase of turbulent fluctuation intensity of temperature and the effects of assumed probability density function on the reconstruction of temperature are small.

### 1.4.2. Combustion Related Measurements on Engines

The regulations on NO<sub>x</sub> emissions from marine diesel engines toughen the limits on emissions every 5-10 years [44]. E.g. for engines constructed on or after 1 January 2011 the total weighted cycle emission limit is 14.4 g/kWh at engine's rated speeds less than 130 rpm whereas for those constructed on or after 1 January 2016 the limit will be 3.4 g/kWh at the same engine's rated speeds [44], i.e. the limit is toughened by a factor of about 4.

It is therefore important to develop fast and efficient methods of emission measurements which would be useful in the phases of engine testing at the engine builder, would facilitate a means to estimate the sensitivity of engine emissions on changes in engine operation parameters (fuel injection profile, exhaust valve operation, turbo-charger turbine area, bypass, etc.). Knowledge on gas temperature is essential for NO<sub>x</sub> emission measurements by UV spectroscopy techniques which are promising for NO<sub>x</sub> emission measurements and have great potential for application to full-scale engines [9, 10].

Therefore, a methodology of fast direct optical measurements of gas temperature by IR spectroscopy was developed and tested in this part of the PhD project, by which gas temperature measurements can be conducted for individual cylinders and with fast temporal response.

Recently the research on engine emissions has mainly been focused on determining the emissions from ocean going ships [45, 46, 47, 48, 49, 50, 51]. Species under consideration include NO<sub>x</sub>, CO<sub>2</sub>, CO, SO<sub>2</sub>, particulate matter, elemental carbon, organic carbon, hydrated sulphate, ash and others.

As discussed above, a special attention is paid to NO<sub>x</sub> emissions from internal combustion engines (ICEs) [52]. Often, standard engine rooms on ships do not have measurement equipment to analyze the exhaust gases [53, 54]. Therefore Kowalski and Tarelko [53, 54] proposed a method to estimate the NO<sub>x</sub> emissions without direct measurement, based on the measurements of working engine parameters.

Verbiezen et al. [55] reports the in-cylinder local experimental NO concentrations obtained by laser-induced fluorescence (LIF) measurements in a six-cylinder, heavy-duty diesel truck engine with one of its cylinders made optically accessible via fused silica windows in the piston, cylinder wall and cylinder head.

The thickness of pressure-resistant fused silica windows, however, usually prevents the operation of the neighboring cylinders and changes in-cylinder heat transfer considerably due to the extended fused silica surfaces [52]. In eliminating this drawback, Reichle et al. [56] exploited UV tracer-based LIF (TLIF) of organic fuel compounds for the acquisition of in-cylinder data in the area close to the ignition spark by using a micro-optical probe with endoscope optics and fibers fitted inside a spark plug in a production engine. The excitation laser and the spectroscopic analysis system for the tracer fluorescence were connected to the spark plug via optical fibers making the application of the system easier.

In general, TLIF techniques have recently been applied extensively to optically accessible single-cylinder ICEs offering high temporal and spatial resolution in case of point measurements as well as capable of measuring the 2D profiles of temperature, exhaust gas residuals, fuel concentration, equivalence ratio and other combustion parameters [57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68].

It is however difficult and costly to apply laser-based techniques to industrial engines due to requirement of extensive optical access, highly accurate alignment of the

optical setup and high price of the laser equipment. Also, TLIF needs a tracer to be seeded into the intake air and/or fuel, allows only certain types of fuel and is critical to the purity of the fuel and tracer. That limits significantly the industrial application.

Also, in some research work the imaging is performed without adding fuel into the combustion chamber [59] or before the combustion process, i.e. in unburnt gases [60, 61]. The maximum measured temperatures are limited to about 627 °C (900 K) [62] or in some works to the range of about 27-477 °C (300-750 K) [63, 64].

A similar laser-based technique, the two-line atomic fluorescence (TLAF), was applied to an optically accessible modified production engine [65, 66]. This technique is however limited to measuring the burned gas temperatures. A range of 527-2527 °C (800-2800 K) is reported.

Orth et al. [67] and Schulz et al. [68] employed a laser setup similar to that from Refs. [65, 66] mentioned above but 2D Rayleigh scattering signal was measured to obtain in-cylinder 2D distributions of temperatures in a transparent single-cylinder engine. Temperatures were measured over the entire combustion cycle covering a range beyond 207-2127 °C (480-2400 K). Using Rayleigh scattering, problems might occur in realistic engine geometries due to strong elastic scattering of surfaces [62]. Also, since Rayleigh signal is a function of a species dependent scattering cross-section this method is not applicable to non-homogeneously mixed systems where local effective Rayleigh cross-sections are unknown [62]. Furthermore, very large laser powers are required that can perturb the flame chemistry [65].

Some applications of optics-based techniques to near-production engines are reported including the following.

Barrag and Lawton [42] tomographically reconstructed the 2D images of gas temperature, optical thickness and soot concentration from 8 line-of-sight emission/transmission measurements performed on a one-cylinder ICE by employing an optical plate with 16 quartz windows mounted between engine cylinder head and engine cylinder block.

Similarly, Wright et al. [21] tomographically reconstructed in-cylinder relative fuel distributions from 27 line-of-sight fast dual wavelength ratiometric transmission measurements using a unique optical access layer carrying embedded optical fibres and collimators implemented in one cylinder of an otherwise standard 4-cylinder ICE engine. Two laser sources and a photodiode receiver were employed for each line-of-sight.

This part of the PhD project concerns first application of fiber-optics-based IR spectroscopy techniques to a large Diesel engine (2-stroke 4-cylinder test marine Diesel engine 4T50ME-X at MAN Diesel & Turbo, Copenhagen, see Sec. 2.1.1, Fig. 26). The exhaust duct of one of the cylinders was slightly modified (by MAN Diesel & Turbo) by making three optical access ports in it to allow three line-of-sight optical measurements across the exhaust gas stream coming from the exhaust valve of the cylinder.

Fast time-resolved transient IR spectroscopy measurements in the three optical ports of the exhaust duct of the large Diesel engine were performed for the first time. The exhaust gas temperature was obtained as the brightness temperature of the intensity of the 4.3  $\mu\text{m}$  CO<sub>2</sub> band simultaneously at the three ports from the simultaneous IR measurements performed by the **multichannel IR spectrometer system** (a first prototype of the 2Dt IR spectroscopic tomography system) developed also in this part of the PhD project.

## 2. Experimental Details

### 2.1. Multichannel IR Spectrometer System

#### 2.1.1. Introduction and Remarks

Simultaneous spectral measurements at several line-of-sights are essential for on-line time-resolved 2D tomographic measurements of gas temperature in a cross-section of a flame or hot flue gas.

A **multichannel IR spectrometer system** (a first prototype of the 2Dt IR spectroscopic tomography system) has been developed in this part of the PhD project for simultaneous fast time-resolved transient IR spectral measurements at several line-of-sights.

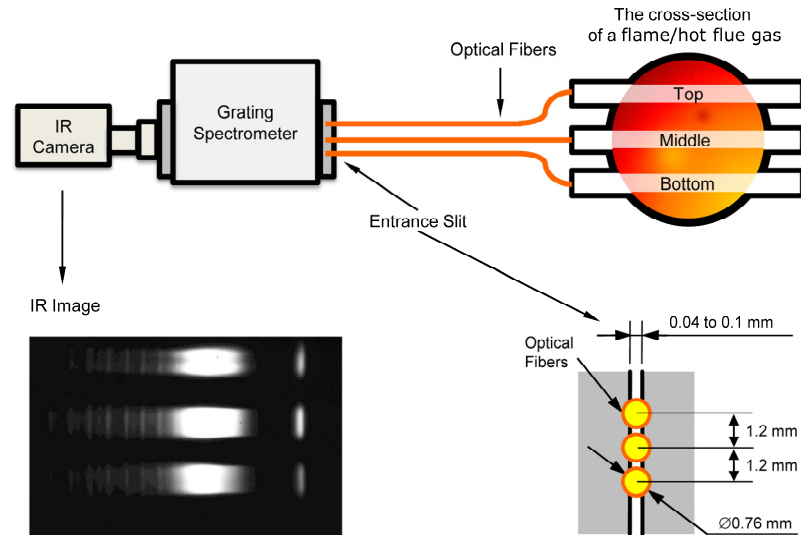
The development of the system was also relevant to the EU-funded HERCULES-B project “High-efficiency engine with ultra-low emissions for ships” carried out under 7<sup>th</sup> Framework Program “Sustainable Surface Transport” (grant agreement SCP7-GA-217878) presented in reports [9], to the project Energinet.dk ForskEL projektnr. 2008-1-0079 “Fast optical measurements and imaging of flow mixing” reported in Ref. [11], and to the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8].

The system was developed with a view to application for simultaneous fast exhaust gas temperature measurements in the three optical ports of the exhaust duct (Sec. 2.4) of a large Diesel engine (Sec. 2.4.2, Fig. 26) as stated in the objectives of the PhD project (Sec. 1.3.2). The in-depth description of the system presented in this part of the PhD thesis extensively refers to that application on the engine.

#### 2.1.2. General Description of the System

The system employs a grating spectrometer (Princeton Instruments, Acton SP2150, 0.150 m Imaging Dual Grating Monochromator/Spectrograph) and an IR camera (CEDIP Infrared Systems, Titanium 560M, InSb detector, 640×512 pixels), Fig. 1. The IR camera is focused onto the rear focal plane of the spectrometer giving an IR image of the spectra. The spectra can be retrieved from the IR images using dedicated software (see also Fig. 2 and associated discussion in Sec. 2. 1. 3).

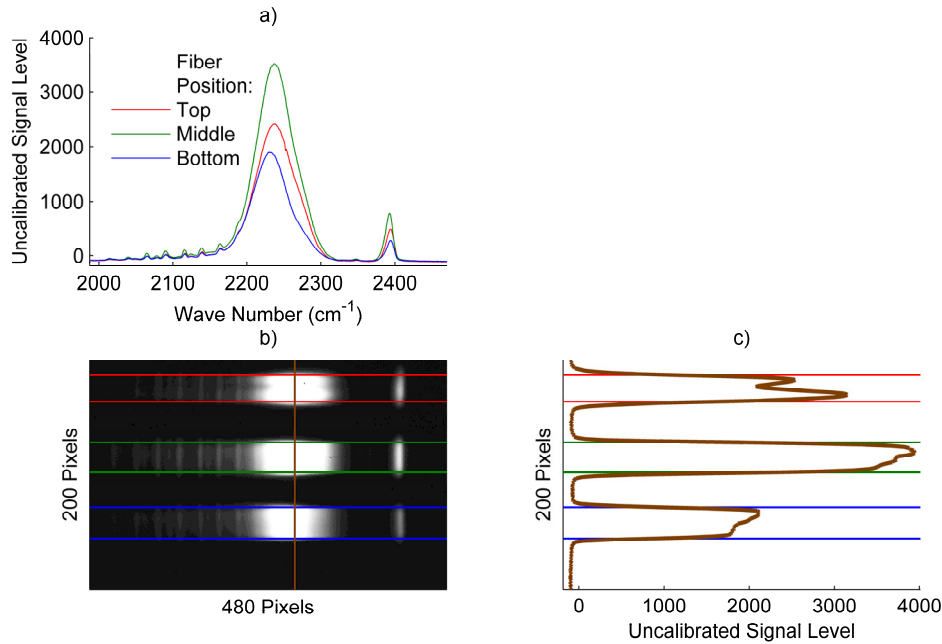
Three optical fibers (chalcogenide IR-glass fibers) were mounted onto the entrance slit of the spectrometer to allow simultaneous spectral measurements at the three optical ports of the exhaust duct. In general, up to 7 fibers can be mounted onto the entrance slit of the spectrometer (See Fig. 4 and associated discussion in Sec. 2. 1. 5). For the application on the large Diesel engine where only three optical ports in the exhaust duct of the engine were available (Sec. 2.4.2, Fig. 26), three fibers were sufficient.



**Figure 1.** The **multichannel IR spectrometer system** (a first prototype of the 2Dt IR spectroscopic tomography system). Here the cross-section of the exhaust duct (conducting hot flue gas and having three optical access ports) of the large Diesel engine (Sec. 2.4.2, Fig. 26) is intended as an example of “the cross-section of a flame/hot flue gas” shown in the upper-right part of the figure.

### 2.1.3. Retrieval of IR Spectra from IR Images

The schematic of the retrieval of IR spectra from the IR images is shown in Fig. 2. The IR image is shown in panel (b). Looking in the horizontal direction, a profile along a marked vertical line can be seen as shown in panel (c). Looking in the vertical direction, three profiles along the horizontal dimension can be seen as shown in panel (a). Each profile color corresponds to the position of one end of a fiber at the slit of the spectrometer and also to the position of the other end of the fiber at the port of the exhaust duct of the large Diesel engine (Sec. 2.4.2, Fig. 26) (top position - red, middle position - green, bottom position - blue). Each profile shown by respective color is an average of all the horizontal profiles located between the horizontal lines of respective color. These lines are also shown in panel (c) to indicate the borders of the averaging when looking in the horizontal direction.



**Figure 2.** a) IR spectra obtained as profiles along the horizontal dimension of the IR image (shown in panel (b)) averaged within the marked horizontal lines (shown using respective colors for each fiber position); b) IR image; c) profile along the marked vertical line on the IR image.

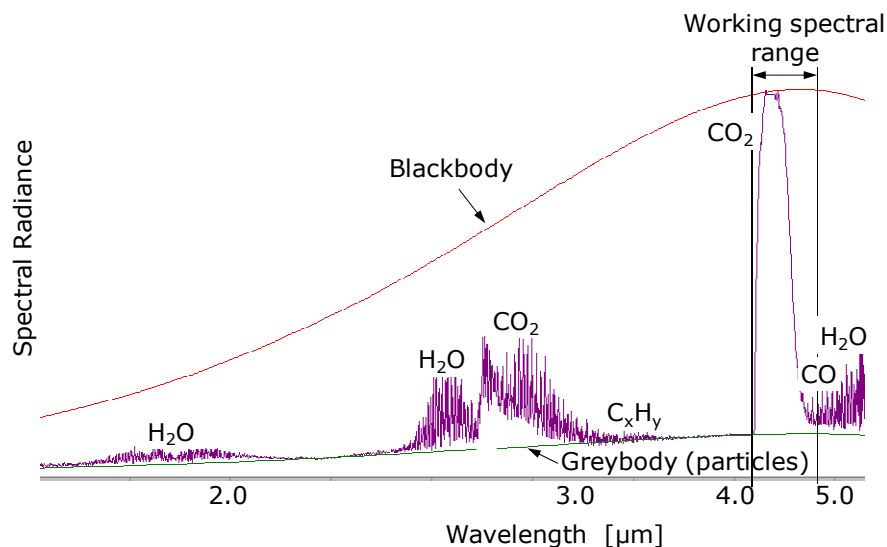
The widths for each averaging region (as shown by horizontal lines of respective color on the IR image for each fiber position) were defined by balancing between the reducing the random error and maximizing the signal-to-noise ratio. Making the region wider corresponds to taking the mean value of more measurements and hence increasing the precision by minimizing the random error. At the same time, as can be seen from panel (c), if the averaging region is too wide then there is more contribution to the profiles from the pixels detecting no useful signal and hence that adds more noisy values to the resulting profiles and hence decreases the signal-to-noise ratio. The horizontal profiles in panel (a) are, in fact, IR spectra. See also Sec. 2.1.6 and Sec. 2.1.8 for the discussion about the calibration of the x- and y-axes of the plots of IR spectra.

#### 2.1.4. Working Spectral Range

The IR region has been chosen as the working spectral range of the system because major combustion species  $\text{H}_2\text{O}$  and  $\text{CO}_2$  have strong fundamental rotational-vibrational bands in the IR region [69]. IR Camera's InSb array allows detection in the range  $1.0\text{-}5.9\ \mu\text{m}$  ( $1700\text{-}10000\ \text{cm}^{-1}$ ). The optical components of the system include a long wave pass filter Spectrogon LP-3300nm in order to cut the higher order wavelengths reflected by the grating at the same angles as the working order wavelengths. Furthermore, the system was optimized to work in the range  $4.2\text{-}4.8\ \mu\text{m}$  ( $2080\text{-}2400\ \text{cm}^{-1}$ ) which covers the  $4.3\ \mu\text{m}$  band of  $\text{CO}_2$ . A typical IR emission spectrum of a hot flue gas (obtained by an FTIR spectrometer having a wider working spectral range) containing spectroscopic features of major combustion species ( $\text{CO}_2$ ,  $\text{CO}$ ,  $\text{H}_2\text{O}$ ,  $\text{C}_x\text{H}_y$ ) and the working spectral range of the **multichannel IR spectrometer system** are shown in

Fig. 3. It should be noted that the working spectral range window (shown in Fig. 3) corresponding to the working spectral range of the system can, in principle, be “moved” all across the shown spectral range of 1.0-5.9  $\mu\text{m}$  ( $1700\text{-}10000\text{ cm}^{-1}$ ) (which is the range of the IR Camera’s InSb array).

An example of the spectrum obtained by the system itself is shown in Fig. 5 (see Sec. 2.1.6 for details). The range of the wavelengths shown in Fig. 5 is the working spectral range of the system. It should be noted that only the narrow region shown by the vertical pale orange strip in Fig. 5 is the optimal region for gas temperature calculations and hence it was used for gas temperature calculations (in the application on the large Diesel engine) (Sec. 3.2.3, Sec. 3.2.4).



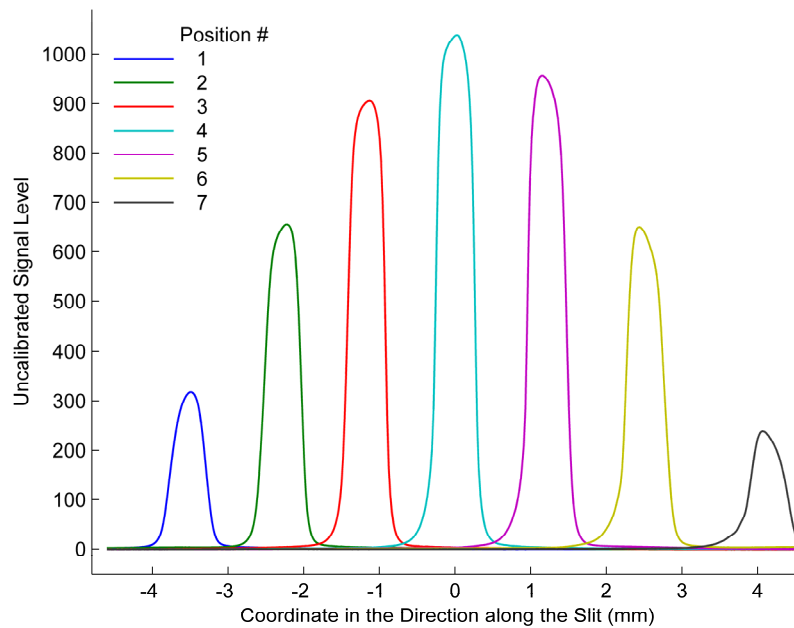
**Figure 3.** A typical IR emission spectrum of a hot flue gas (obtained by an FTIR spectrometer having a wider working spectral range, shown for information) and the working spectral range of the **multichannel IR spectrometer system** ( $4.2\text{-}4.8\text{ }\mu\text{m}$  or  $2080\text{-}2400\text{ cm}^{-1}$ ). See Fig. 5 (Sec. 2.1.6) for an example of the spectrum obtained by the system itself.

### 2.1.5. Number of Fibers Coupled onto the Entrance Slit and Time Resolution

The time resolution of the system is defined by the frame rate of the camera which in turn depends on the dimensions of the frame. The latter are determined by the number of optical fibers coupled onto the slit and the distance between them. The distance between the fibers is defined, on one hand, from the condition of minimal cross-talk between the signals from the adjacent fibers and, on the other hand, from the condition of minimal IR image width (and hence, of maximal frame rate).

The signal from a fiber at 7 positions in front of the entrance slit of the spectrometer is shown in Fig. 4. The horizontal axis is the coordinate along the direction defined by the entrance slit of the spectrometer. Each curve is an average of the curves along the horizontal dimension of the IR Image (shown in Fig. 1 in which the signal from only the three central positions is shown) at each fiber position.





**Figure 4.** The signal from a fiber at 7 positions in front of the entrance slit of the spectrometer. Each curve is an average of the profiles along all the vertical lines in the IR image for each fiber position.

It can be seen that the three central positions yield the highest signal level. The crosstalk level does not exceed 0.4% for each of those three positions within  $\pm 0.2$  mm from the axis of symmetry of each curve. The distance between the symmetry axes of curves 3 and 4 is 1.13 mm and the one between curves 4 and 5 is 1.17 mm.

As mentioned above, three optical fibers were mounted onto the entrance slit of the spectrometer to allow simultaneous spectral measurements at the three optical ports of the exhaust duct of the Large Diesel engine (Sec. 2.4). To guarantee the minimal crosstalk, the distance between the fiber ends at the entrance slit was 1.2 mm (see Fig. 1).

The IR image was trimmed such that the signal from only the three central positions could be detected to maximize the frame rate of the camera. At such dimensions of the IR image, the camera can be run at up to 500 frames per second (500 Hz) providing the temporal resolution for simultaneous IR spectral measurements of down to 2 ms. It should be noted that, the external trigger signal was used to synchronize the acquisition of the IR images with the piston crank angle and reference time points. The maximum frame rate of the camera was sufficient to guarantee the synchronization with the trigger signal at the frequency of the latter.

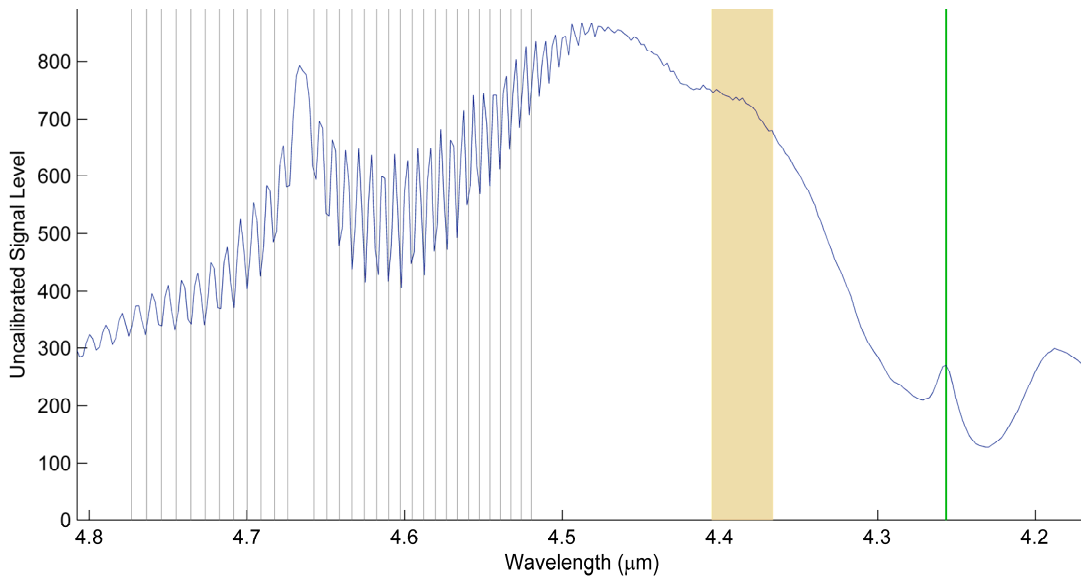
#### 2.1.6. Calibration of the x-axis (wavelength axis)

The calibration of the x-axis is necessary to establish the correspondence between the pixel numbers along the horizontal dimension of the IR image and the wavelengths (or wave numbers). Since the system has a dispersive element which is a reflection grating, the correspondence was established for wavelengths. The coordinate at the

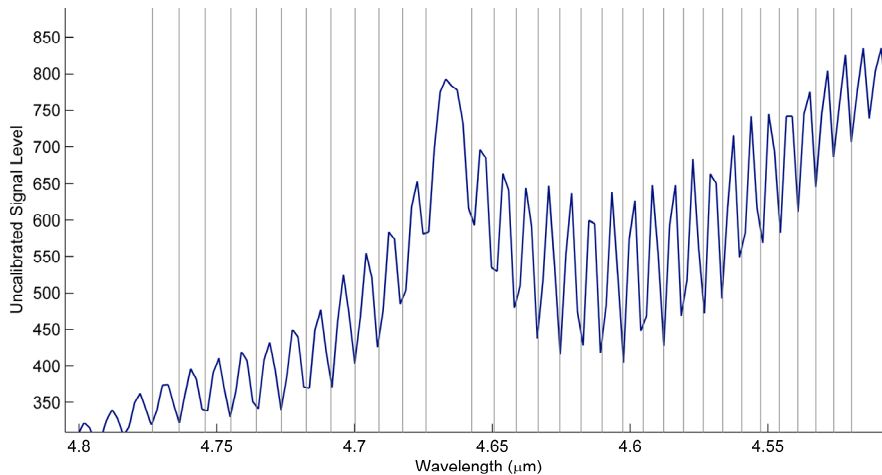
rear focal plane of the spectrometer in the plane perpendicular to the surface of the grating, or, alternatively, a pixel number along the horizontal dimension of the IR image, is expected to depend linearly on the wavelength of the incident light. Hence a linear correspondence was established between the pixel number and the wavelength.

The positions of several strong CO absorption lines were used to obtain reference points for the linear dependence between the pixel numbers and wavelengths. The least squares method was employed to calculate the coefficient and the constant of the linear dependence from several reference points.

A cell containing 100% CO was inserted between the the entrance slit of the spectrometer and the source of IR radiation to obtain a signal having the CO absorption features. An example of the latter is shown in Fig. 5. A CaF<sub>2</sub> lens (focal length 50.8 mm) was used to focus the light onto the entrance slit of the spectrometer instead of the optical fiber in order to maximize the signal-to-noise ratio and to avoid strong absorption from the fiber which takes place in the region around 4.3 μm. In Fig. 5, the sharp rotational fine structure of the 4.7 μm rotational-vibrational band of the primary isotopologue of CO (<sup>12</sup>C<sup>16</sup>O) can clearly be observed in the signal (blue curve). Corresponding line positions of the band were taken from Ref. [69, p. 70] and are shown in the figure by the gray vertical lines (See also Fig. 6 where the zoom on the region of strong CO absorption of Fig 5 is shown). The pixel numbers of the sharp edges (looking downwards) in the experimental curve and the reference values of the corresponding CO absorption line positions were used in the least squares algorithm to obtain a linear dependence between the pixel numbers and the wavelengths, or, in other terms, to calibrate the x-axis.



**Figure 5.** The signal (blue) obtained when the CO cell was inserted between the entrance slit of the spectrometer and the source of IR radiation. The range of the wavelengths shown in the figure is the working spectral range of the system (see Sec. 2.1.4). The positions of the CO absorption lines (the rotational fine structure of the 4.7 μm rotational-vibrational band of <sup>12</sup>C<sup>16</sup>O) are shown by the vertical gray lines and were taken from Ref. [69, p. 70]. The origin of the 4.3 μm band of CO<sub>2</sub> is shown by the vertical green line and was taken from Ref. [69, p. 74]. The narrow region used for gas temperature calculations (in the application on the large Diesel engine, Sec. 2.4) is shown by the vertical pale orange strip.



**Figure 6.** A zoom in Fig 5 on the region of strong CO absorption.

The origin of the 4.3  $\mu\text{m}$  band of  $\text{CO}_2$  can also be observed in Fig. 5. Its position is shown by the vertical green line and was taken from Ref. [69, p. 74]. It was not used for the calibration of the x-axis as an additional reference point to those obtained from the CO absorption. On the contrary, it was used to estimate the accuracy of the calibration. The position of the 4.3  $\mu\text{m}$   $\text{CO}_2$  band origin was estimated on the calibrated x-axis and compared to the reference value obtained from Ref. [69, p. 74]. The relative error is 0.008 %. This value also, in fact, includes the contribution of some nonlinearity in the relationship between the pixel numbers and the wavelengths. It can be concluded that this contribution is negligible. In general, it can be concluded that contribution of the x-axis calibration error to the total measurement error is negligible.

It should be noted that the value of 0.008 % is applicable for the version of the optical setup as described in the current section (2.1.6). As will be shown in the next section (2.1.7), the spectral resolution for the version of the optical setup as described in the beginning of the current section is 9 nm ( $3.9 \text{ cm}^{-1}$ ). For the version of the optical setup used for the simultaneous measurements on the exhaust duct when the three fibres are coupled onto the entrance slit of the spectrometer (Sec. 2.1.2, Fig 1), the resolution for all the three fibres is 23 nm ( $= 12 \text{ cm}^{-1}$ ). A lower resolution decreases the quality of the x-axis calibration since it is important that the CO absorption features can clearly be observed in the signal from the CO cell. The x-axis calibration error for this case was observed to be within 0.8 %. However, this error still brings negligible contribution to the total measurement error compared to the other errors considered in the following sections.

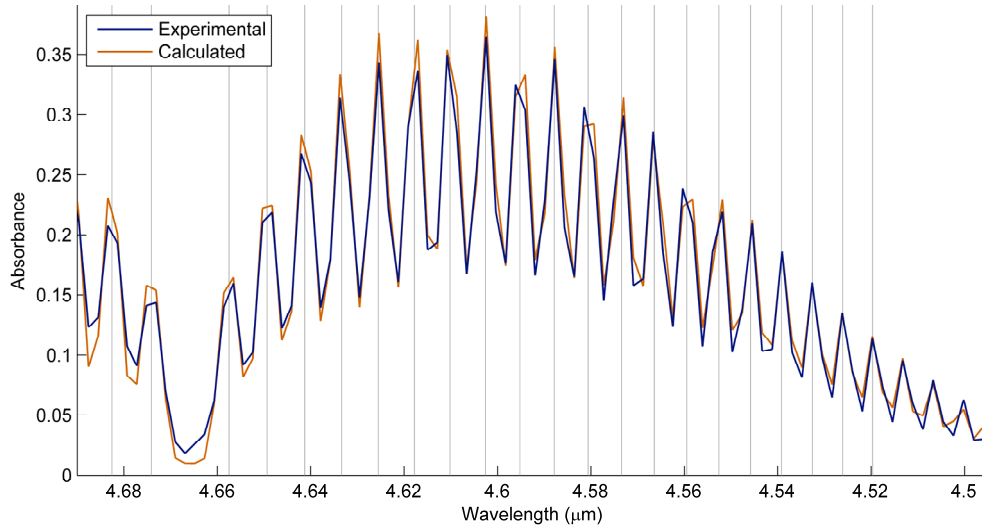
### 2.1.7. Instrument Line Shape Function and Spectral Resolution

Again, the measurements with the CO cell (see also Sec. 2.1.6) were used to analyze the instrument line shape (ILS) function of the system. The absorbance of the 4.7  $\mu\text{m}$  band of CO (100 %) at room temperature (296 K) was measured using the CO cell and compared to the calculated absorbance using the HITRAN-2008 database [70]

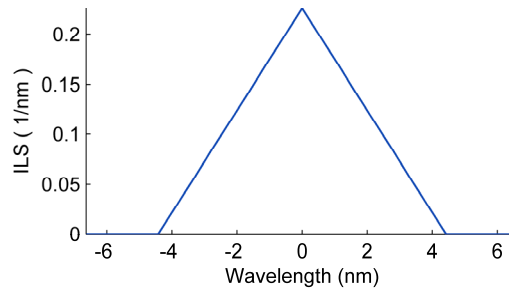
and applying the triangular instrument line shape function and a spectral resolution  $R = 9 \text{ nm}$  ( $3.9 \text{ cm}^{-1}$ ) (Fig. 7). The system is well described by triangular ILS (Fig. 8) which is given by

$$\text{ILS}(\lambda - \lambda_0) = \frac{1}{0.5R} \left( 1 - \frac{|\lambda - \lambda_0|}{0.5R} \right), \quad |\lambda - \lambda_0| \leq 0.5R \quad (1)$$

where  $\lambda$  is any wavelength in a spectrum [nm],  $\lambda_0$  is any wavelength observed by the system [nm],  $R$  is the spectral resolution of the system [nm].



**Figure 7.** Absorbance of the  $4.7 \text{ } \mu\text{m}$  band of CO (100 %) at room temperature (296 K): blue - measured using the CO cell containing 100 % CO (the experimental details are given in Sec. 2.1.6); red - calculated using the HITRAN-2008 database [70] and applying the triangular instrument line shape function (Fig. 8) and a value of spectral resolution  $R = 9 \text{ nm}$  ( $3.9 \text{ cm}^{-1}$ ). As in Fig. 5, vertical gray lines show the absorption line positions of the  $4.7 \text{ } \mu\text{m}$  band of  $^{12}\text{C}^{16}\text{O}$ .



**Figure 8.** The instrument line shape function of the system as given by Eq. (1). The value of spectral resolution  $R = 9 \text{ nm}$  ( $3.9 \text{ cm}^{-1}$ ) was applied. This resolution was found for the version of the optical setup as described in Sec. 2.1.6.

For the version of the optical setup as described in Sec. 2.1.6 (when the lens is used to focus the light onto the entrance slit of the spectrometer), the value of  $R$  is  $9 \text{ nm}$  ( $3.9 \text{ cm}^{-1}$ ). For the version of the optical setup used for the simultaneous measurements on the three ports of the exhaust duct when the three fibres are coupled onto the en-

trance slit of the spectrometer (Sec. 2.1.2, Fig 1), the resolution of the system for all the three fibres is  $R = 23 \text{ nm}$  ( $12 \text{ cm}^{-1}$ ). It should be noted that the resolution varies slightly from fiber to fiber however this variation has negligible effect on the temperature measurements.

For the version of the optical setup as described in Sec. 2.1.6, the width of the entrance slit of the spectrometer was set to  $40 \text{ }\mu\text{m}$ . The spectral resolution of a grating spectrometer is becoming higher with narrowing the entrance slit [71, p. 63-67] however the signal-to-noise ratio is decreasing. For the version of the optical setup used for the simultaneous measurements on the exhaust duct (Sec. 2.1.2, Fig 1), the optimal width of the entrance slit was found to be  $100 \text{ }\mu\text{m}$ . It is clear that the slit width for the latter case is 2.5 times larger than for the former case. Accordingly, the spectral resolution  $R$  for the latter case ( $23 \text{ nm}$ ) is about 2.56 times lower than the one for the former case ( $9 \text{ nm}$ ).

### 2.1.8. Calibration of the y-axis (spectral radiance axis)

Calibration of the y-axis is necessary to establish the correspondence between the uncalibrated signal levels (the raw values on the y-axis of Fig. 2, panel (a)) and spectral radiance. Once the widths of each averaging region (shown by horizontal lines in panels (b) and (c) of Fig. 2) have been defined and the x-axis has been calibrated, the response function of the system as a function of wavelength  $\rho(\lambda)$  can be found from the reference measurements of the blackbody radiation at two temperatures provided the spectral measurements performed by the system can be described by the following linear model:

$$S(\lambda) = \rho(\lambda)L(\lambda) + \beta(\lambda), \quad (2)$$

where  $S(\lambda)$  is a received signal from a source [uncalibrated signal level] at some observed wavelength  $\lambda$  [ $\mu\text{m}$ ];  $L(\lambda)$  is unknown spectral radiance of the source [ $\text{W} / \text{m}^2 \text{ sr m}$ ];  $\rho(\lambda)$  is a response function of the system [uncalibrated signal level]/[ $\text{W} / \text{m}^2 \text{ sr m}$ ]; and  $\beta(\lambda)$  is constant background radiation [uncalibrated signal level] detected when no source of radiation is available in the field of view of the system, i.e. when  $L(\lambda) = 0$  or  $L(\lambda)$  is negligibly small.

The response function  $\rho(\lambda)$  can then be found as

$$\rho(\lambda) = \frac{S_2(\lambda) - S_1(\lambda)}{L_{BB}(\lambda, T_2) - L_{BB}(\lambda, T_1)}, \quad (3)$$

where  $S_1(\lambda)$  and  $S_2(\lambda)$  are received signals from a black body radiation source which was set to temperatures  $T_1$  and  $T_2$  respectively; and  $L_{BB}(\lambda, T_1)$  and  $L_{BB}(\lambda, T_2)$  are the spectral radiances of the black body calculated from the Planck radiation law [72, Eq. (2.15)], [73] for temperatures  $T_1$  and  $T_2$  respectively. The background radiation can be found as

$$\beta(\lambda) = S_1(\lambda) - \rho(\lambda)L_{BB}(\lambda, T_1). \quad (4)$$

The measured signal  $S(\lambda)$  can then be converted to  $L(\lambda)$  having the units of spectral radiance as

$$L(\lambda) = \frac{S(\lambda) - \beta(\lambda)}{\rho(\lambda)}, \quad (5)$$

or, directly from  $S_1(\lambda)$ ,  $S_2(\lambda)$ ,  $L_{BB}(\lambda, T_1)$  and  $L_{BB}(\lambda, T_2)$  using the following expression

$$L(\lambda) = S(\lambda) \frac{L_{BB}(\lambda, T_2) - L_{BB}(\lambda, T_1)}{S_2(\lambda) - S_1(\lambda)} + \frac{S_2(\lambda)L_{BB}(\lambda, T_1) - S_1(\lambda)L_{BB}(\lambda, T_2)}{S_2(\lambda) - S_1(\lambda)}. \quad (6)$$

As can be seen from Eqs. (2, 3, 4, 5, 6), in order to calibrate the system (the uncalibrated signal levels) the reference measurements of the two signals  $S_1(\lambda)$  and  $S_2(\lambda)$  from a blackbody radiation source at two different temperatures  $T_1$  and  $T_2$  have to be taken. However, it takes relatively much time for a black body radiation source to stabilize at a certain given temperature. On the other hand, using two black body devices being set to two different temperatures would bring some displacement error due to the replacement of the black body devices may lead to some changes in the optical path.

Hence, a simplified calibration can also be done if instead of  $S_1(\lambda)$  or  $S_2(\lambda)$  the background radiation  $S_{amb}(\lambda)$  is used, i.e. the signal from the surroundings having some ambient temperature  $T_{amb}$  is measured. To a very good approximation, such a signal equals to the signal from a black body at temperature  $T_{amb}$ . During the experiment on the exhaust duct,  $T_{amb}$  was 308 K and the temperature  $T_2$  of the black body radiation source used for calibration for the fibres at the top and bottom positions was 842 K, and it was 897 K for the fibre at the middle position. The difference between  $L_{BB}(\lambda, T_2 \approx 842 \text{ K}) - L_{BB}(\lambda, T_{amb} \approx 308 \text{ K})$  and  $L_{BB}(\lambda, T_2 \approx 842 \text{ K}) - 0$  is 0.12 % (see [72, Eq. (2.15)] or [73] for the expression for  $L_{BB}(\lambda, T)$ ) and hence, to a very good approximation, the expression for the response function (3) can be simplified in the following way

$$\rho(\lambda) = \frac{S_2(\lambda) - S_{amb}(\lambda)}{L_{BB}(\lambda, T_2)}, \quad (7)$$

i.e. the reference measurements involve the black body radiation source being set to just a single one temperature. In this case

$$\beta(\lambda) = S_2(\lambda) - \rho(\lambda)L_{BB}(\lambda, T_2) = S_{amb}(\lambda). \quad (8)$$

Under the industrial conditions where time efficiency is of chief importance, which was the case in the application on the large Diesel engine (Sec. 2.4), the calibration procedure as given by Eqs. (7, 8) is highly advantageous and was employed instead of the one as given by Eqs. (3, 4).

It should be noted, however, that the procedure as given by Eqs. (3, 4) employs two strong signals  $S_1(\lambda)$  and  $S_2(\lambda)$  for the calibration as opposed to the case as given by Eqs. (7, 8) where  $S_{amb}(\lambda)$  is, in fact, background noise of the system. Therefore the resulting signal-to-noise ratio for the former case is higher than the one for the latter case.

### 2.1.9. Thermal Stability of the System

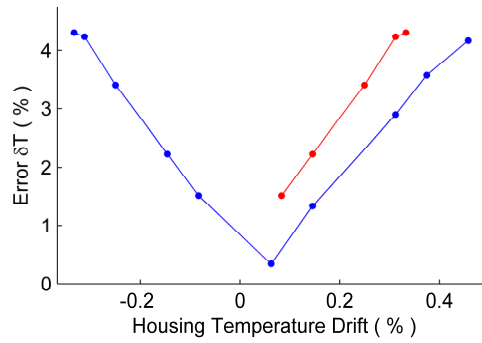
The IR camera of the system has InSb detector array having  $640 \times 512$  elements. The thermal stability of the InSb detector array is an important issue since this part of the system is the most sensitive to temperature variations.

An optical setup similar to the one from Sec. 2.1.6 was used to investigate the thermal stability of the system. The IR camera has built-in temperature sensor measuring the temperature of the IR camera housing.

In the experiment on thermal stability, the IR camera was being run within about six hours in the laboratory equipped with an air conditioning system maintaining constant temperature at about 22 °C (295 K). The drift of the housing temperature of the IR cam-

era was recorded. The temperature was gradually increasing from 26.25 °C (299.40 K) to 28.62 °C (301.77 K) within about six hours. That is the temperature increased by 2.38 degrees which is by 0.8 %.

The calibration as given by Eqs. (7, 8) was performed at a time point when housing temperature was 27.25 °C (300.40 K). Also, the signal  $S_{BB}(\lambda)$  from the blackbody radiation source having the temperature of the cavity of 894.4 °C (1167.6 K) was being measured at certain intervals within those six hours and the housing temperature was being registered at the same time points. Using Eq. (5), the measured signal  $S_{BB}(\lambda)$  was converted to  $L_{BB}(\lambda, T)$  which corresponds to the spectral radiance of a blackbody at temperature  $T$  which is unknown. Using the inversion of the Planck radiation law (see [72, Eq. (2.15)] or [73]) the temperature  $T$  was calculated from  $L_{BB}(\lambda, T)$  and compared to the true temperature of the blackbody radiation source of 894.4 °C (1167.6 K). The relative error  $\delta T$  [%] of the blackbody temperature measurement was then calculated and plotted as a function of the temperature drift of the IR camera housing [%] (Fig. 9, blue). The red curve in Fig. 9 is a mirror reflection of the negative part of the blue curve.



**Figure 9.** The relative error  $\delta T$  [%] of the blackbody temperature measurement as a function of the temperature drift of the IR camera housing [%] (blue). The red curve is a mirror reflection of the negative part of the blue curve. The optical setup used in the experiment was similar to the one described in Sec. 2.1.6.

It can be seen that the relative error is linearly proportional to the absolute value of the housing temperature drift. A red curve can be used to obtain a linear relation for the upper bound of the temperature measurement error as a function of the absolute value of the housing temperature drift

$$\delta T[\%] = 11.46 \cdot \left( \frac{|T_{Cam} - T_{CamCalib}|}{T_{CamCalib}[\text{K}]} \cdot 100 \right) + 0.562 \quad (9)$$

where  $\delta T$  is the relative error [%] of blackbody temperature measurement;  $T_{Cam}$  is the housing temperature [K] of the IR camera at the moment of recording the corresponding signal from the blackbody;  $T_{CamCalib}$  is the housing temperature [K] of the IR camera at the moment of performing the calibration as given by Eqs. (7, 8). The relation

$$\frac{T_{Cam} - T_{CamCalib}}{T_{CamCalib}[\text{K}]} \cdot 100 = T_{CamDrift}[\%] \quad (10)$$

gives, in fact, the drift  $T_{CamDrift}$  [%] of the IR camera housing temperature.

During the measurements on the exhaust duct of the large Diesel engine (Sec. 2.4) which were lasting for about 1.5 hours, the camera housing temperature was first increasing slowly starting from 39.06 °C (312.21 K) and reaching the maximum of 39.44 °C (312.59 K) and then decreasing down to 38.87 °C (312.02 K) when the calibration measurements were taken. Hence, the housing temperature drift was within  $\pm 0.2\%$  which, according to Eq. (9), means that the relative error of temperature measurement was within  $\pm 3\%$ .

It can be seen that even minor thermal instability of the IR camera housing (and hence the InSb array) leads to significant error on temperature measurement. The drift of the housing temperature is one of the major contributors to the total measurement error. The issue of stabilizing the camera temperature, which can be achieved by, e.g., placing the whole system into the temperature chamber, is under consideration for future developments.

## 2.2. Small-Scale Laboratory Burner

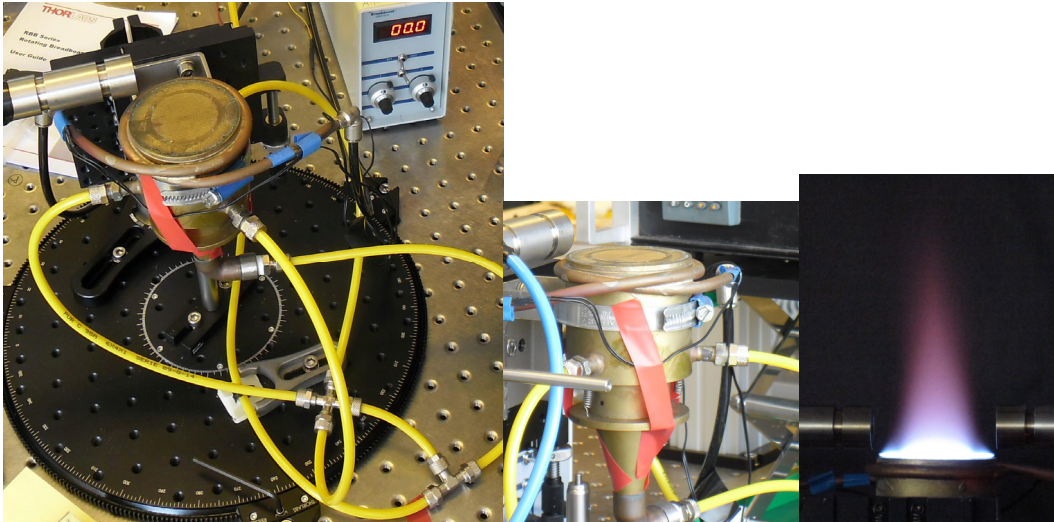
### 2.2.1. Purpose

As stated in the objectives of the PhD project (Sec. 1.3.1), the small scale laboratory burner producing a flame with known temperature profile is needed to prove the developed tomographic algorithms for gas temperature profile reconstruction (Sec. 3.3).

### 2.2.2. Description of the Burner

A flat flame burner producing laminar pre-mixed methane ( $\text{CH}_4$ ) / air flame with known axisymmetric temperature profile as described in Ref. [74] was used (Fig. 10). The copy of the burner from Ref. [74] was commissioned and setup in our lab (Sec. 2.2.3).





**Figure 10.** Flat flame burner. Shown in the photos and used in this project is a copy of the burner from Ref. [74].

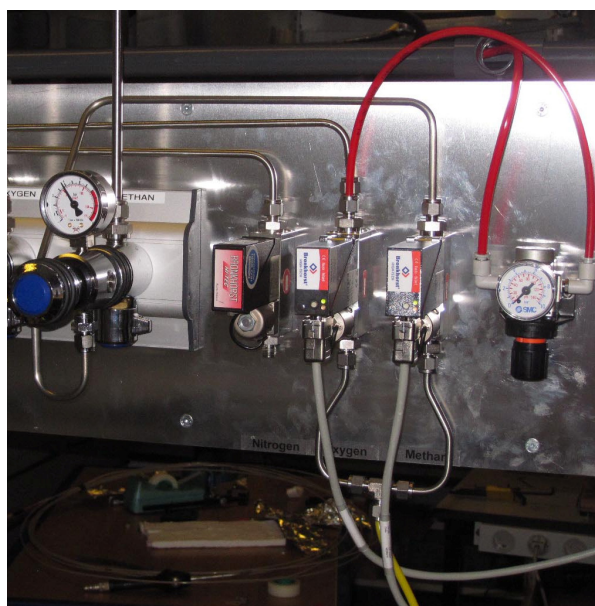
### 2.2.3. Setup of the Burner

The burner was mounted on either a translation stage for the parallel scanning scheme of measurements (Sec. 2.3.6, Fig. 23) or a rotation stage (as in Fig. 10, left) for the sweeping scanning scheme (Sec. 2.3.8, Fig. 25).

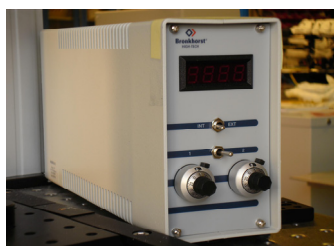
The burner was supplied with the gas mixture of methane and air from the two flow controllers – BRONKHORST HIGH-TECH EL-FLOW<sup>®</sup> Series Digital Gas Mass Flow Controllers, Fig. 11 and Fig. 12 (see Sec. 2.2.4 for the flow rate values). The flow controllers were operated by a BRONKHORST HIGH-TECH E-5700 Series Economical Power Supply/Readout System, Fig. 13. The air was supplied to the flow controller from a compressor. The methane was supplied from a gas bottle (Air Liquide Danmark).



**Figure 11.** BRONKHORST HIGH-TECH EL-FLOW<sup>®</sup> Series Digital Gas Mass Flow Controllers.

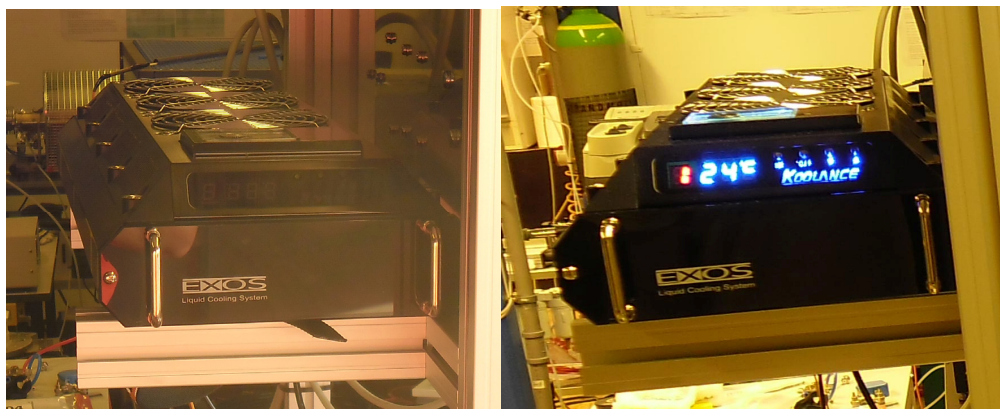


**Figure 12.** The system supplying the methane/air mixture to the burner.



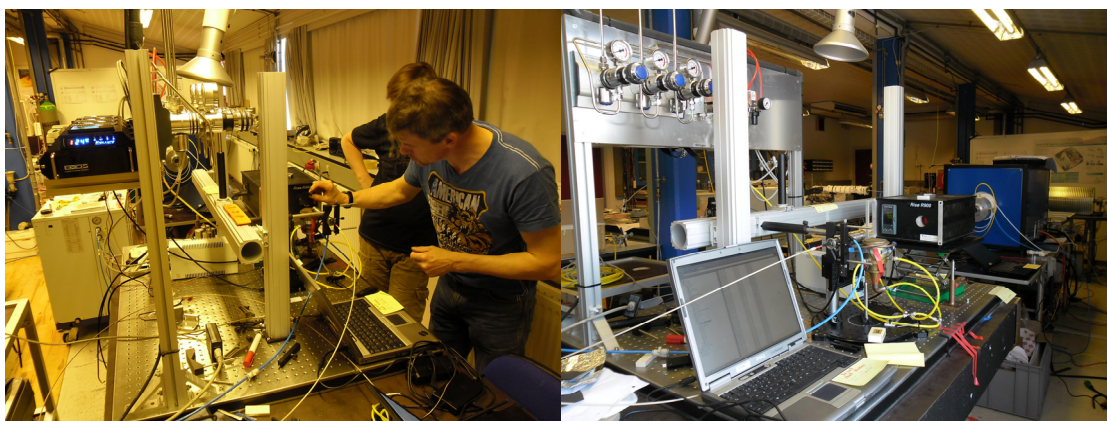
**Figure 13.** BRONKHORST HIGH-TECH E-5700 Series Economical Power Supply/Readout System.

The burner plate is equipped with a single water cooling loop (inner diameter 4 mm) which effectively conducts away the heat transferred to the burner plate from the flame [74]. The cooling loop is made of copper and is hard-soldered onto the circumference of the burner plate [74]. It should be noted that, the above mentioned cooling loop was developed in the work of Ref. [74]. In our setup, the cooling water was supplied to the loop by a Koolance Exos-2.5 Liquid Cooling System, Fig. 14. The temperature of the water “leaving” the loop did not exceed 60 °C.



**Figure 14.** Koolance Exos-2.5 Liquid Cooling System.

The whole setup of the burner is shown in Fig. 15 (the photos also include the elements of the optical setup used for the FTIR measurements, Sec. 2.3.3, Fig. 22).



**Figure 15.** The setup of the burner. The photos also include the elements of the optical setup used for the FTIR measurements (Sec. 2.3.3, Fig. 22).

#### 2.2.4. Operation Parameters

The measurements in this project were performed at the same height above the burner plate (HAB, [mm]) and at the same equivalence ratios ( $\varphi$ ) between air and methane as the reference measurements as reported in Ref. [74]. In particular, the measurements were performed at HAB = 12 mm and at  $\varphi = 1$  and  $\varphi = 0.8$ . The reference temperature profiles for the above conditions are given in Ref. [74]. It should be noted that, as follows from Ref. [74], the temperature profile is independent of other operating conditions such as methane and air flow rates through the inner and outer parts of the burner plate, etc.

The methane and air flow rates through the inner and outer parts of the burner plate which were established during the measurements in this PhD project are given in Table 1 ( $\varphi = 1$ ) and Table 2 ( $\varphi = 0.8$ ).

**Table 1.** The methane and air flow rates through the inner and outer parts of the burner plate at  $\phi = 1$  (stoichiometric combustion)

Inner		Outer	
CH <sub>4</sub>	Air	CH <sub>4</sub>	Air
1.03	9.84	1.19	11.37
[ltr./min]	[ltr./min]	[ltr./min]	[ltr./min]

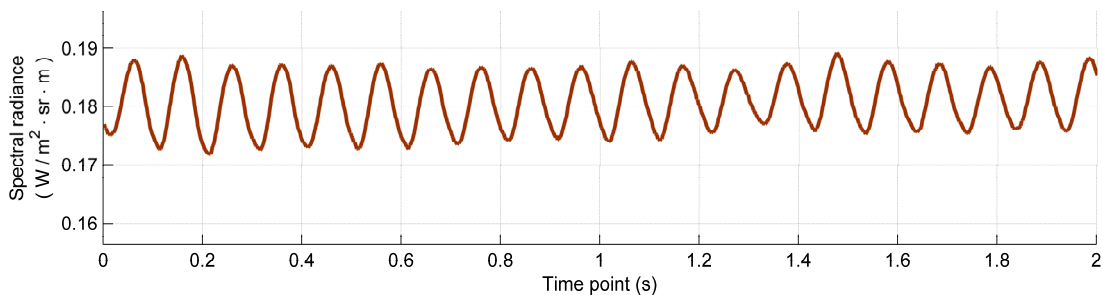
**Table 2.** The methane and air flow rates through the inner and outer parts of the burner plate at  $\phi = 0.8$  (lean combustion)

Inner		Outer	
CH <sub>4</sub>	Air	CH <sub>4</sub>	Air
1.03	12.30	1.19	14.20
[ltr./min]	[ltr./min]	[ltr./min]	[ltr./min]

It should be noted that the methane/air ratios were the same as in the work of Ref. [74] as expressed in the values of  $\phi$ , the absolute values of the flow rates were about two times lower in this PhD project than those in the work of Ref. [74] at  $\phi = 1$ . However at  $\phi = 0.8$  the flow rates in this project were almost the same as those in work [74] (more precisely, about 1.1 times higher than those of [74]).

### 2.2.5. Flame Fluctuations

Although the flow rates were kept constant by flow controllers some fluctuations in the emission signal from the flame were observed (Fig. 16). The emission signal from the flame instead of being constant resembles a sine wave with the amplitude of about 5 % of the mean value and a frequency of about 10 Hz. These fluctuations are essential consequence of the phenomena related to flame propagation dynamics taking into account buoyancy effects and convection.



**Figure 16.** Average emission signal from the flame of the burner. The signal was recorded using the **multichannel IR spectrometer system**. The version of the optical setup used in this experiment was similar to the one described in Sec. 2.1.6. The frame rate of the IR camera was 500 Hz.

### 2.2.6. Reference Temperature Profile Measurements in the Work of Ref. [74]

The reference measurements were performed in the work of Ref. [74] using a laser-based technique employing coherent anti-Stokes Raman scattering (CARS). As follows from Ref. [74], the standard deviation of the point temperature measurement there was 75 K which means that the values in the reference temperature profile should be considered with the deviation of  $\pm 75$  K.

## 2.3. FTIR Tomographic Measurements on the Lab Burner

### 2.3.1. Introduction and Important Remarks

As discussed in Sec. 2.1.9 the **multichannel IR spectrometer system** (a first prototype of the 2Dt IR spectroscopic tomography system) is sensitive to variations in the temperature of the IR camera housing which take place even under lab conditions. For this reason it was decided to use a commercially available FTIR spectrometer for the test spectral measurements on the burner (see Sec. 2.2 for the detailed description of the burner). The results of the spectral measurements are input data to the developed tomographic algorithms for gas temperature profile reconstruction (Sec. 3.3).

The FTIR spectrometer provides more thermal stability and a higher spectral resolution thus providing input data to the reconstruction algorithm of highest possible quality. This is important for testing the developed algorithms since tomographic reconstruction algorithms are known to be very sensitive to uncertainties in input data (see Sec. 3.3).

### 2.3.2. FTIR Spectrometer

A commercially available Bomem/ABB model MB155 FTIR spectrometer (Fig. 17) was used in the test measurements on the burner. The highest nominal spectral resolution of the spectrometer is  $1 \text{ cm}^{-1}$  however most of measurements were performed at the resolution of  $2 \text{ cm}^{-1}$  since at this resolution an optimal balance between the resolution and other operation parameters of the spectrometer is achieved. Particular values of parameters at which the spectral measurements were performed are given in detail in Sec. 2.3.6, Sec. 2.3.8 and Sec. 4.

The detector used was InSb. The detector was cooled using liquid nitrogen. The detector provides spectral measurements in the  $1.5\text{-}5.1 \mu\text{m}$  spectral range where several emission bands of molecules under interest are located (e.g.  $\text{H}_2\text{O}$ ,  $\text{C}_x\text{H}_y$ ,  $\text{CO}_2$ ,  $\text{CO}$ ).



**Figure 17.** Bomem/ABB model MB155 FTIR spectrometer.

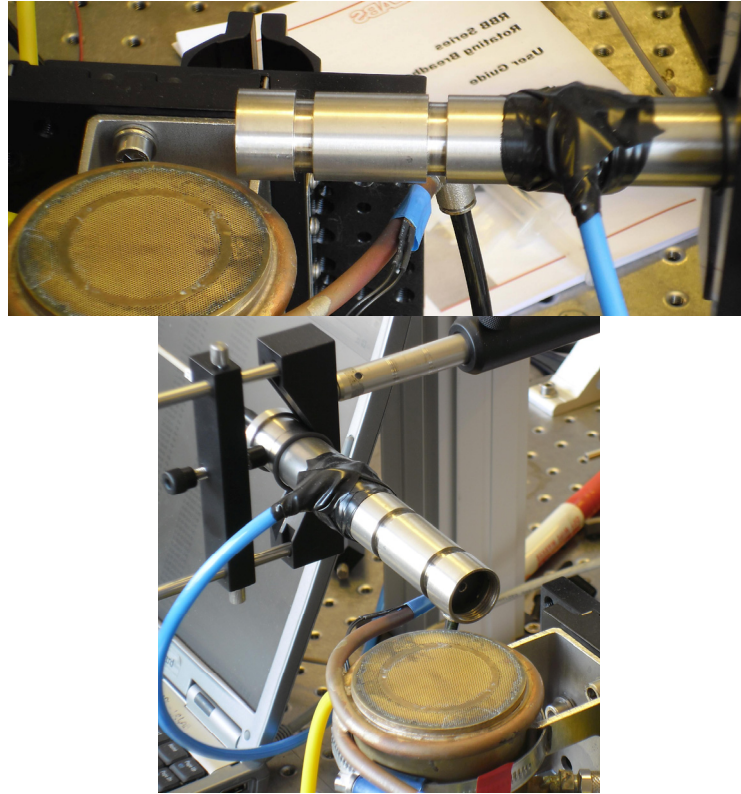
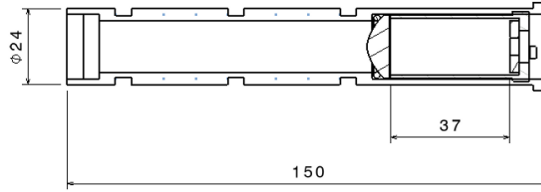
The spectrometer as well as the other parts of the optical setup (Sec. 2.3.3) was purged with pressurized air generated by BALSTON<sup>®</sup> 75-45 FT-IR Purge Gas Generator (Fig. 18).



**Figure 18.** BALSTON® 75-45 FT-IR Purge Gas Generator.

### 2.3.3. Optical Setup for the FTIR Measurements

The IR optical fiber (chalcogenide IR-glass fiber) and the fiber optical adaptor (Fig. 19) used in the FTIR measurements here were the same as those three fibers and three fiber optical adaptors used in the **multichannel IR spectrometer system** when it was applied on the exhaust duct of the large Diesel engine (Sec. 2.4.2, Fig. 26).



**Figure 19.** Fiber optical adaptor.

The fiber optical adaptor was purged with the pressurized air generated by BALSTON<sup>®</sup> 75-45 FT-IR Purge Gas Generator (Fig. 18). The purge air was delivered to the adaptor using the blue hose which can be seen in the photos of Fig. 19.

A lens was used inside the fiber adaptor to focus the light onto the fiber end. Also, an aperture (inner diameter 3 mm, antireflection coated) was inserted into the front part of the adaptor (before the lens on the way of the incident beam, i.e. aperture-lens-fiber) to provide a narrow field of view. The field of view provided by the adaptor was observed to lie within a cylinder having the diameter of about 6 mm. That was established before the parallel scanning measurements (Sec. 2.3.6).

A blackbody radiation source (Fig. 20) with a cavity temperature of 894.4 °C (1167.6 K) was used for transmission measurements within each scheme of measurements (see Sec. 2.3.7 for more details on the types of measurements).

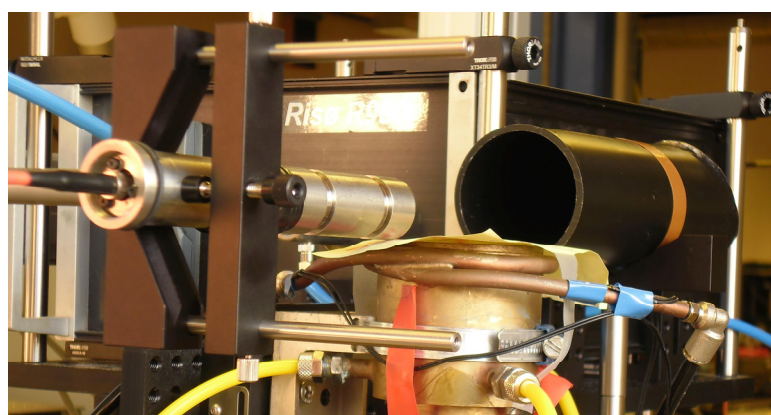




**Figure 20.** The blackbody radiation source. The temperature of the cavity was 894.4 °C (1167.6 K).

The blackbody cavity was also purged with the pressurized air generated by BALSTON<sup>®</sup> 75-45 FT-IR Purge Gas Generator (hoses delivering the purge air to the cavity are not shown in Fig. 20). It is important to purge the hot cavity of the blackbody in order to remove hot CO<sub>2</sub> and H<sub>2</sub>O (present inside the cavity from the ambient air) which significantly affect the spectral measurements on the flame since any flame also contains hot CO<sub>2</sub> and H<sub>2</sub>O.

The flame emission measurements were performed using a cavity having ambient temperature (Fig. 21). The ambient temperature cavity as well as the cavity of the blackbody radiation source has a shape of a cylinder with tilted bottom.



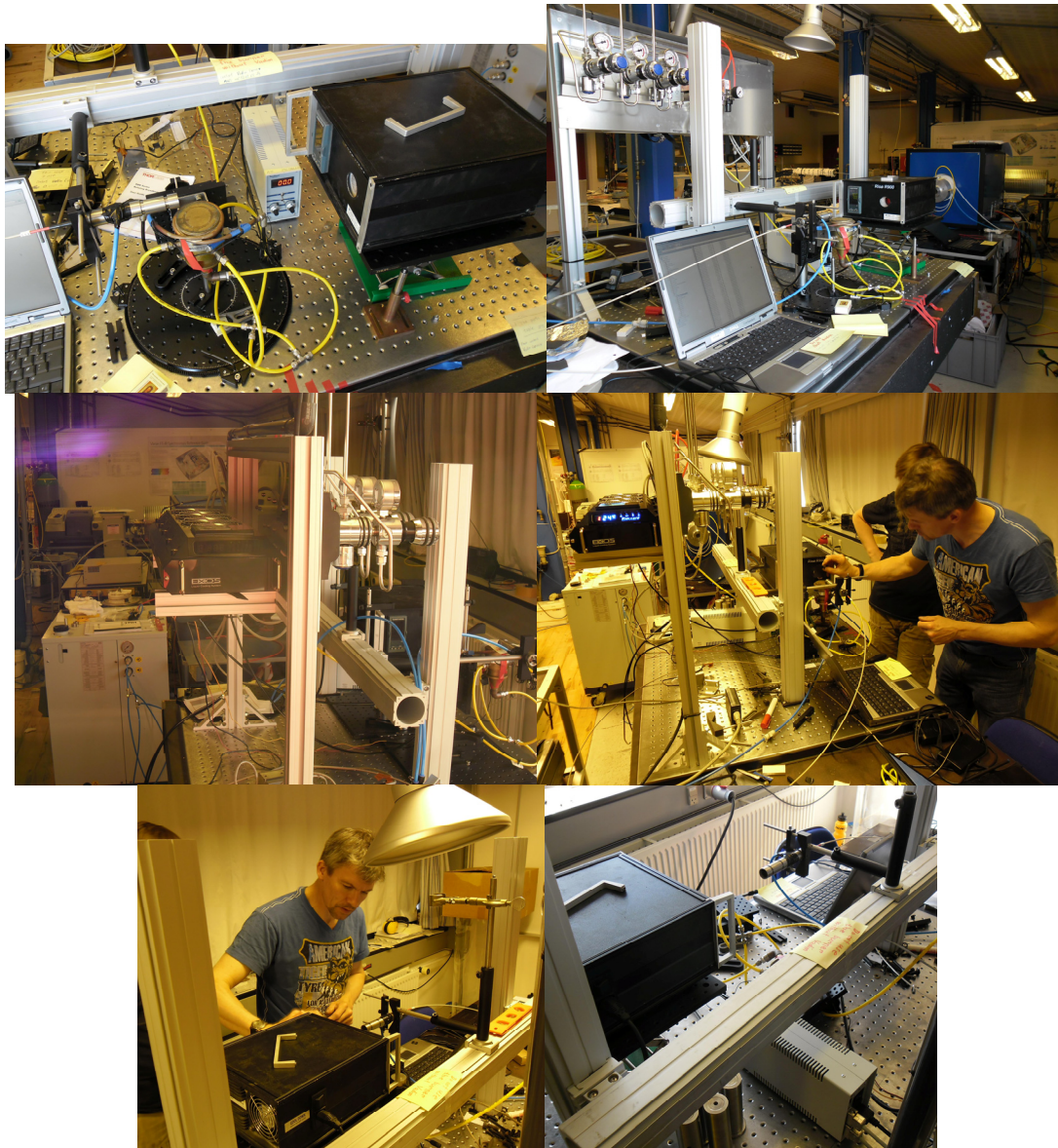
**Figure 21.** The ambient temperature cavity used for flame emission measurements (see Sec. 2.3.7 for more details on the types of measurements).

When the optical setup including the fiber adaptor and the blackbody cavity is aligned and calibrated, even minor changes in the positions lead to an error in the temperature measurements. However it is necessary to perform transmission measurements (see Sec. 2.3.7 for more details on the types of measurements) when the blackbody is in its aligned position behind the burner (as in Fig. 20) and emission measurements when there should not be any source of radiation behind the burner. Furthermore, tomographic algorithms (Sec. 3.3) require to perform many such pairs of measurements such that to “scan” the flame of the burner either in the scheme of parallel (Sec. 2.3.6) or sweeping scanning (Sec. 2.3.8). It means that either the blackbody should be removed from

the field of view of the fiber optical adaptor and then returned precisely to its aligned position or the radiation of the blackbody should be blocked using some non-transparent sheet. The latter solution preserves the position of the blackbody however it is not suitable for flame emission measurements when the significant part of flame radiation is reflected from the sheet and is detected by the fiber adaptor hence bringing an additive error to the flame emission measurements of about 5%. For this reason the blackbody was mounted on a translation stage such that it could be removed from the field of view and returned back precisely to its aligned position. Such a displacement of the blackbody brings an error on the temperature measurement of about 0.1 % which can be regarded as negligible compared to the other sources of error (to be considered further throughout Sec. 2, Sec. 3 and Sec. 4).

The ambient temperature cavity was fixed together with the blackbody on the same translation stage. When the blackbody was removed from the field of view, the cavity was placed within the field of view (as in Fig. 21). It should be noted that the flame emission signal obtained with the ambient temperature cavity was observed to be by about 0.8-2.1% higher than that obtained without using anything but just when having the surroundings within the field of view of the fiber adaptor. It means that there is still some reflection from the cavity. The high value of about 2.1% of the difference observed can also be due to flame fluctuations (Sec. 2.2.5) which are significant even though the average of 50 scans (spectra) at the rate of 57 scans/min was taken for each spectral measurement, i.e. it does not necessarily mean that the reflections from the cavity are so high. It was concluded that the reflections from the cavity are anyway less significant than other sources of error (considered throughout Sec. 2, Sec. 3 and Sec. 4) so the cavity was used for emission measurements to guarantee absence of significant radiation occurring in the field of view by chance, i.e. to guarantee constant background.

The whole optical setup is shown in Fig. 22 (the photos also include elements of the burner setup, Sec. 2.2.3, Fig. 15).



**Figure 22.** The optical setup. The photos also include elements of the burner setup (Sec. 2.2.3, Fig. 15). In the photos, the burner is installed on the rotation stage for the sweeping scanning scheme of measurements (Sec. 2.3.8).

#### 2.3.4. Calibration of the System

Once all the elements of the optical setup (Sec. 2.3.3) are aligned and fixed before the measurements, the whole FTIR measurement system, i.e. the FTIR spectrometer and the optical setup, has to be calibrated.

The spectral measurements performed by the FTIR system can be described by the linear model (2) in the same way as those performed by the **multichannel IR spectrometer system** (Sec. 2.1.8). Hence the calibration of the FTIR system was performed according to Eqs. (7, 8) similarly to the calibration of the **multichannel IR spectrome-**

**ter system.** The temperature of the blackbody cavity in the reference measurements of the blackbody radiation was 894.4 °C (1167.6 K).

It should be noted that during the reference (calibration) measurements the blackbody radiation source was located at some distance from the entrance aperture of the fiber adaptor (Fig. 19, Sec. 2.3.3). The distance varied from about 92 mm to about 160 mm depending on the layout of the measurements (Sec. 2.3.6 and Sec. 2.3.8). The burner having the diameter of the plate of 60 mm [74] was located very close to the detector (fiber adaptor). In fact, if looking from top, the entrance aperture of the fiber adaptor was lying on the edge of the burner plate. Taking into account that the field of view provided by the fiber adaptor (Sec. 2.3.3) is not strictly cylindrical some mismatch occurs between the measured value of flame radiation expressed in the units of spectral radiance and the true value of the flame radiation when measurements on the flame are performed. However this error was observed to be minor. No significant change in the received signal was observed when moving the blackbody to and fro within the distance 60-160 mm from the entrance aperture of the fiber adaptor.

### 2.3.5. Thermal Stability of the System

The thermal stability of the system was investigated by comparing the reconstructed temperature profile value at the center of the burner plate (see e.g. Fig. 36) (calculated using the algorithms described in Sec. 3.3) from the calibration data (the response function and the background radiation, Eqs. (7, 8)) obtained before the measurements on the flame and after.

The resulting temperature value at the center of the burner plate (where the temperature of the flame is expect to be at about its maximum) differed by about 25 K or 1.1%.

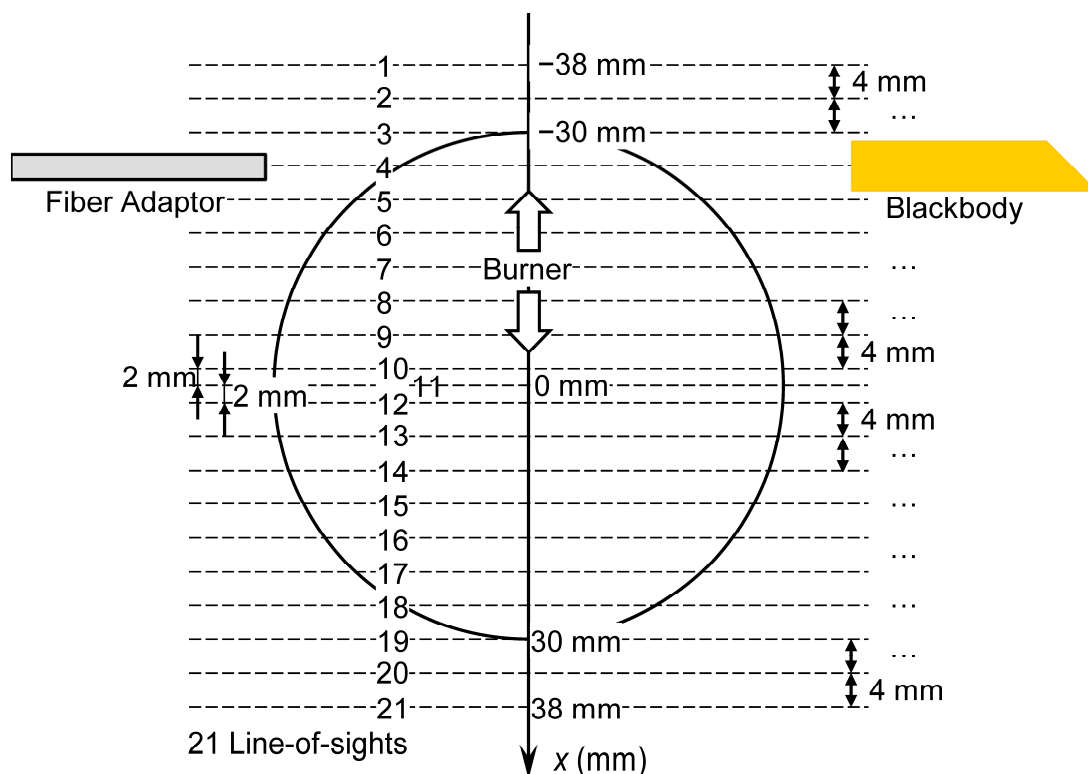
It should be noted that, since the fiber adaptor (Fig. 19) was located very close to the flame (see Sec. 2.3.6, Sec. 2.3.8) during the FTIR spectral measurements on the flame, adaptor's temperature was increasing during the measurements on the flame. Hence it is important to use the calibration data obtained immediately after the measurements on the flame to guarantee that the temperature of the fiber adaptor during the calibration measurements is the same as that during the measurements on the flame. The calibration data used for the temperature profile calculations (Sec. 3.3) was obtained immediately after the measurements on the flame hence the error on the temperature measurement caused by thermal instability of the system was expected to be not more than 1.1% or 25 K.

### 2.3.6. Parallel Scanning Scheme of Measurements

The temperature profile of the burner is axisymmetric [74]. Hence one projection with several line-of-sight measurements is enough for tomographic reconstruction (Sec. 3.3). Furthermore, it is sufficient that line-of-sights "cover" just half of the burner.

In the parallel scanning scheme of measurements, there was only one projection consisting of several line-of-sights which were parallel to each other. The plane where the line-of-sights were lying was located at the height above the burner plate of 12 mm (HAB = 12 mm, see also Sec. 2.2.4).

In the experiment, the number of line-of-sights was 21 (Fig. 23). To minimize the misalignment error, they were arranged to cover the whole burner plate and the average between the corresponding line-of-sights was taken (1 and 21, 2 and 20, etc.). The distance between the central axes of line-of-sights 1 to 10 and 12 to 21 was 4 mm. The distance between the central axes of line-of-sights 10 and 11 was 2 mm. The same distance was between line-of-sights 11 and 12. Line-of-sight 11 was the central line-of-sight.



**Figure 23.** The layout of the parallel scanning measurements. The plane where the line-of-sights were lying was located at the height above the burner plate of 12 mm ( $HAB = 12$  mm, see also Sec. 2.2.4). The values of  $x = \pm 30$  mm correspond to the edge of the burner plate [74].

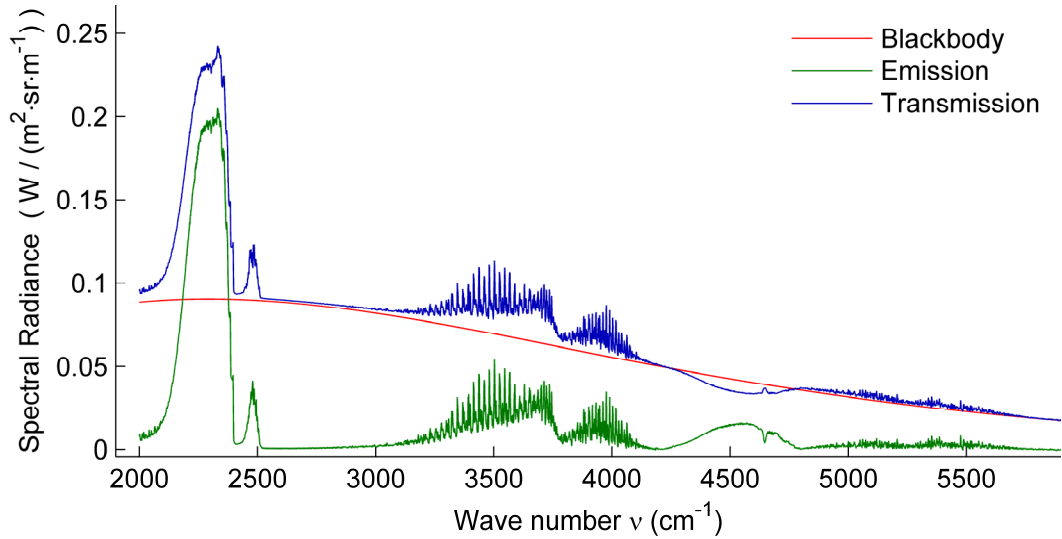
The fiber adaptor (Fig. 19), acting as a receiver or detector in the scheme, and the blackbody (Fig. 20), acting as a source, were fixed in their aligned positions during the parallel scanning measurements (as mentioned in Sec. 2.3.3, the blackbody was mounted on the translation stage such that it could be removed from the field of view of the detector and returned back precisely to its aligned position in the field of view). The distance between the entrance aperture of the fiber adaptor and the exit of the blackbody cavity was about 92 mm (the diameter of the burner plate is 60 mm [74]). The burner was mounted on a translation stage such that it could be moved across the field of view of the detector in order to provide the 21 line-of-sights.

At each line-of-sight, denoted by position  $x$  [mm] (Fig. 23), two FTIR spectral measurements were taken – a transmission measurement and an emission measurement (Sec. 2.3.7). This is necessary for the reconstruction algorithm (Sec. 3.3). Hence two spectra are recorded at each line-of-sight position: an emission spectrum  $I_g(\nu, x)$

$[\text{W}/(\text{m}^2 \text{ sr m}^{-1})]$  ( $\nu$  is any observed wave number  $[\text{cm}^{-1}]$ ) and a transmission spectrum  $I_R(\nu, x)$   $[\text{W}/(\text{m}^2 \text{ sr m}^{-1})]$ .

For the emission and transmission FTIR spectral measurements the FTIR spectrometer was set to the spectral resolution of  $2 \text{ cm}^{-1}$ . Each spectrum  $I_g(\nu, x)$  and  $I_R(\nu, x)$  is an average of 50 spectra. The spectrometer was working at the rate of 57 spectra per minute. The following parameters for conversion of the interferograms (which is the “raw” signal received by an FTIR spectrometer) to spectra were applied: Y Units: Single Beam; Apodization: Triangular; Zero Fill: None; Scale Factor: 1; Adjust for X Resolution: Yes (refer to the Bomem/ABB model MB155 FTIR spectrometer Manual).

An example of the spectra  $I_g(\nu, x)$  and  $I_R(\nu, x)$  is given in Fig. 24 together with the blackbody radiation spectrum. The blackbody radiation spectrum is given by the Planck radiation law [72, Eq. (2.15)], [73]. The curve in Fig. 24 corresponds to the temperature of the blackbody  $T_0 = 894.4 \text{ }^\circ\text{C}$  (1167.6 K). It was the temperature of the cavity of the blackbody radiation source (Sec. 2.3.3, Fig. 20) during the transmission measurements. The spectra  $I_g(\nu, x)$  and  $I_R(\nu, x)$  in Fig. 24 correspond to the central line-of-sight (number 11) defined by  $x = 0 \text{ mm}$  (Fig. 23), the height above the burner plate of 12 mm (HAB = 12 mm) (Sec. 2.2.4) and the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) (Sec. 2.2.4).



**Figure 24.** Examples of the emission spectrum  $I_g(\nu, x)$  (olive), transmission spectrum  $I_R(\nu, x)$  (blue), and the blackbody radiation spectrum (the Planck radiation law [72, Eq. (2.15)], [73] at  $T_0 = 894.4 \text{ }^\circ\text{C}$  (1167.6 K)) (red). The spectra correspond to the central line-of-sight (number 11,  $x = 0 \text{ mm}$ ) (Fig. 23), the height above the burner plate of 12 mm (HAB = 12 mm) (Sec. 2.2.4) and the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) (Sec. 2.2.4).

The signals  $I_R(\nu, x)$  and  $I_g(\nu, x)$  at some observed wave number  $\nu$   $[\text{cm}^{-1}]$  and all the 21 values of  $x$  are the input data to the algorithm for 2Dt reconstruction of gas temperature profiles (Sec. 3.3). The output data of the algorithm is the reconstructed temperature profile of the burner. An example of the reconstructed temperature profile from the results of the parallel scanning measurements described here is given in Fig. 36

(Sec. 4) (see also the description under that figure). The reconstructed temperature profile in Fig. 36 obtained from the parallel scanning measurements corresponds to the input data of the algorithm as given in Table 3. It should be noted that the algorithm is using the average of the values at  $x = -38$  mm and  $x = 38$  mm,  $x = -34$  mm and  $x = 34$  mm, etc. In this way the misalignment error is reduced to minimum.

**Table 3.** The signals  $I_R(v, x)$  and  $I_g(v, x)$  at  $v = 2270.0496 \text{ cm}^{-1}$  and all the 21 values of  $x$  as the input data to the algorithm for 2Dt reconstruction of gas temperature profiles (Sec. 3.3). The data from this table was used to obtain the temperature profile for parallel scanning in Fig. 36, Sec. 4. It should be noted that the algorithm is using the average of the values at  $x = -38$  mm and  $x = 38$  mm,  $x = -34$  mm and  $x = 34$  mm, etc. In this way the misalignment error is reduced to minimum. The values of  $x = \pm 30$  mm correspond to the edge of the burner plate [74].

Parallel scanning:		
$\tilde{n} = 1$		
HAB = 12 mm (height above the burner plate)		
Wave number $v = 2270.0496 \text{ cm}^{-1}$		
(NB: $x = 0$ mm is the central line-of-sight)		
$x$ [mm]	$I_R(v,x)$ [ $\text{W}/(\text{m}^2 \text{ sr m}^{-1})$ ]	$I_g(v,x)$ [ $\text{W}/(\text{m}^2 \text{ sr m}^{-1})$ ]
-38	0.091911605	0.000105947
-34	0.092019957	0.000335024
-30	0.092374185	0.011090653
-26	0.096858287	0.015659354
-22	0.133410393	0.071970683
-18	0.176038541	0.131435054
-14	0.199813454	0.157522864
-10	0.214786857	0.174463271
-6	0.223092449	0.184831717
-2	0.227768251	0.190274317
0	0.22913932	0.19265806
2	0.228797595	0.192933107
6	0.226192981	0.191837086
10	0.219946076	0.18516094
14	0.208935853	0.17387567
18	0.181447803	0.144799847
22	0.147917049	0.098816957
26	0.115915724	0.049546002
30	0.092374185	0.011090653
34	0.091244824	0.000500628
38	0.09139485	0.000208173

The wave number  $\nu = 2270.0496 \text{ cm}^{-1}$  was chosen from the conditions of maximum signal from the flame, i.e. to guarantee the maximum signal-to-noise ratio, the maximum absorption from the flame, and the minimum absorption from cold ambient gases. The value  $\nu = 2270.0496 \text{ cm}^{-1}$  is within the  $4.3 \text{ }\mu\text{m}$  ( $2350 \text{ cm}^{-1}$ ) band of  $\text{CO}_2$ . At this wave number, cold  $\text{CO}_2$  in the ambient air has negligible absorption at the same time absorption from hot  $\text{CO}_2$  in the flame is sufficiently strong. Also, as can be seen from Fig. 24, the emission from  $\text{CO}_2$  in the flame at this wave number is very strong.

### 2.3.7. Transmission and Emission Measurements

As was mentioned, at each line-of-sight, denoted by position  $x$  [mm] (Fig. 23) in the parallel scanning scheme (Sec. 2.3.6) or by angle  $\theta$  [degrees] (Fig. 25) in the sweeping scanning scheme (Sec. 2.3.8), two FTIR spectral measurements were taken – a transmission measurement and an emission measurement. This is necessary for the reconstruction algorithm (Sec. 3.3).

The transmission measurement is taken when the blackbody radiation source is in the field of view of the detector such that the signal  $I_R(\nu, x)$  [ $\text{W}/(\text{m}^2 \text{ sr m}^{-1})$ ], the transmission signal at some observed wave number  $\nu$  [ $\text{cm}^{-1}$ ] and position  $x$  [mm] (or  $\theta$  [degrees]) of the line-of-sight, contains emission from the flame and the portion of the blackbody radiation transmitted by the flame.

The emission measurement is taken when there is no any source of radiation except the flame radiation in the field of view of the detector such that the signal  $I_g(\nu, x)$  [ $\text{W}/(\text{m}^2 \text{ sr m}^{-1})$ ], the emission signal at some wave number  $\nu$  [ $\text{cm}^{-1}$ ] and position  $x$  [mm] (or  $\theta$  [degrees]) of the line-of-sight, is in fact the emission signal from the flame.

As mentioned in Sec. 2.3.3, in the parallel scanning measurements (Sec. 2.3.6) the blackbody was mounted on the translation stage such that it could be removed from the field of view of the detector and returned back precisely to its aligned position in the field of view thus providing emission and transmission measurements respectively.

Also in the parallel scanning measurements (Sec. 2.3.6), the ambient temperature cavity was fixed together with the blackbody on the same translation stage such that when the blackbody was removed from the field of view, the cavity was placed within the field of view (as in Fig. 21) thus providing emission measurements (see Sec. 2.3.3 for a more detailed discussion).

It should be noted that in the sweeping scanning measurements (Sec. 2.3.8) the radiation from the blackbody was blocked using a non-transparent sheet of paper thus providing emission measurements (see a more detailed discussion in Sec. 2.3.8).

### 2.3.8. Sweeping Scanning Scheme of Measurements

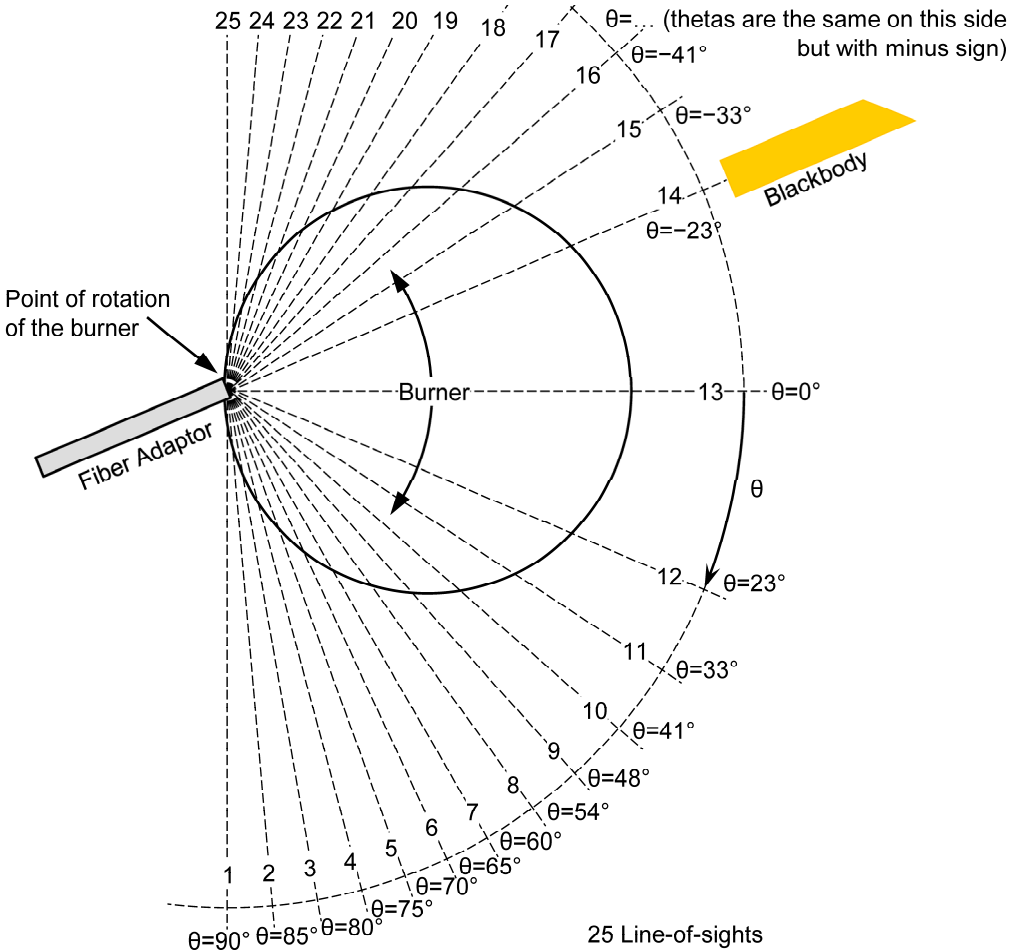
As mentioned above, the temperature profile of the burner is axisymmetric [74]. Hence one projection with several line-of-sight measurements is enough for tomographic reconstruction (Sec. 3.3). Furthermore, it is sufficient that line-of-sights “cover” just half of the burner.

In the sweeping scanning scheme of measurements, there was only one projection consisting of several line-of-sights which were “radiating” from the point where the



entrance aperture of the detector (the fiber adaptor) was located, if looking from top (Fig. 25). At the same time, if looking from top, that point coincides with the edge of the burner. The plane where the line-of-sights were lying was located at the height above the burner plate of 12 mm ( $HAB = 12\text{ mm}$ , see also Sec. 2.2.4).

In the experiment, the number of line-of-sights was 25 (Fig. 25). To minimize the misalignment error, they were arranged to cover the whole burner plate and the average between the corresponding line-of-sights was taken (1 and 25, 2 and 24, etc.). The angular separation between the central axes of adjacent line-of-sights varied from  $5^\circ$  to  $23^\circ$ . The line-of-sights were more densely arranged ( $5^\circ$  of angular separation) at the parts of the burner corresponding to expected steep gradients in the temperature profile of the flame which takes place closer to the edge of the burner. The line-of-sights covering parts of the burner where no steep temperature gradients were expected were arranged less densely (the angular separation up to  $23^\circ$ ).



**Figure 25.** The layout of the sweeping scanning measurements. The plane where the line-of-sights were lying was located at the height above the burner plate of 12 mm ( $HAB = 12\text{ mm}$ , see also Sec. 2.2.4). The diameter of the burner plate is 60 mm [74].

The fiber adaptor (Fig. 19), acting as a receiver, or detector, in this scheme as well as in the scheme of parallel scanning (Sec. 2.3.6), and the blackbody (Fig. 20), acting as

a source, were fixed in their aligned positions during the sweeping scanning measurements. The burner was mounted on a rotation stage (as shown in the leftmost photo of Fig. 10) such that it could be rotated with respect to a point on the edge of the burner plate (if looking from top) in which also the entrance aperture of the detector (fiber adaptor) was located as shown in Fig. 25.

It should be noted that in the sweeping scanning scheme it was difficult to locate the exit of the blackbody cavity sufficiently close to the burner due to the burner requires some space around it such that it can be rotated together with its numerous hoses (Fig. 10). The distance between the entrance aperture of the fiber adaptor (coinciding with the point of rotation of the burner in Fig. 25) and the exit of the blackbody cavity was about 160 mm whereas the diameter of the burner plate is 60 mm [74]. That may bring some error to the measurements since the blackbody in its aligned position for the sweeping scanning measurements was also used for the calibration measurements (the calibration was performed according to Eqs. (7, 8) similarly to the calibration of the **multichannel IR spectrometer system** (Sec. 2.3.4)). Since the blackbody radiation source is sufficiently far from the burner and the field of view provided by the fiber adaptor (Sec. 2.3.3) is not perfectly cylindrical some mismatch occurs between the measured value of flame radiation expressed in the units of spectral radiance and the true value of the flame radiation. However this error was observed to be minor. No significant change in the received signal was observed when moving the blackbody to and fro within the distance 60-160 mm from the entrance aperture of the fiber adaptor.

Also in the sweeping scanning measurements, since the blackbody is sufficiently far from the flame, emission measurements (Sec. 2.3.7) were performed by simply blocking the blackbody radiation using a non-transparent sheet of thick paper (see also Sec. 2.3.3). During those measurement the blackbody was not installed on the translation stage like in the parallel scanning measurements during which the blackbody was closer to the entrance aperture of the fiber adaptor (the distance between the entrance aperture of the fiber adaptor and the entrance of the blackbody cavity during the parallel scanning measurements was about 92 mm, i.e. by about 68 mm shorter than that during the sweeping scanning measurements).

As in the parallel scanning measurements, two FTIR spectral measurements - emission and transmission (Sec. 2.3.7) - were taken at each line-of-sight position defined in this case by angle  $\theta$  [degrees] (Fig. 25) instead of  $x$  [mm], i.e. an emission  $I_g(\nu, \theta)$  [ $W/(m^2 \text{ sr m}^{-1})$ ] ( $\nu$  is any observed wave number [ $\text{cm}^{-1}$ ]) and transmission  $I_R(\nu, \theta)$  [ $W/(m^2 \text{ sr m}^{-1})$ ] spectrum were recorded at each line-of-sight.

For the emission and transmission FTIR spectral measurements in the sweeping scanning scheme, the FTIR spectrometer was set to the spectral resolution of  $1 \text{ cm}^{-1}$ . Each spectrum  $I_g(\nu, \theta)$  and  $I_R(\nu, \theta)$  is an average of 20 spectra.

The spectra  $I_g(\nu, \theta)$  and  $I_R(\nu, \theta)$  for the central line-of-sight ( $\theta = 0^\circ$ ), the height above the burner plate of 12 mm ( $HAB = 12 \text{ mm}$ ) (Sec. 2.2.4) and the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) (Sec. 2.2.4) are the same as those given in Fig. 24 for central line-of-sight of the parallel scanning layout (other parameters, as can be seen, are the same). The temperature of the cavity of the blackbody radiation source (Sec. 2.3.3, Fig. 20) during the transmission measurements in the sweeping scanning scheme was also  $T_0 = 894.4 \text{ }^\circ\text{C}$  (1167.6 K) as in the parallel scanning scheme.

The signals  $I_R(v, \theta)$  and  $I_g(v, \theta)$  at some observed wave number  $v$  [ $\text{cm}^{-1}$ ] and all the 25 values of  $\theta$  are the input data to the algorithm for 2Dt reconstruction of gas temperature profiles (Sec. 3.3). The output data of the algorithm is the reconstructed temperature profile of the burner. An example of the reconstructed temperature profile from the results of the sweeping scanning measurements described here is given in Fig. 36 (Sec. 4) (see also the description under that figure). The reconstructed temperature profile in Fig. 36 obtained from the sweeping scanning measurements corresponds to the input data of the algorithm as given in Table 4. It should be noted that the algorithm is using the average of the values at  $\theta = 90^\circ$  and  $\theta = -90^\circ$ ,  $\theta = 85^\circ$  and  $\theta = -85^\circ$ , etc. In this way the misalignment error is reduced to minimum.

**Table 4.** The signals  $I_R(v, \theta)$  and  $I_g(v, \theta)$  at  $v = 2270.0496 \text{ cm}^{-1}$  and all the 25 values of  $\theta$  as the input data to the algorithm for 2Dt reconstruction of gas temperature profiles (Sec. 3.3). The data from this table was used to obtain the temperature profile for sweeping scanning in Fig. 36, Sec. 4. It should be noted that the algorithm is using the average of the values at  $\theta = 90^\circ$  and  $\theta = -90^\circ$ ,  $\theta = 85^\circ$  and  $\theta = -85^\circ$ , etc. In this way the misalignment error is reduced to minimum.

Sweeping scanning:		
fi = 1		
HAB = 12 mm		
Wave number $v = 2270.0496 \text{ cm}^{-1}$		
(NB: theta = 0 degrees is the central angular position)		
theta, deg	IR(v,theta) [W/(m <sup>2</sup> sr m <sup>-1</sup> )]	Ig(v,theta) [W/(m <sup>2</sup> sr m <sup>-1</sup> )]
90	0.09180212	0.00341215
85	0.091700172	0.003230498
80	0.0916631	0.003630255
75	0.091746512	0.003945366
70	0.091369615	0.004696996
65	0.091125559	0.005419896
60	0.090507696	0.009209869
54	0.095444423	0.027999708
48	0.110090871	0.052960147
41	0.140829567	0.092132676
33	0.167604671	0.126482783
23	0.201636578	0.164765589
0	0.228359163	0.194969833
-23	0.219072678	0.183913171
-33	0.202724017	0.165766528
-41	0.187434991	0.148305712
-48	0.168815683	0.12541079
-54	0.148939022	0.097467925
-60	0.12450871	0.069240844
-65	0.112166892	0.043976416
-70	0.100696261	0.02868739

Sweeping scanning:		
$f_i = 1$		
HAB = 12 mm		
Wave number $\nu = 2270.0496 \text{ cm}^{-1}$		
(NB: theta = 0 degrees is the central angular position)		
<b>theta, deg</b>	<b>IR(<math>\nu</math>,theta) [W/(m<sup>2</sup> sr m<sup>-1</sup>)]</b>	<b>Ig(<math>\nu</math>,theta) [W/(m<sup>2</sup> sr m<sup>-1</sup>)]</b>
-75	0.097171351	0.019167971
-80	0.092997685	0.013140097
-85	0.092268606	0.009132019
-90	0.092284053	0.007037771

The wave number  $\nu = 2270.0496 \text{ cm}^{-1}$  is the same as for the parallel scanning measurements and the choice of the wave number has been discussed in the end of Sec. 2.3.8.

## 2.4. Simultaneous Fast Exhaust Gas Temperature Measurements on a Large Diesel Engine

### 2.4.1. Introduction and Remarks

The **multichannel IR spectrometer system** developed in this PhD project (Sec. 2.1) was applied for simultaneous fast exhaust gas temperature measurements in the three optical ports of the exhaust duct of a large Diesel engine as stated in the objectives of the PhD project (Sec. 1.3.2).

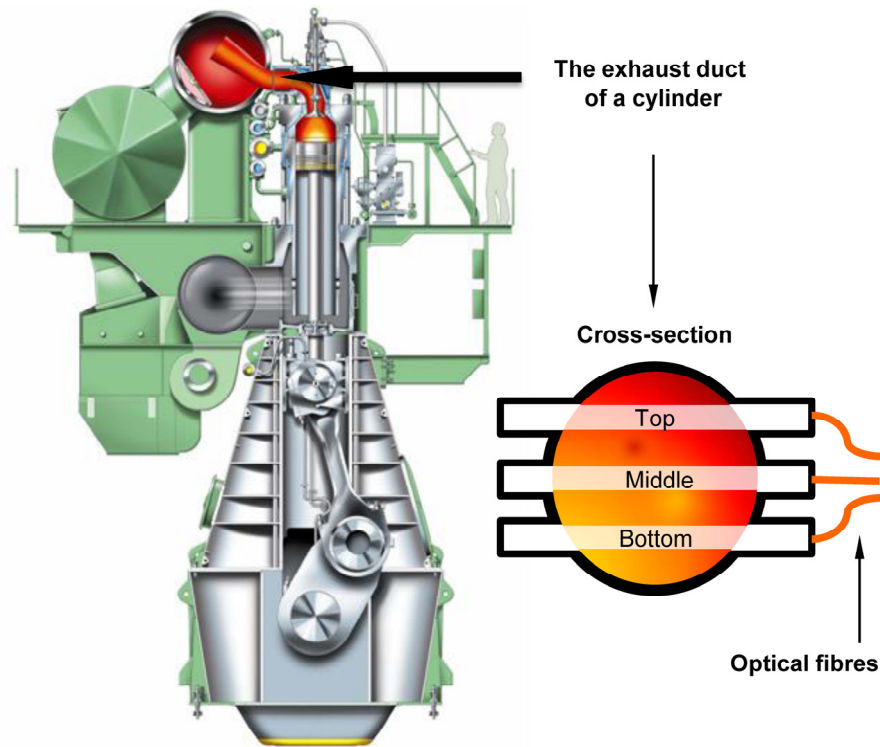
The application of the system on the large Diesel engine was also relevant to the EU-funded HERCULES-B project “High-efficiency engine with ultra-low emissions for ships” carried out under 7<sup>th</sup> Framework Program “Sustainable Surface Transport” (grant agreement SCP7-GA-217878) presented in reports [9].

The system was, in fact, developed with a view to this application on the large Diesel engine (Sec. 2.1.1).

### 2.4.2. Large Diesel Engine

The engine was a 2-stroke 4-cylinder test marine Diesel engine 4T50ME-X at MAN Diesel & Turbo, Copenhagen (Fig. 26). The simultaneous fast gas temperature measurements were performed across the exhaust gas flow in specially designed (by MAN Diesel & Turbo) exhaust duct having three optical access ports. The exhaust duct was mounted after the exhaust valve on one of the four cylinders. Optical absorption

path lengths were calculated to 151, 225 and 151 mm for the top, middle and bottom optical ports, respectively.

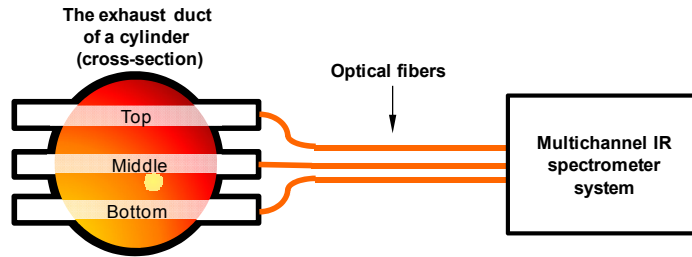


**Figure 26.** The 2-stroke 4-cylinder test marine Diesel engine 4T50ME-X at MAN Diesel & Turbo, Copenhagen.

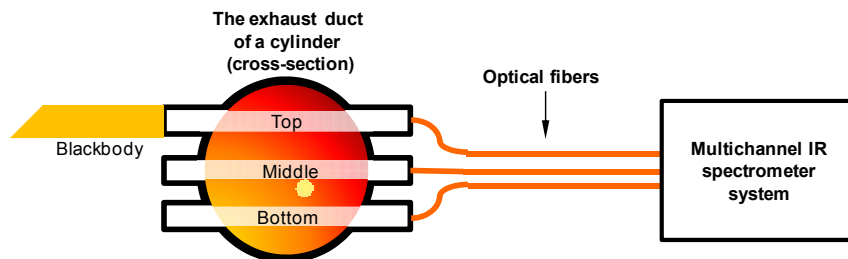
### 2.4.3. Measurement Layout

For the measurements on the exhaust duct the three optical fibers of the **multi-channel IR spectrometer system** were connected to the three fiber optical adaptors (Fig. 19, Sec. 2.3.3) which, in turn, were connected to the three optical access ports (top, middle, bottom) of the exhaust duct on its right side (if looking on the cross-section of the exhaust duct shown in Fig. 26, Sec. 2.4.2). The three optical access ports on the left side of the exhaust duct were used to introduce the reference source radiation for transmission measurements (see also Sec. 2.3.7 for the definition of transmission measurements).

A simplified diagram of the measurement layout for the case of emission measurements is shown in Fig. 27 (see also Sec. 2.3.7 for the definition of emission measurements). A diagram of transmission measurements is shown in Fig. 28. A handheld blackbody was used as the source of reference radiation.



**Figure 27.** The measurement layout for the case of emission measurements in the three optical ports of the exhaust duct of the large Diesel engine (Sec. 2.4.2, Fig. 26).



**Figure 28.** The measurement layout for the case of transmission measurements in the three optical ports of the exhaust duct of the large Diesel engine (Sec. 2.4.2, Fig. 26).

It can be seen that the emission measurements were performed simultaneously on the three ports and that was possible due to the special design of the **multichannel IR spectrometer system** (Sec. 2.1). However the transmission measurements were performed separately on each port since the handheld blackbody was used as the source of reference radiation. This can be improved by using a setup with three blackbodies mounted together. Also, each pair of emission and transmission measurements corresponding to the same port was taken at different time points. However, the simultaneous emission measurements alone can provide with essential information and together with the available transmission measurement results can be used to make certain assumptions on the stage of data analysis (Sec. 3.2.3).

It should also be noted that the transmission measurements conducted when the three fiber optical adaptors of the **multichannel IR spectrometer system** are all together mounted and connected to the three optical ports of the exhaust duct are much more time and engine fuel efficient than if a system with only one channel were used. In that case that would be necessary to stop the engine, disassemble a single fiber optical adaptor from one port, to mount it onto another port, then to run the engine again and to take a transmission measurement. That procedure, being time and engine fuel consuming, would have to be repeated for all the three ports. But with the **multichannel IR spectrometer system**, only the position of the handheld blackbody had to be changed (which takes a few seconds) in order to take the transmission measurements at different ports.

## 3. Theory and Methods

### 3.1. Line-by-line Modeling of Gas Spectra

A detailed description of the line-by-line modeling procedure presented in this section was implemented in software developed using the C++ Programming Language (Microsoft Visual Studio 2008) with the code given in Appendix A which was used for the line-by-line modeling of transmission spectra of CO<sub>2</sub> (+CO) performed in this PhD project with the results and discussion presented in Ref. [17] (Ref. [17] is included in this PhD thesis, Appendix B). It should be noted that this detailed description as well as the software code are not given in Ref. [17].

The effective, or measured, transmittance  $\tau(\nu_0, T_g)$  at any observed wave number  $\nu_0$  [cm<sup>-1</sup>] of the slab of a gas mixture at temperature  $T_g$  [K] can be calculated as

$$\tau(\nu_0, T_g) = \int_{\nu_1=0}^{\nu_2=\infty} ILS(\nu - \nu_0) e^{-\sigma_{true}(\nu, T_g) n L} d\nu, \quad (11)$$

(a combination of Eqs. (2) and (3) from Ref. [17]) where  $ILS(\nu - \nu_0)$  is the instrument line shape function [1/cm<sup>-1</sup>]; for an FTIR spectrometer used in the measurements in work [17], the ILS was given by

$$ILS(\nu - \nu_0) = \frac{1}{R_{spec}} \frac{\sin(\pi(\nu - \nu_0)/R_{spec})}{\pi(\nu - \nu_0)/R_{spec}} \quad (12)$$

with  $R_{spec} = 0.0625$  cm<sup>-1</sup> [17] ( $R_{spec}$  is the half of the nominal resolution of the spectrometer which was 0.125 cm<sup>-1</sup> [17]), the actual values of integration limits  $\nu_1$  and  $\nu_2$  used in (11) should make  $ILS(\nu_1 - \nu_0)$  and  $ILS(\nu_2 - \nu_0)$  sufficiently close to zero and

$\int_{\nu_1}^{\nu_2} ILS(\nu - \nu_0) d\nu$  sufficiently close to unity at a given value of  $\nu_0$  (a more detailed discussion on this issue can be found below);

$$[M] = \frac{c_V p}{k_B T_g} \quad (13)$$

is the total concentration [molecule/cm<sup>3</sup>] [76, eq. (2)] of a molecular species under consideration (e.g. CO<sub>2</sub>) in the slab of a gas mixture at temperature  $T_g$  [K] having length  $L$  [cm] (which is supposed to be sufficiently small, actually  $dL$ ),  $c_V$  is the volume part (volume ratio, or mole fraction) [fractions of 1] of the species under consideration,  $p$  is the total pressure in the slab of the gas mixture [atm],  $k_B$  is the Boltzmann constant;

$$\sigma_{true}(\nu, T_g) = \sum_{i=1}^{N_T} k_i(\nu, T_g) \quad (14)$$

is the true absorption cross-section (Sec 3.1 in [17]; cf. eq. (4) in [76]), where  $\nu$  is any wave number [cm<sup>-1</sup>],

$$k_i(\nu, T_g) = S_{T_g, i} g(\nu - \nu_i^*, T_g) \quad (15)$$

is the *monochromatic* absorption coefficient (Appendix A in [77]) (not to be confused with the absorption coefficient  $k(\nu_0, T_g)$  (used throughout Sec. 3.3) which is referred to as without the adjective *monochromatic*),

$$S_{T_g, i} = S_i \frac{Q(T_{ref}) \exp(-c_2 E_i / T_g) [1 - \exp(-c_2 \nu_i / T_g)]}{Q(T_g) \exp(-c_2 E_i / T_{ref}) [1 - \exp(-c_2 \nu_i / T_{ref})]} \quad (16)$$

is the temperature corrected line intensity [ $\text{cm}^{-1}/(\text{molecule cm}^{-2})$ ] of the  $i$ -th transition ( $i = 1, 2, \dots, N_T$  is the sequence number of a current transition (or spectral *line*) under consideration and  $N_T$  is the total number of transitions under consideration, see also a more detailed discussion on the number of transition under consideration below),  $S_i$  is the line intensity [ $\text{cm}^{-1}/(\text{molecule cm}^{-2})$ ] of the  $i$ -th transition at a reference temperature  $T_{ref} = 296$  K ( $S_i$  is taken from the database, see also Fig. 29),  $Q(T)$  is the total internal partition sum at temperature  $T$ , the values of this function for the four most abundant isotopes of  $\text{CO}_2$  and for the other molecular species on the HITRAN/HITEMP databases can be calculated using the Fortran programs for Partition Sums [78],  $c_2$  is the second radiation constant  $= hc/k = 1.4388$  cm K (Appendix A in [77]),  $E_i$  is the lower state energy of the  $i$ -th transition [ $\text{cm}^{-1}$ ] (taken from the database, see also Fig. 29),  $\nu_i$  is the frequency of the  $i$ -th transition (also taken from the database) (not to be confused with  $\nu_0, \nu_1$  and  $\nu_2$  mentioned above which are NOT  $\nu_i$  at  $i = 0, 1, 2$  respectively),

$g(\nu - \nu_i^*, T_g)$  is the line shape function caused by Doppler broadening (Gaussian profile), pressure broadening (Lorentzian profile), or a combination of the two (Voigt profile) [79]. The choice of the line profile depends on gas temperature and pressure. A detailed discussion on this issue can be found in [79]. However, a Voigt profile is the most general form of a line shape function as it combines both Gaussian and Lorentzian line shapes. The Voigt profile cannot be evaluated in closed analytical form and a large number of numerical algorithms have been developed in the past [80]. One can use an algorithm presented in e.g. [81]. In this work, an empirical approximation to the Voigt profile as given by (Eqs. (6) and (7) in [79]) was used:

$$\begin{aligned} \frac{g_V(\nu - \nu_i^*, \gamma_{p,i}, \gamma_{V,i})}{g_{V,\max}} &= \left(1 - \frac{\gamma_{p,i}}{\gamma_{V,i}}\right) \exp\left(-2.772 \left(\frac{\nu - \nu_i^*}{2\gamma_{V,i}}\right)^2\right) + \\ &+ \left(\frac{\gamma_{p,i}}{\gamma_{V,i}}\right) \frac{1}{1 + 4 \left(\frac{\nu - \nu_i^*}{2\gamma_{V,i}}\right)^2} + 0.016 \left(1 - \frac{\gamma_{p,i}}{\gamma_{V,i}}\right) \left(\frac{\gamma_{p,i}}{\gamma_{V,i}}\right) \times \\ &\times \left[ \exp\left(-0.4 \left(\frac{|\nu - \nu_i^*|}{2\gamma_{V,i}}\right)^{2.25}\right) - \frac{10}{10 + \left(\frac{|\nu - \nu_i^*|}{2\gamma_{V,i}}\right)^{2.25}} \right] \end{aligned} \quad (17)$$



$$\mathcal{G}_{V,\max} = \frac{1}{2\gamma_{V,i} \left( 1.065 + 0.447 \left( \frac{\gamma_{p,i}}{\gamma_{V,i}} \right) + 0.058 \left( \frac{\gamma_{p,i}}{\gamma_{V,i}} \right)^2 \right)} \quad (18)$$

where

$$\gamma_{p,i} = \gamma_{air,i} \left( \frac{T_{ref}}{T_g} \right)^{n_{air,i}} (p - p_s) + \gamma_{self,i} \left( \frac{T_{ref}}{T_g} \right)^{n_{self,i}} p_s \quad (19)$$

is the temperature and pressure corrected line half-width at half-maximum (HWHM) [ $\text{cm}^{-1}$ ] of the  $i$ -th transition,  $\gamma_{air,i}$  is the air-broadened HWHM [ $\text{cm}^{-1}/\text{atm}$ ] at  $T_{ref} = 296$  K and a reference pressure 1 atm ( $\gamma_{air,i}$  is also taken from the database),  $n_{air,i}$  is the coefficient of temperature dependence of  $\gamma_{air,i}$  (also taken from the database),  $\gamma_{self,i}$  is the self-broadened HWHM [ $\text{cm}^{-1}/\text{atm}$ ] at  $T_{ref} = 296$  K and pressure 1 atm (also taken from the database),  $n_{self,i}$  is the coefficient of temperature dependence of  $\gamma_{self,i}$  (available in the CDS-4000 [16] and CDS-HITEMP [14] databases, for the other databases  $n_{self,i}$  is assumed to be equal to  $n_{air,i}$ ),  $p$  is the total pressure in the slab of the gas mixture [atm],

$$p_s = c_V p$$

is the partial pressure [atm] of the molecular species under consideration (e.g.  $\text{CO}_2$ ) in the slab of the gas mixture,  $c_V$  is the volume part (volume ratio, or mole fraction) [fractions of 1] of the species under consideration;

$$\gamma_{V,i} \approx 0.5 \left( 1.0692 \gamma_{p,i} + \sqrt{0.86639 \gamma_{p,i}^2 + 4 \gamma_{D,i}^2} \right) \quad (20)$$

is the HWHM of the Voigt line shape (eq. (14) in [80]) for the  $i$ -th transition,

$$\gamma_{D,i} = v_i^* \sqrt{\frac{2 \ln 2 R_{gas} T_g}{M_w c^2}} \quad (21)$$

is the HWHM of the Doppler feature [ $\text{cm}^{-1}$ ] (eq. (3) in [79]) for the  $i$ -th transition,  $R_{gas}$  is the ideal gas constant,  $M_w$  is the molecular weight [kg/mol] of the isotope of the molecular species under consideration (e.g.  $M_w$  of  $^{12}\text{C}^{16}\text{O}_2$  is 0.04398982923912 kg/mol),  $c$  is the speed of light in vacuum,

$$v_i^* = v_i + \delta_{air,i} p \quad (22)$$

is the pressure-shift corrected frequency of the  $i$ -th transition [ $\text{cm}^{-1}$ ],  $\delta_{air,i}$  is the air-broadened pressure shift [ $\text{cm}^{-1}/\text{atm}$ ] of the transition frequency at  $T_{ref} = 296$  K and a reference pressure 1 atm ( $\delta_{air,i}$  is taken from the database, see also Fig. 29).

The format of the databases, such as HITEMP-2010 [13], CDS-1000 [15], is shown in Fig. 29. The meaning of the notation can be found in the preceding text except  $Mol_i$  which is the molecular species identification number ( $Mol_i = 2$  for  $\text{CO}_2$ ) and  $Iso_i$  which is the isotope identification number (see Appendix A in [77] or table 6 in [82] for details). The spectroscopic data in the databases is a text list sorted with increasing transition frequency  $v_i$ . Databases, such as HITEMP-2010, CDS-1000, consist of many text files.

The calculation of  $\tau(v_0, T_g)$  according to the above given procedure is a suggestion of a possible approach to the line-by-line modeling (cf. Sec 2.1 in [20]). The name *line-by-line* is given because the spectroscopic parameters for every transition (or spectral *line*) are used in the calculations. These spectroscopic parameters are given in the databases as shown in Fig. 29. In the procedure described above, only 7 parameters are in use (or 8 parameters if the CDSD-4000 database [16] is used, see above) and they are highlighted each with a certain color in the figure. Databases contain also other spectroscopic parameters which are used in the other models. This issue is out of the scope of this work.

Mol <sub>i</sub>	Iso <sub>i</sub>	$\nu_i$	$S_i$	$\gamma_{air,i}$	$\gamma_{self,i}$	$E_i$	$n_{air,i}$	$\delta_{air,i}$	...
21	2250	.000290	3.993E-38	2.284E-01	.05580	.067	8560.5169	0.62	-.004602 ...
21	2250	.000510	2.346E-34	6.508E-01	.06660	.065	6907.5099	0.71	-.002959 ...
22	2250	.000510	9.572E-46	7.018E+02	.07120	.065	12909.5251	0.83	-.005032 ...
25	2250	.000960	2.744E-33	5.725E+02	.06890	.079	5507.1862	0.71	-.003741 ...
21	2250	.000990	7.019E-36	1.007E-04	.07310	.095	5581.6607	0.68	-.002050 ...
22	2250	.001190	1.770E-26	9.398E-01	.07630	.098	2045.0823	0.69	-.002648 ...
22	2250	.001200	5.028E-43	2.982E+02	.07070	.065	11439.5954	0.82	-.004995 ...
27	2250	.001390	1.191E-34	1.829E+02	.06800	.087	5489.9053	0.68	-.003266 ...
25	2250	.001430	1.552E-37	1.877E+02	.05980	.062	7421.5829	0.65	-.004763 ...
22	2250	.001700	3.485E-36	8.124E-03	.07900	.112	5405.0404	0.64	-.000485 ...
...	...	...	...	...	...	...	...	...	...

**Figure 29.** Format of the spectroscopic databases, e.g. HITEMP-2010 [13], CDSD-1000 [15]. The meaning of the notation can be found in the text (Sec. 3.1). The CDSD-4000 [16] and CDSD-HITEMP [14] databases include one more parameter,  $n_{self,i}$ , the coefficient of temperature dependence of  $\gamma_{self,i}$ , which is not shown in the figure (when databases other than CDSD-4000 or CDSD-HITEMP are used,  $n_{self,i}$  is assumed to be equal to  $n_{air,i}$ ).

The calculation of  $\tau(v_0, T_g)$  at a certain  $v_0$  requires to determine the finite limits  $v_1$  and  $v_2$  of the integration in (11). They are determined based on the condition that  $ILS(v_1 - v_0)$  and  $ILS(v_2 - v_0)$  should be sufficiently close to zero and  $\int_{v_1}^{v_2} ILS(v - v_0) dv$  should be sufficiently close to unity at a given value of  $v_0$ .

Once  $v_1$  and  $v_2$  are determined, it is necessary to determine the range of values for the transition frequency  $\nu_i$  ( $\nu_i$  should not to be confused with  $v_0, v_1$  and  $v_2$  mentioned above which are NOT  $\nu_i$  at  $i = 0, 1, 2$  respectively) or, in other words, to choose the transitions for the line-by-line calculations from the database files. That will also give  $N_T$  which is the total number of chosen transitions. The range for  $\nu_i$  is chosen based on the values of  $v_1$  and  $v_2$  and the wing cut-off of the line profile as discussed in the following.

A line profile  $g(v - \nu_i^*, T_g)$ , e.g. such as given by Eqs. (17, 18), determines the contribution of  $i$ -th transition having frequency  $\nu_i$  (or pressure-shift corrected frequency  $\nu_i^*$ ) to  $\sigma_{true}(v, T_g)$  at a current value of  $v$  ( $v_1 \leq v \leq v_2$ ) that is used in the integration in (11). But this contribution never reaches zero even at  $\nu_i^* \gg v$  and  $\nu_i^* \ll v$ . It is rea-

sonable to assume that at values of  $\nu_i^*$  sufficiently different from  $\nu$ , or such that  $|\nu - \nu_i^*| > \Delta\nu_{\text{wing cutoff}}$ , a line shape function turns to zero. In this work,  $\Delta\nu_{\text{wing cutoff}}$  was determined from the following condition

$$g_V(\Delta\nu_{\text{wing cutoff}}, \max_i \gamma_{p,i}, \max_i \gamma_{V,i}) = C_{\text{wing cutoff}} g_V(0, \max_i \gamma_{p,i}, \max_i \gamma_{V,i}) \quad (23)$$

where a constant  $C_{\text{wing cutoff}}$  was taken as, e.g.,  $C_{\text{wing cutoff}} = 10^{-3}$  or other value. In such a way

$$g_V(\nu - \nu_i^*, \gamma_{p,i}, \gamma_{V,i}) = 0 \quad \text{for } |\nu - \nu_i^*| > \Delta\nu_{\text{wing cutoff}} \quad (24)$$

that is also to be accounted for in addition to Eqs. (17, 18).

The range of values for the transition frequency  $\nu_i$  can now be determined from the condition

$$\nu_1 - \Delta\nu_{\text{wing cutoff}} \leq \nu_i^* = \nu_i + \delta_{\text{air},i} p \leq \nu_2 + \Delta\nu_{\text{wing cutoff}} \quad (25)$$

where Eq. (22) is taken into account.

The described above line-by-line procedure for the calculation of the effective (or measured) transmittance spectrum  $\tau(\nu_0, T_g)$  of a mixture of CO<sub>2</sub> and CO using the line-by-line parameters from the HITEMP-1995 [12], HITEMP-2010 [13], CSDS-HITEMP [14] and CSDS-4000 [16] databases is implemented in the software developed in this PhD project using the C++ Programming Language (Microsoft Visual Studio 2008) as a primary tool and MATLAB as an auxiliary tool. The software was used for the line-by-line modeling of transmission spectra of CO<sub>2</sub> (+CO) with the results and discussion presented in Ref. [17] (Ref. [17] is included in this PhD thesis, Appendix B). The software code is given in Appendix A in this PhD thesis. It should be noted that the software code is not given in Ref. [17].

## 3.2. Gas Temperature Measurements on the Large Diesel Engine

### 3.2.1. Remarks

This section gives theoretical background related to the gas temperature measurements in the three optical ports (top, middle, bottom) of the exhaust duct of the large Diesel engine (experimental details are given in Sec. 2.4) and is designed with a reference to that application.

### 3.2.2. The Spectral Emission-Absorption Method of Gas Temperature Measurement

The emission measurements (Fig. 27, Sec. 2.4.3) yield the spectral radiances  $[W / m^2 \text{ sr m}^{-1}]$  for each port  $N_{\nu}^{\text{Top}}(T^{\text{Top}}(t))$ ,  $N_{\nu}^{\text{Mid}}(T^{\text{Mid}}(t))$ ,  $N_{\nu}^{\text{Bot}}(T^{\text{Bot}}(t))$  at any wave number  $\nu$   $[\text{cm}^{-1}]$  as functions of gas temperature  $T^{\text{Top}}(t)$ ,  $T^{\text{Mid}}(t)$ ,  $T^{\text{Bot}}(t)$  [K] at each

port (top, middle, bottom) which are the functions of time  $t$  [s]. The task is to calculate the temperatures (at every time point) for each port.

The transmission measurements are also necessary for the calculation. The simplified diagram of the transmission measurements is shown in Fig. 28 (Sec. 2.4.3). Transmission measurements yield spectral absorptances  $\alpha_{\nu}^{Top}(T^{Top}(t))$ ,  $\alpha_{\nu}^{Mid}(T^{Mid}(t))$ ,  $\alpha_{\nu}^{Bot}(T^{Bot}(t))$ . The spectral absorptance  $\alpha_{\nu}(T)$  (see Eq. (2.12) in [72]) is the portion of incident energy absorbed at wave number  $\nu$  by the gas at temperature  $T$  [K]. It is given by

$$\alpha_{\nu}(T) = 1 - \frac{N_{\nu}^{sg}(T) - N_{\nu}^g(T)}{N_{\nu}^0} \quad (26)$$

where  $N_{\nu}^{sg}(T)$  is the spectral radiance received by the detector in a transmission measurement,  $N_{\nu}^g(T)$  is the spectral radiance of the gas obtained from the emission measurement, the difference  $N_{\nu}^{sg}(T) - N_{\nu}^g(T)$  gives the spectral radiance corresponding to the transmitted by the gas radiant energy of the reference source,  $N_{\nu}^0$  is the spectral radiance of the reference source radiation. Note that the ratio  $N_{\nu}^{sg}(T) - N_{\nu}^g(T) / N_{\nu}^0$  gives the spectral transmittance (the portion of incident energy transmitted by the gas) at wave number  $\nu$ .

Assuming local thermodynamic equilibrium, the relationship between the emission and absorption of radiant energy at wave number  $\nu$  is given by Kirchhoff's law (Eq. (2.16) in [72]),

$$\frac{N_{\nu}^g(T)}{\alpha_{\nu}(T)} = N_{\nu}^{Planck}(T) \quad (27)$$

where

$$N_{\nu}^{Planck}(T) = \frac{C_1}{\pi} \frac{\nu^3}{\exp(C_2\nu/T) - 1} \quad (28)$$

is the Planck function (Eq. (2.15) in [72] converted from a function of wavelength to a function of wave number),  $C_1 = 3.7418 \times 10^{-16} \text{ kg} \cdot \text{m}^4/\text{s}^3$  is the first radiation constant,  $C_2 = 1.4388 \times 10^{-2} \text{ m} \cdot \text{K}$  is the second radiation constant.

The Planck function depends only on temperature. Therefore once  $N_{\nu}^{sg}(T)$  and  $N_{\nu}^g(T)$  are measured and  $N_{\nu}^0$  is known the temperature can be calculated using Eqs. (26, 27, 28). This is the spectral emission-absorption method of gas temperature measurement [72, pp. 25–30]. If Eqs. (26, 27, 28) are applied to a path with a non-isothermal profile, the wave-number-dependent result is a weighted average of the temperatures along the path [72, p. 65]. The method is applicable at any wave number at which the gas radiates [72, p. 26].

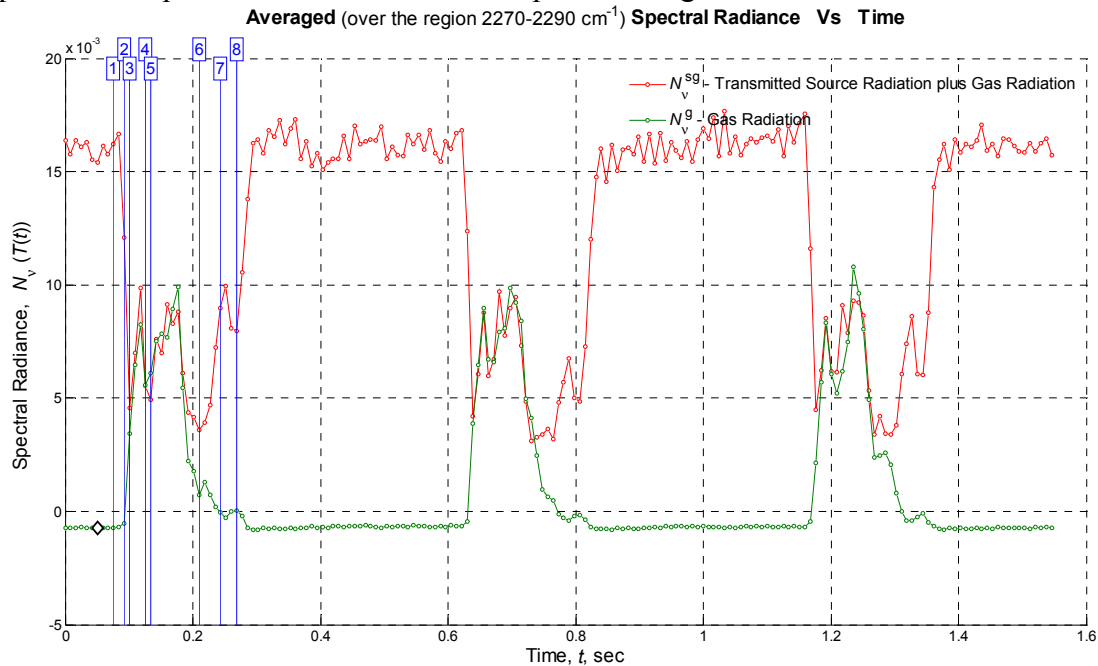
Tomographic algorithms for gas temperature profile reconstruction developed in this PhD project (Sec. 3.3) are, in fact, the application of the equations on which the spectral emission-absorption method is based [83] to the case of a non-isothermal gas profile.

### 3.2.3. The Assumed Value of Spectral Absorptance

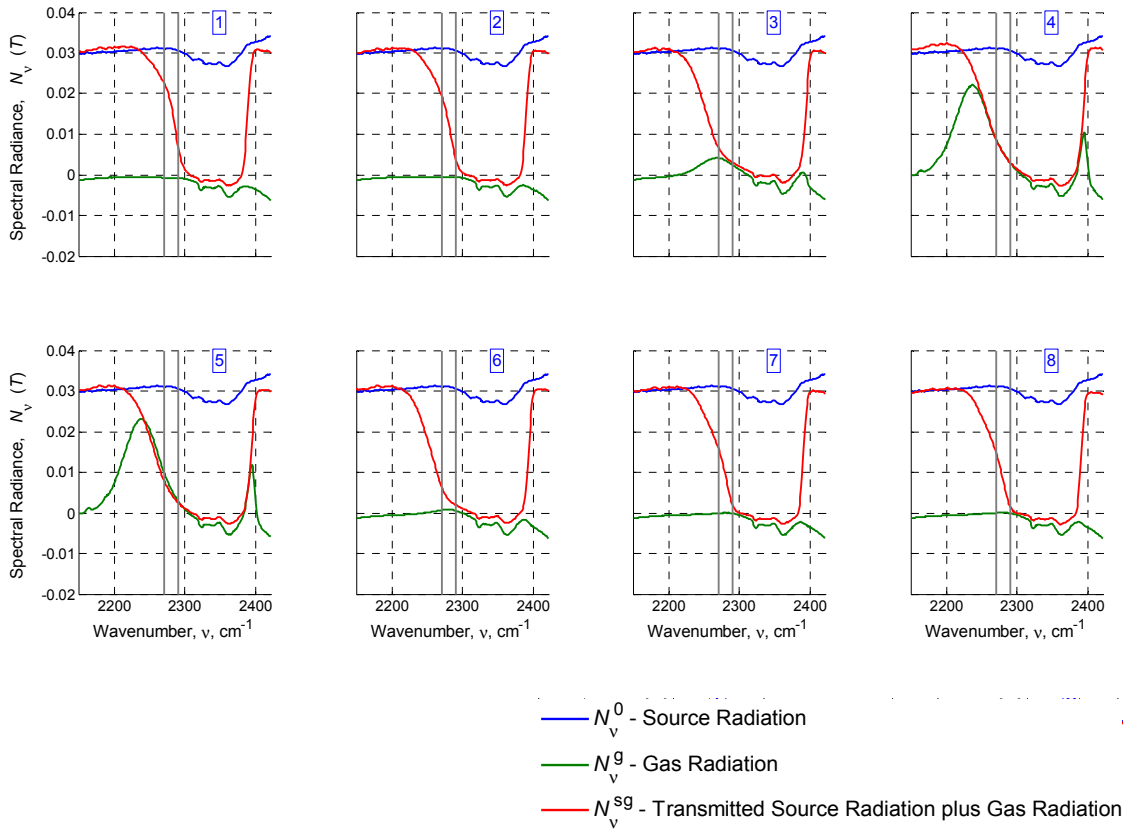
It was not possible to take the emission and transmission measurements simultaneously (Sec. 2.4.3) although Eq. (26) implies that  $N_v^{sg}(T)$  and  $N_v^g(T)$  should be measured at the same time to ensure that the gas is being in the same state during both emission and transmission measurements.

An example of transmission and emission signals as functions of time for the middle port in the exhaust duct is shown in Fig. 30. These are the averaged with respect to wave number  $\nu$  over the range 2270–2290  $\text{cm}^{-1}$  values of  $N_v^{sg}(T(t_i))$  and  $N_v^g(T(t_i))$  taken at each time point  $t_i$  and plotted versus time  $t$ .

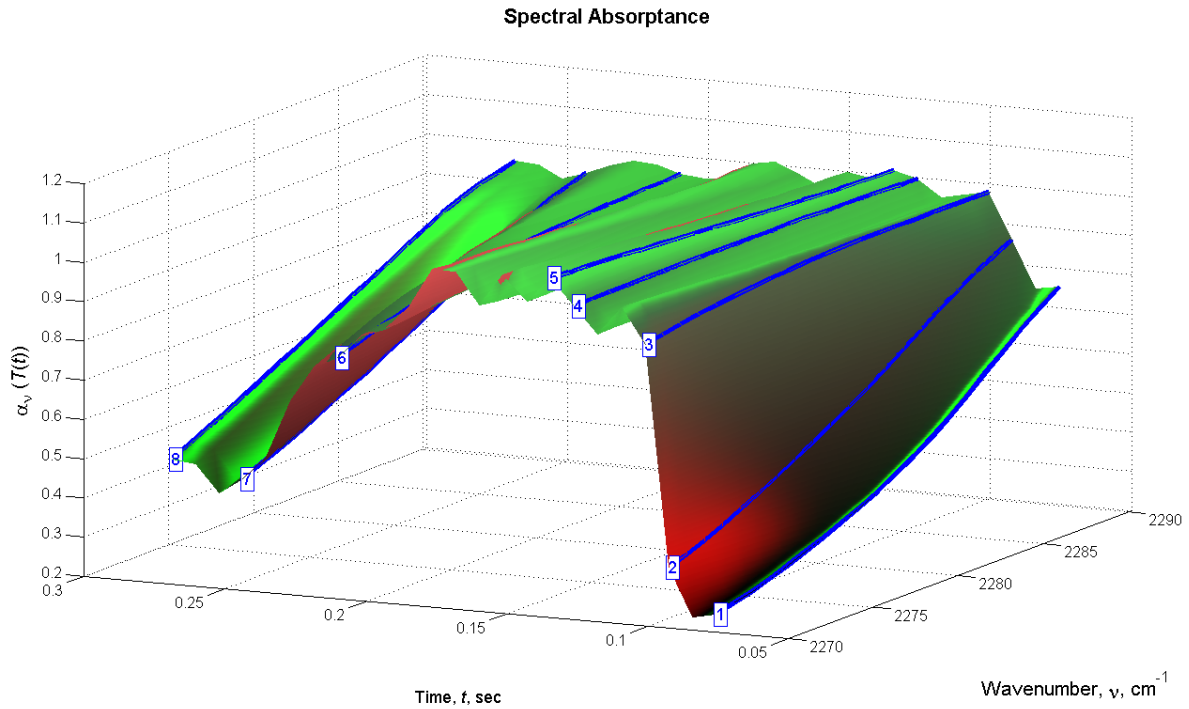
The emission and transmission signals were aligned manually with respect to the time axis and the result of such alignment is, in fact, given in Fig. 30. It is now possible to calculate the spectral absorptance using Eq. (26) at each time point. The spectral absorptance as a function of both time and wave number is shown in Fig. 32. The time range in Fig. 32 corresponds to the time range of Fig. 30 between the lines labeled “1” and “8”. The spectral range in Fig. 32 corresponds to the range denoted by the vertical gray lines in Fig. 31 (2270–2290  $\text{cm}^{-1}$ ). The blue curves in Fig. 32 are the curves of the spectral absorptance at the 8 selected time points of Fig. 30.



**Figure 30.** An example of the transmission and emission signals for the middle port of the exhaust duct of the large Diesel engine (see also Fig. 27 and Fig. 28, Sec. 2.4.3). The vertical blue lines denote the selected spectral sections shown in Fig. 31. The signals were taken at the frame rate of the IR camera of 119 Hz.



**Figure 31.** The selected spectral sections corresponding to the labeled time points of Fig. 30. The vertical gray lines show the 2270-2290  $\text{cm}^{-1}$  spectral range over which the spectral radiances were averaged to give the points on the plot of Fig. 30.



**Figure 32.** The spectral absorbance as a function of both time and wave number. The time range here corresponds to the one of Fig. 30 between the lines labeled “1” and “8”. The spectral range corresponds to the range denoted by the vertical gray lines in Fig. 31 ( $2270\text{--}2290\text{ cm}^{-1}$ ). The blue curves are the curves of the spectral absorbance at the 8 selected time points of Fig. 30.

It can be seen that the spectral absorbance curves are confined between the levels of 0.8 and 1 at the time points when the emission signal is high enough (the peaks on the emission signal curve in Fig. 30). This is also the case for the other two ports (top and bottom) of the exhaust duct. Therefore, the value of spectral absorbance was assumed to be equal to 0.9 (within the considered spectral range  $2270\text{--}2290\text{ cm}^{-1}$ ) at all the time points and for all the three ports in the exhaust duct, i. e.

$$\alpha_{\nu}^{Top}(T(t)) = \alpha_{\nu}^{Mid}(T(t)) = \alpha_{\nu}^{Bot}(T(t)) = 0.9 = \text{const}, \quad 2270 \leq \nu \leq 2290\text{ cm}^{-1}. \quad (29)$$

It should be noted that at the time points when the emission signal is low (the regions between the peaks on the emission signal curve in Fig. 30) the temperature cannot be obtained with as high accuracy as for the points when the emission signal is high due to low signal-to-noise ratio. Hence the value of spectral absorbance does not play a significant role in the calculations at those time points (the value of spectral absorbance at those time points is about 0.1).

The  $2270\text{--}2290\text{ cm}^{-1}$  range was chosen from the conditions of the maximum emission signal from the exhaust gas, the maximum absorption by the exhaust gas, and the minimum absorption by cold  $\text{CO}_2$  present in the optical path of the system. The range  $2270\text{--}2290\text{ cm}^{-1}$  lies within the  $4.3\text{ }\mu\text{m}$  band of  $\text{CO}_2$ . In that range, cold  $\text{CO}_2$  in the optical path of the system has weak absorption and at the same time the absorption by hot  $\text{CO}_2$  in the exhaust gas is sufficiently strong. Also, as can be seen from Fig. 31 (and Fig. 41), the emission from  $\text{CO}_2$  in the exhaust gas in that range is strong.

### 3.2.4. Gas Temperature Calculation

The temperature can be calculated at each time point and at each wave number  $\nu$  (within the considered spectral range 2270–2290  $\text{cm}^{-1}$ ) using Kirchhoff's law (27) and the Planck radiation law (28)

$$T_\nu(t) = \frac{C_2 \nu}{\ln\left(\frac{C_1}{\pi} \frac{\nu^3}{N_\nu^g(t)} 0.9 + 1\right)} \quad (30)$$

where 0.9 is the assumed value of spectral absorptance. Thus, the temperature  $T_\nu(t)$  is calculated from the corresponding value of spectral radiance  $N_\nu^g(t)$  of the gas at each wave number  $\nu$  (within the considered spectral range 2270–2290  $\text{cm}^{-1}$ ) and at every time point  $t$ . The final value of gas temperature at a given time point  $t$  is the average of the calculated using Eq. (30) temperature values at the wave numbers within the considered spectral range at that given time point  $t$  (the wave numbers were [ $\text{cm}^{-1}$ ] 2270; 2271; 2272.1; 2273.1; 2274.2; 2275.3; 2276.3; 2277.4; 2278.5; 2279.5; 2280.6; 2281.7; 2282.7; 2283.8; 2284.9; 2285.9; 2287; 2288.1; 2289.2; 2290.2). In other words, the value of gas temperature at a given time point  $t$  is the average of the temperature values calculated using Eq. (30) at the above given wave numbers from the corresponding values of  $N_\nu^g(t)$  at those wave numbers and at that given time point  $t$ .

The plots of gas temperature in the three ports (top, middle, bottom) of the exhaust duct calculated using the approach described above are presented in Sec. 4.2.3.

## 3.3. Tomographic Algorithms for Gas Temperature Profile Reconstruction

### 3.3.1. Introduction and Remarks

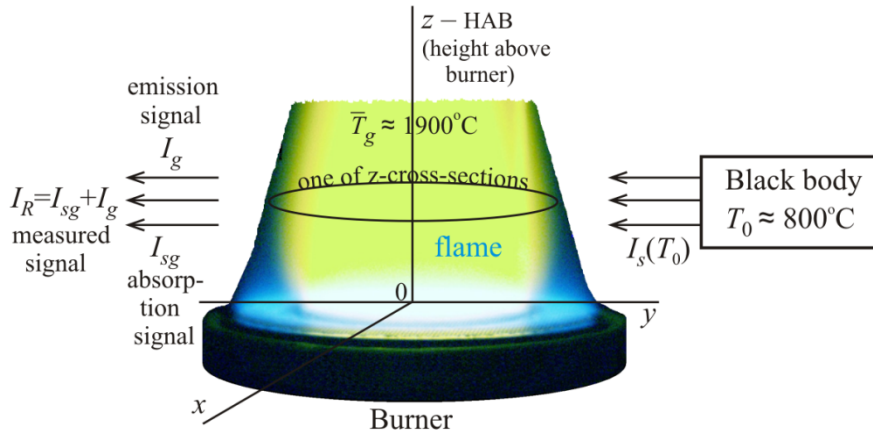
The work on this part of the PhD project (as well as of the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8]) was conducted in collaboration with Prof. Valery Sizikov from National Research University of Information Technologies, Mechanics and Optics (St Petersburg, Russia) who is a well-recognized expert in solution of inverse ill-posed problems (see, e.g., Ref. [88]: Petrov Yu. P., Sizikov V. S. *Well-Posed, Ill-Posed, and Intermediate Problems with Applications*. VSP, Leiden-Boston, 2005, 234 pp.).

Prof. V. S. Sizikov was responsible for developing the tomographic methodology and algorithms for gas temperature profile reconstruction. The methodology is presented in this section of the PhD thesis to give general acquaintance only.



### 3.3.2. Problem Statement

Consider the burner producing a premixed laminar flame [74]. The flame is irradiated by a black body radiation source at temperature  $T_0 = 894.4 \text{ }^\circ\text{C}$  (1167.6 K) (see fig. 33)



**Figure 33.** The burner producing the pre-mixed laminar flame see (Sec. 2.2). The blackbody temperature during the FTIR measurements was  $T_0 = 894.4 \text{ }^\circ\text{C}$  (1167.6 K) (Sec. 2.3.3).

At any wave number  $\nu$ , the spectral radiance of a blackbody  $I_s(\nu, T_0)$  is given by the Planck radiation law [72, p. 16], [73]

$$I_s(\nu, T_0) = \frac{2hc^2\nu^3}{e^{h\nu/k_B T_0} - 1}$$

where  $h$  is the Planck constant,  $c$  is the velocity of light in vacuum,  $\nu = 1/\lambda$  is a wave number,  $k_B$  is the Boltzmann constant,  $\lambda$  is the wavelength. All further considerations will concern the monochromatic radiation at some fixed wave number  $\nu$ . So we will omit  $\nu$  in order to simplify the notation. The Planck function will then be written as

$$I_s(T_0) = \frac{2hc^2\nu^3}{e^{h\nu/k_B T_0} - 1}. \quad (31)$$

The radiation emitted by the blackbody penetrates the flame (which has the temperature around  $1900^\circ\text{C}$ ) and the intensity of this radiation is diminishing due to absorption by the gas of the flame. The gas is not only absorbing the radiation from the source but emitting its own radiation. This is the *emission signal*, which is proportional to the radiant energy emitted by the gas. This radiant energy is given by *spectral radiance*  $I_g$ .  $I_{sg}$  on fig. 33 is the intensity of the radiation from the source transmitted through the gas alone. This is the *absorption signal*, which is proportional to the amount of source radiant energy transmitted through the gas. This radiant energy is given by spectral radiance  $I_{sg}$ .  $I_R$  on fig. 33 is the total intensity received by the detector which is equal to  $I_R = I_{sg} + I_g$ .

The above discussion was about integral properties of the flame. Our task is to find local properties of the flame e.g.  $T_g(x, y, z)$  and others.

### 3.3.3. About Absorption

The absorption is described by the Lambert–Beer law which has the following differential form

$$dI = -\sigma(T_g)[M]I dl \quad (32)$$

where  $I$  is the current intensity of a beam penetrating a layer of the gas,  $\sigma(T_g)$  is the absorption cross-section,  $[M]$  is the total concentration of absorbing particles in the layer of the gas under consideration,  $l$  is the coordinate along the beam,  $T_g$  is the temperature of the gas in the layer.

The absorption coefficient can be written as  $k(T_g) = \sigma(T_g)[M]$ . If the layer is homogeneous (i. e.  $T_g$  and  $[M]$  are constant within the layer) then the Lambert–Beer law can be written in the integral form as

$$I = I_0 \exp(-k(T_g)L) = I_0 \exp(-\sigma(T_g)[M]L), \quad (33)$$

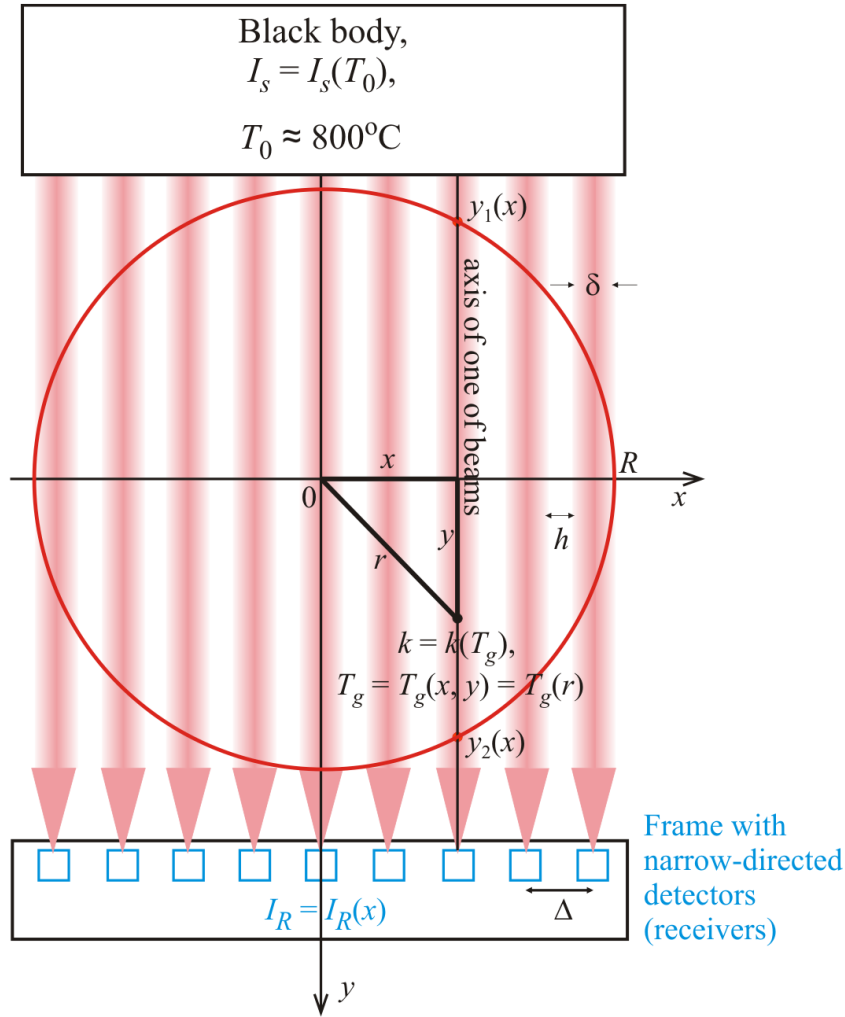
where  $I_0$  is the initial intensity of the beam (before it penetrates the layer),  $I$  is the final intensity of the beam (as it emerges from the layer),  $L$  is the thickness of the homogeneous layer.

Let us consider the two schemes of the burner flame diagnostics.

### 3.3.4. Parallel Scanning

In this scheme, the parallel scanning by IR beams takes place and the axial symmetry of the flame is taken into account. Diagnostics scheme I (parallel scanning scheme) is given on fig. 34 which shows a view from top on a certain cross-section of the flame at some value of the vertical coordinate  $z$ .

On fig. 34:  $N$  beams are shown by pink lines,  $y_1(x)$  is the coordinate where a beam enters the flame,  $y_2(x)$  is the coordinate where the beam emerges from the flame,  $I_R(x)$  are the intensities received by the detectors.



**Figure 34.** Parallel Scanning Scheme of Measurements. This figure is explanatory. See Fig. 23 and Sec. 2.3.6 for the actual layout and values. The blackbody temperature during the measurements was  $T_0 = 894.4 \text{ }^\circ\text{C}$  (1167.6 K) (Sec. 2.3.3).

### 3.3.5. The Differential Equation of Emission and Absorption

In a hot gas in local thermodynamic equilibrium, the spectral irradiance by a one-dimensional beam of monochromatic radiation varies according to the equation [83]

$$dI(x, y) = -k(T_g)[I(x, y) - I_s(T_g)]dy \quad (34)$$

(the monochromatic radiative transfer equation without scattering [84]), where  $T_g = T_g(x, y) = T_g(r)$  is the gas temperature (the temperature profile),  $r$  is the distance from the vertical axis of symmetry of the flame (axis  $z$ ),

$$I_s(T_g) = \frac{2hc^2\nu^3}{e^{hc\nu/k_B T_g} - 1} \quad (35)$$

is the Planck function at the temperature of the gas  $T_g$  (assuming local thermodynamic equilibrium).

Here  $I$  is the intensity of a beam which has been emitted by the blackbody radiation source and which is passing through the flame along coordinate  $y$  at some  $x$ . At the point where the beam enters the flame, the 1<sup>st</sup> boundary condition holds, i. e.

$$I(x, y_1(x)) = I_s(T_0) \quad (36)$$

(see (31) and fig. 34) and the 2<sup>nd</sup> boundary condition holds at the point where the beam emerges from the flame

$$I(x, y) = I(x, y_2(x)) = I_R(x). \quad (37)$$

The first term  $-k(T_g)I(x, y)dx$  of the right-hand side in (34) is the decrease in irradiance due to absorption along the path  $dx$ , as given by Lambert's law, and the second term  $k(T_g)I_s(T_g)dx$  is the increase in irradiance due to the emittance of the hot gas within  $dx$ , as given by Kirchoff's law. As mentioned above, scattering is assumed to be negligible.

Equation (34) is a differential equation with respect to  $I(x, y)$  with given  $k(T_g)$  and  $T_g = T_g(x, y) = T_g(r)$ . It can be solved analytically [83], [85, p. 308]

$$I(x, y) = \exp\left(-\int k(T_g(x, y)) dy\right)C + \exp\left(-\int k(T_g(x, y)) dy\right) \cdot \int k(T_g(x, y)) I_s(T_g(x, y)) \exp\left(\int k(T_g(x, y')) dy'\right) dy, \quad (38)$$

where  $C$  is the integration constant,  $C = I_s(T_0)$ . Dividing by  $C = I_s(T_0)$  and switching to definite integrals one can get

$$\begin{aligned} \frac{I(x, y)}{I_s(T_0)} &= \exp\left(-\int_{y_1(x)}^y k(T_g(x, y')) dy'\right) + \\ &+ \left\{ \int_{y_1(x)}^y k(T_g(x, y')) \frac{I_s(T_g(x, y'))}{I_s(T_0)} \exp\left(\int_{y_1(x)}^{y'} k(T_g(x, y'')) dy''\right) dy' \right\} \times \\ &\times \exp\left(-\int_{y_1(x)}^y k(T_g(x, y')) dy'\right). \end{aligned} \quad (39)$$

This is a solution for the intensity  $I$  of the beam as a function of coordinate  $y$  at some value of  $x$  and at some known (given) values of  $k(T_g)$  and  $T_g = T_g(x, y) = T_g(r)$ . However, measured by the detector value of  $I$  is more interesting from the practical point of view

$$\begin{aligned} \frac{I_R(x)}{I_s(T_0)} &= \exp\left(-\int_{y_1(x)}^{y_2(x)} k(T_g(x, y)) dy\right) + \\ &+ \left\{ \int_{y_1(x)}^{y_2(x)} k(T_g(x, y)) \frac{I_s(T_g(x, y))}{I_s(T_0)} \exp\left(\int_{y_1(x)}^y k(T_g(x, y')) dy'\right) dy \right\} \times \\ &\times \exp\left(-\int_{y_1(x)}^{y_2(x)} k(T_g(x, y)) dy\right). \end{aligned} \quad (40)$$

The calculation of  $I_R$  provided  $k$  and  $T_g$  are known (or given) is a direct problem which is usually considered while modelling the process. However, the inverse problem is more important from practical point of view and unfortunately more complex. The inverse problem is to find  $k(T_g(x, y))$  from measured  $I_R(x)$ . In this case equation (40) is considered as an integral equation with respect to local function  $k(T_g(x, y))$  with

measured integral function  $I_R(x)$ . Note that differential equation (34) does not allow to find local properties from integral properties (which are measured by the detectors). Equation (40) only allows to do that.

### 3.3.6. A Method to Solve Integral Equation (40)

Integral Equation (40) is very complex from the point of view of finding function  $k$  (as well as  $T_g$ ) because it occurs four times within the integrals. There are almost no publications about the solution of equations like equation (40).

If the right-hand side of equation (40) had only first term then that would be Radon's integral equation known in x-ray computer tomography [86, 87, 88]. Taking into account axial symmetry that equation would be Abel's integral equation [89]. At the same time the methods for solving Radon's and Abel's equations are well studied. Therefore we will consider the iteration method using only the first term of the right-hand side of equation (40) on the 0<sup>th</sup> iteration whereas other terms we will take into account on the following iteration in order to correct the solution. In other words, let us consider the case when  $T_0 \gg T_g$  (the temperature of the blackbody radiation source is considerably higher than that of the flame), i.e. absorption in the gas is stronger than its emission.

#### The Initial Approximation

Let us take into account the axial symmetry of the flame and replace the variables according to  $r = \sqrt{x^2 + y^2}$  (see fig. 34). Let us change the first term in the right-hand side of (40)

$$\begin{aligned} \exp\left(-\int_{y_1(x)}^{y_2(x)} k(T_g(x, y)) dy\right) &= \exp\left(-2\int_0^{y_2(x)} k(T_g(x, y)) dy\right) = \\ &= \exp\left(-2\int_x^R \frac{r}{\sqrt{r^2 - x^2}} k(T_g(r)) dr\right). \end{aligned} \quad (41)$$

Let us write equation (40) as follows taking into account only the first term in its right-hand side

$$\exp\left(-2\int_x^R \frac{r}{\sqrt{r^2 - x^2}} k_0(T_{g_0}(r)) dr\right) = \frac{I_R(x)}{I_s(T_0)} \quad (42)$$

or

$$2\int_x^R \frac{r}{\sqrt{r^2 - x^2}} k_0(T_{g_0}(r)) dr = q_0(x) \quad (43)$$

where

$$q_0(x) = -\ln \frac{I_R(x)}{I_s(T_0)}. \quad (44)$$

Equation (43) is well known Abel's singular integral equation with respect to  $k_0$  with known  $q_0$  from measurements. It is reasonable to solve it numerically using the generalized quadrature method [89]. Once equation (43) is solved  $T_{g_0}$  can be found

using the spectroscopic databases [76, 77]. Thus we can get the initial (0<sup>th</sup>) approximations for  $k$  and  $T_g$ .

Åkesson et al. [90] have considered axially symmetrical flame and Abel's integral equation which has been solved using the Tikhonov regularization method. However, the generalized quadrature method is more effective when applied to Abel's integral equation due to Abel's singular integral equation possesses a self-regularization effect [89].

#### The Following Approximations

Let us consider the  $i$ -th approximation ( $i=1,2,3,\dots$ ). Let us write (40) as follows

$$2 \int_x^R \frac{r}{\sqrt{r^2 - x^2}} k_i(r) dr = q_i(x) \quad (45)$$

where

$$q_i(x) = -\ln \left\{ \frac{I_R(x)}{I_s(T_0)} - \int_{y_1(x)}^{y_2(x)} k_{i-1}(T_{g_{i-1}}(x, y)) \frac{I_s(T_{g_{i-1}}(x, y))}{I_s(T_0)} \exp \left( \int_{y_1(x)}^y k_{i-1}(T_{g_{i-1}}(x, y')) dy' \right) dy \times \right. \\ \left. \times \exp \left( - \int_{y_1(x)}^{y_2(x)} k_{i-1}(T_{g_{i-1}}(x, y)) dy \right) \right\}. \quad (46)$$

The integrals in (46) are simple and can be solved numerically. However,  $k(r)$  is a solution to equation (45) but for calculating  $q(x)$  according to (46)  $k(x, y)$  is necessary. Therefore, e.g., interpolation is needed. In order to avoid this "inconsistence" an expression for  $q_i(x)$  can be written in a different way

$$q_i(x) = -\ln \left\{ \frac{I_R(x)}{I_s(T_0)} - \int_x^R \frac{r}{\sqrt{r^2 - x^2}} k_{i-1}(r) \frac{I_s(T_{g_{i-1}}(r))}{I_s(T_0)} \left[ \exp \left( \int_x^r \frac{r'}{\sqrt{r'^2 - x^2}} k_{i-1}(r') dr' \right) + \right. \right. \\ \left. \left. + \exp \left( - \int_x^r \frac{r'}{\sqrt{r'^2 - x^2}} k_{i-1}(r') dr' \right) \right] dr \cdot \exp \left( - \int_x^R \frac{r}{\sqrt{r^2 - x^2}} k_{i-1}(r) dr \right) \right\}. \quad (47)$$

The integrals in (47) are more complex (they are singular) than in (46) but they contain  $k(r)$  (as (45) also does) and interpolation is not needed. Furthermore, Sizikov et al. [89] have developed an effective method (the generalized quadrature method) for numerical solution of integrals of this type. Final conclusions about advantages and disadvantages of equations (46) and (47) can be made based on computer modeling results.

In each  $i$ -th iteration, the absorption coefficient  $k_i(r) \equiv k_i(T_{g_i}(r))$ ,  $i=0,1,2,\dots$  can be found by solving Abel's equation (43) and equation (45). The  $i$ -th approximation  $T_{g_i}(r)$  to the temperature profile for some  $z$ -cross-section of the flame can be found using the spectroscopic databases [76, 77].

The question about the convergence of the iterations is important. Since the problem of solving integral equation (40) is ill-posed [87, 88, 91, 92] the iterations, as fol-

lows from experience [92], converge to the exact solution only in the beginning ( $i \leq 10$ ) and after that they move away from the exact solution and eventually diverge. So usually 3 to 5 iterations are made or the iterations are stopped when [92]

$$\frac{\|k_i(r) - k_{i-1}(r)\|}{\|k_i(r)\|} \leq \varepsilon \quad (48)$$

where, for instance,  $\varepsilon = 10^{-3}$ .

Now let us consider the case when  $T_g \gg T_0$  (the flame temperature is considerably higher than that of the blackbody radiation source). In other words, the emission from the flame is stronger than absorption by the flame (this case is given on fig. 33). In this case differential equation (34) can be approximated by

$$dI(x, y) = k(T_g) I_s(T_g) dy, \quad (49)$$

where  $T_g = T_g(x, y) = T_g(r)$ ,  $I_s(T_g)$  is given by (35). The solution of equation (49) is (cf. (39))

$$I(x, y) = I(y_1) + \int_{y_1(x)}^y k(T_g(x, y')) I_s(T_g(x, y')) dy'$$

where  $I(y_1) = I_s(T_0)$ . If  $T_g = \text{const}$  then

$$I(x, y) = I_s(T_0) + k(T_g) I_s(T_g) \cdot (y - y_1)$$

i. e. the intensity of the beam is a linear function of  $y$ .

The measured by the detector value of  $I$  divided by  $I_s(T_0)$  ( $T_g \neq \text{const}$ ) is given by (cf. (40))

$$\frac{I_R(x)}{I_s(T_0)} = 1 + \int_{y_1(x)}^{y_2(x)} k(T_g(x, y)) \frac{I_s(T_g(x, y))}{I_s(T_0)} dy \quad (50)$$

(this equation is the same as (40) with  $\int_{y_1(x)}^{y_2(x)} k(T_g(x, y)) dy = 0$ , i. e. in the absence of the absorption).

Equation (50) is an integral equation with respect to  $k(T_g(x, y))$ . Let us consider the problem of solving equation (50). We will take into account the axial symmetry of the flame and replace the variables according to  $r = \sqrt{x^2 + y^2}$

$$2 \int_x^R \frac{r}{\sqrt{r^2 - x^2}} p(r) dr = \frac{I_R(x)}{I_s(T_0)} - 1 \quad (51)$$

where  $p(r) = k(T_g(r)) I_s(T_g(r)) / I_s(T_0)$ . Equation (51) is Abel's singular integral equation with respect to  $p(r)$ . We will solve it using the generalized quadrature method (like we do with (43)). Once equation (51) is solved  $T_{g_0} = T_{g_0}(r)$  and

$k_0(T_{g_0}(r)) = p(r) \frac{I_s(T_0)}{I_s(T_{g_0}(r))}$  can be found using the spectroscopic databases [76, 77].

Thus we can get the initial (0<sup>th</sup>) approximations for  $T_g$  and  $k$ .

Let us consider further approximations. We will take into account the absorption by the gas in every  $i$ -th approximation ( $i = 1, 2, 3, \dots$ ). We will solve integral equation (45) where  $q_i(x)$  is expressed by (47). The approximations  $k_i(r)$  will not differ much from

$k_0(r)$  because in this case the emission is considerably stronger than the absorption. Hence the iterations are expected to converge (this fact will be checked by means of computer computations).

### 3.3.7. Sweeping Scanning

In this scheme, the sweeping scanning takes place and the axial symmetry of the flame is taken into account (fig. 35). The sources can be placed along the edge of the burner or on the circumference of radius  $2R$ . In both cases, correlations (31) – (35), (38) are valid for this scheme in the same as they are for scheme I. However, the following correlations will look in a different way.

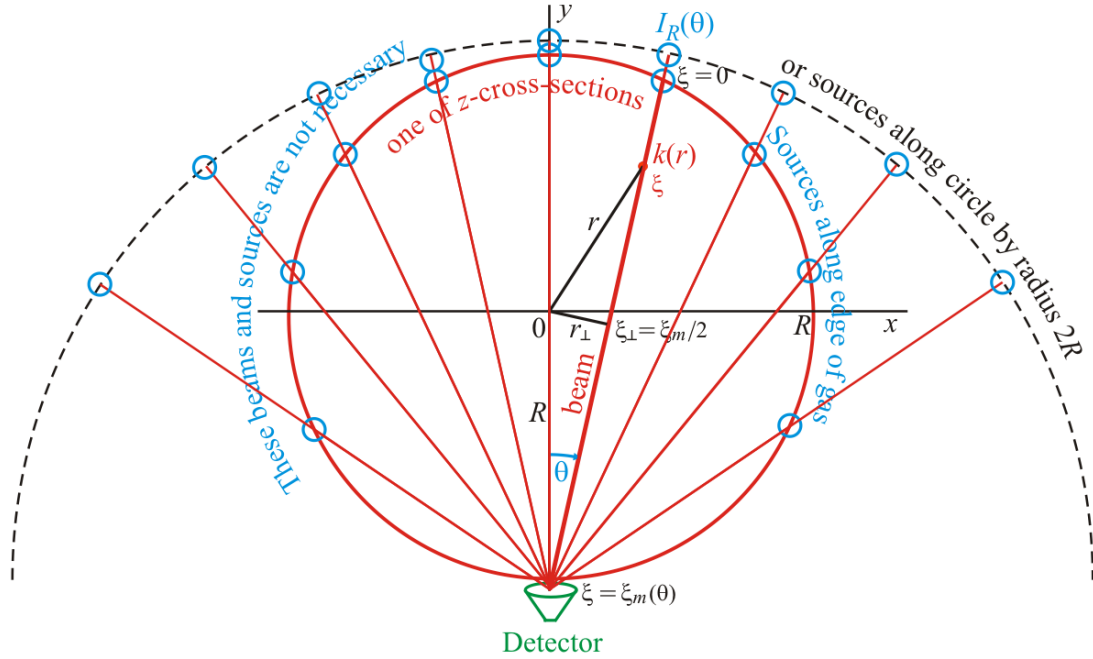
The intensity on the detector will be given by (cf. (40)]

$$\begin{aligned} \frac{I_R(\theta)}{I_s(T_0)} = & \exp\left(-\int_0^{\xi_m(\theta)} k(T_g(\theta, \xi)) d\xi\right) + \\ & + \left\{ \int_0^{\xi_m(\theta)} k(T_g(\theta, \xi)) \frac{I_s(T_g(\theta, \xi))}{I_s(T_0)} \exp\left(\int_{y_1(x)}^{\xi} k(T_g(\theta, \xi')) d\xi'\right) d\xi \right\} \times \\ & \times \exp\left(-\int_0^{\xi_m(\theta)} k(T_g(\theta, \xi)) d\xi\right), \end{aligned} \quad (52)$$

where the coordinate  $\xi$  is directed along the beam,  $\theta$  is the angle of scanning (the angle between the beam and the  $y$ -axis),  $\xi_m = \xi_m(\theta) = 2R \cos\theta$  is the distance from the source to the detector (for the case when the sources are placed along the edge of the burner).

We will consider equation (52) as an integral equation with respect to the local function  $k(T_g(\theta, \xi))$  with the measured integral function  $I_R(\theta)$  (in the similar way as we did for equation (40)].





**Figure 35.** Sweeping Scanning Scheme of Measurements. This figure is explanatory. See Fig. 25 and Sec. 2.3.8 for the actual layout and values. The blackbody temperature during the measurements was  $T_0 = 894.4 \text{ }^\circ\text{C}$  (1167.6 K) (the blackbody was the source, Sec. 2.3.3).

### 3.3.8. A Method for Solving Integral Equation (52)

We will solve equation (52) in the same way as we did with equation (40) using the iteration method.

#### The Initial Approximation

In the initial approximation we will only take into account the first term in the right-hand side of (52)

$$\frac{I_R(\theta)}{I_s(T_0)} = \exp\left(-\int_0^{\xi_m(\theta)} k_0(T_{g_0}(\theta, \xi)) d\xi\right). \quad (53)$$

Let us take into account the geometry of scheme II and the axial symmetry of the flame.

We will express  $\xi$  in (53) in terms of  $r$ . According to the law of cosines  $r^2 = \xi^2 + R^2 - 2\xi R \cos\theta$  and thus  $\xi^2 - 2R \cos\theta \xi + R^2 - r^2 = 0$  which is a quadratic equation with respect to  $\xi$ . Its solution is

$$\xi_{1,2} = R \cos\theta \pm \sqrt{R^2 \cos^2\theta - R^2 + r^2} = R \cos\theta \pm \sqrt{r^2 - R^2 \sin^2\theta} \quad (54)$$

and thus (53) can be written as

$$q_0(\theta) = 2 \int_{r_{\perp}(\theta)}^R k_0(T_{g_0}(r)) \frac{r dr}{\sqrt{r^2 - R^2 \sin^2\theta}} \quad (55)$$

where

$$q_0(\theta) = -\ln \frac{I_R(\theta)}{I_s(T_0)}. \quad (56)$$

Note that  $r_{\perp}(\theta) = R \sin \theta$  and thus

$$2 \int_{R \sin \theta}^R \frac{r}{\sqrt{r^2 - R^2 \sin^2 \theta}} k_0(T_{g_0}(r)) dr = q_0(\theta), \quad 0 \leq \theta \leq \pi/2. \quad (57)$$

Integral equation (57) is a new equation in the field of tomography. It is reasonable to solve it in the same way as equation (43) using the generalized quadrature method [89]. Once equation (57) is solved  $T_{g_0}(r)$  can be found using the spectroscopic databases [76, 77]. Thus we can get the initial (0<sup>th</sup>) approximations for  $k(T_g(r))$  and  $T_g(r)$ .

### The Subsequent Approximations

Let us consider the  $i$ -th approximation ( $i = 1, 2, 3, \dots$ ). Equation (52) can be written as

$$2 \int_{R \sin \theta}^R \frac{r}{\sqrt{r^2 - R^2 \sin^2 \theta}} k_i(T_{g_i}(r)) dr = q_i(\theta), \quad 0 \leq \theta \leq \pi/2 \quad (58)$$

where (cf. (46))

$$\begin{aligned} q_i(\theta) = & -\ln \left\{ \frac{I_R(\theta)}{I_s(T_0)} - \right. \\ & \left. - \int_0^{\xi_m(\theta)} k_{i-1}(T_{g_{i-1}}(\theta, \xi)) \frac{I_s(T_{g_{i-1}}(\theta, \xi))}{I_s(T_0)} \exp \left( \int_0^{\xi} k_{i-1}(T_{g_{i-1}}(\theta, \xi')) d\xi' \right) d\xi \times \right. \\ & \left. \times \exp \left( - \int_0^{\xi_m(\theta)} k_{i-1}(T_{g_{i-1}}(\theta, \xi)) d\xi \right) \right\} \end{aligned} \quad (59)$$

or (cf. (47))

$$\begin{aligned} q_i(\theta) = & -\ln \left\{ \frac{I_R(\theta)}{I_s(T_0)} - \right. \\ & \left. - \int_{R \sin \theta}^R \frac{r}{\sqrt{r^2 - R^2 \sin^2 \theta}} k_{i-1}(r) \frac{I_s(T_{g_{i-1}}(r))}{I_s(T_0)} \left[ \exp \left( \int_{R \sin \theta}^r \frac{r'}{\sqrt{r'^2 - R^2 \sin^2 \theta}} k_{i-1}(r') dr' \right) + \right. \right. \\ & \left. \left. + \exp \left( - \int_{R \sin \theta}^r \frac{r'}{\sqrt{r'^2 - R^2 \sin^2 \theta}} k_{i-1}(r') dr' \right) \right] dr \cdot \exp \left( - \int_{R \sin \theta}^R \frac{r}{\sqrt{r^2 - R^2 \sin^2 \theta}} k_{i-1}(r) dr \right) \right\}. \end{aligned} \quad (60)$$

The calculation of  $q_i(\theta)$  according to (59) or (60) is analogous to the calculation of  $q_i(x)$  in scheme I according to (46) or (47). The behaviour of the iterations is also analogous to that for  $q_i(x)$ .

## 4. Results and Discussion

### 4.1. Tomographic Reconstruction of the Lab Flame Temperature Profile

#### 4.1.1. Introduction and Remarks

To fulfill the objectives of the PhD project (Sec. 1.3.1), the developed in this PhD project tomographic algorithms for gas temperature profile reconstruction (Sec. 3.3) was proved by means of measuring the temperature profile of the small scale laboratory burner (Sec. 2.2) using the FTIR spectrometer and the optical setup as described in Sec. 2.3 and by comparing the results to the temperature profile obtained in the work of Ref. [74].

As discussed in Sec. 2.1.9 the **multichannel IR spectrometer system** (the first prototype of the 2Dt IR tomography system, Sec. 2.1.2, Fig 1) developed in this PhD project is sensitive to variations in the temperature of the IR camera housing which take place even under lab conditions. The FTIR spectrometer provides more thermal stability and a higher spectral resolution (Sec. 2.3). It was decided to use the FTIR spectrometer for spectral measurements on the burner. The results of the spectral measurements are input data to the developed tomographic algorithms for gas temperature profile reconstruction (Sec. 3.3). It is important for testing of the developed algorithms to have the spectral measurements of highest possible quality since tomographic reconstruction algorithms are known to be very sensitive to uncertainties in the input data (Sec. 3.3).

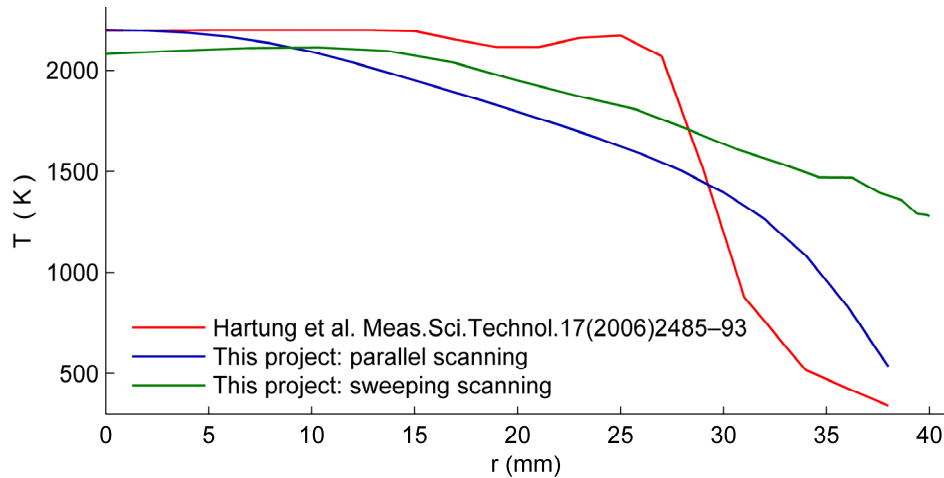
It should be noted that most of the components of the optical setup, in particular the IR optical fiber and the fiber optical adaptor (Sec. 2.3.3, Fig. 19), used for the spectral measurements discussed in this section were the same as those used in the **multichannel IR spectrometer system** (Sec. 2.1.2, Fig 1) when it was applied on the exhaust duct of the large Diesel engine (Sec. 2.4.2, Fig 26).

#### 4.1.2. Results for Stoichiometric Combustion ( $\phi = 1$ )

As described in Sec. 2.3 and Sec. 3.3, two schemes of measurements were implemented: parallel scanning (Sec. 2.3.6) and sweeping scanning (Sec. 2.3.8).

The results for the two schemes are presented in Fig. 36. The results are the burner temperature profiles as functions of the distance from the center of the burner plate,  $r$  [mm] ( $r = 30$  mm corresponds to the edge of the burner plate [74]). The burner has axial symmetry [74] hence the temperature profiles were expected to be axisymmetric and one coordinate, the distance from the center of the burner plate  $r$  [mm], is sufficient. The temperature profiles were obtained from the parallel scanning measurements (Table 3, Sec. 2.3.6) and the sweeping scanning measurements (Table 4, Sec. 2.3.8) and compared to the measurements on the same burner operated at the similar conditions conducted in the work of Ref. [74] (Sec. 2.2). The temperature profiles correspond to

the height above the burner plate of 12 mm ( $HAB = 12$  mm) (Sec. 2.2.4) and to the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) (Sec. 2.2.4).



**Figure 36.** Burner temperature profile as a function of the distance from the center of the burner plate,  $r$  [mm] ( $r = 30$  mm corresponds to the edge of the burner plate [74]), obtained from the parallel scanning measurements (blue) (Table 3, Sec. 2.3.6) and the sweeping scanning measurements (olive) (Table 4, Sec. 2.3.8) and compared to the measurements on the same burner operated at the similar conditions conducted in the work of Ref. [74] (red) (Sec. 2.2). The temperature profiles correspond to the height above the burner plate of 12 mm ( $HAB = 12$  mm) (Sec. 2.2.4) and to the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) (Sec. 2.2.4).

As can be seen, the temperature profiles obtained in this work are in agreement with the expected result that the temperature of the flame is at about its highest value at around the center of the burner plate and is decreasing towards the edges.

However the temperature profiles of this work significantly underestimate the temperature profile of Ref. [74] at  $r <$  about 30 mm ( $r = 30$  mm corresponds to the edge of the burner plate [74]) and overestimate it at  $r >$  about 30 mm. The deviation from the results of Ref. [74] is greater than the standard deviation of the temperature points of Ref. [74] which was 75 K (Sec. 2.2.6). However, the temperature profile obtained from the parallel scanning measurements is in very good agreement with the one of Ref. [74] at the center of the burner plate.

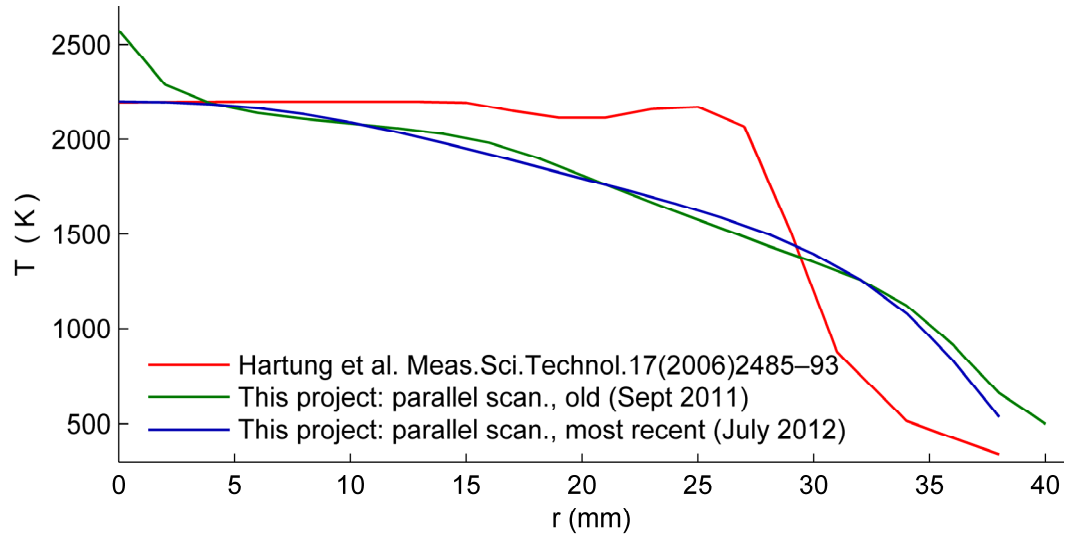
Also, the temperature profiles of this work do not have a sharp gradient at about  $25 \text{ mm} < r < 35 \text{ mm}$  that can be seen in the profile of Ref. [74].

The underestimation at  $r <$  about 30 mm can be due to the flow rates of methane and air in this work were about two times lower than those in the work of Ref. [74] at  $\phi = 1$  (Sec. 2.2.4). It follows, however, from Ref. [74] that the temperature profile is independent of the actual values of the flow rates (see also Sec. 2.2.4) but that it depends only on the height above the burner plate and the equivalence ratio  $\phi$  which were the same in this work as in work [74] (Sec. 2.2.4). Also, higher flow rates would, probably, produce a flame having a temperature profile with a sharper gradient at about  $25 \text{ mm} < r < 35 \text{ mm}$ . Hence that can also partially explain the small gradient in the profiles of this work at about  $25 \text{ mm} < r < 35 \text{ mm}$ . However, the fact that the flow rates were two times lower in this work does not explain the overestimation of the profile from Ref. [74] by the profiles obtained in this project at  $r >$  about 30 mm.

The fluctuations of the flame discussed in Sec. 2.2.5 (Fig. 16) having the amplitude of about 5% from the mean value and the frequency of about 10 Hz should also be taken into account when discussing the deviations in the results although in the parallel scanning scheme (Sec. 2.3.6) the FTIR spectral measurements on the flame were performed at the rate of 57 spectra per minute (which is almost 1 Hz) and each FTIR spectrum was the average of 50 spectra. It follows roughly from the simple consideration of the Nyquist frequency that the frequency at which the spectra are taken should be at least 20 Hz. Hence the flame fluctuations may cause significant deviations even between the results obtained during different sessions within this PhD project. This fact can partially explain the deviations between the temperature profiles obtained from the parallel and sweeping scanning measurements performed in this project.

A well known fact in the field of tomographic reconstruction algorithms (see references throughout Sec. 3.3) is that they are very sensitive to even minor deviations in the input data. This fact should also be taken into account since the profiles in this project were obtained from the line-of-sight FTIR spectral measurements with subsequent application of the developed tomographic reconstruction algorithm (Sec. 3.3) whereas the profiles of Ref. [74] were obtained from the point measurements. It should be stressed that the application on the industrial scale of the techniques employed in the work of Ref. [74], which are laser-based techniques (Sec. 2.2.6), could probably be more difficult and expensive since they are known to be critical to the alignment and require extensive optical access. On the contrary, the method of sweeping scanning (Sec. 2.3.8) developed in this project, is expected to be promising for applications on industrial scale since it was designed with a view to a limited optical access.

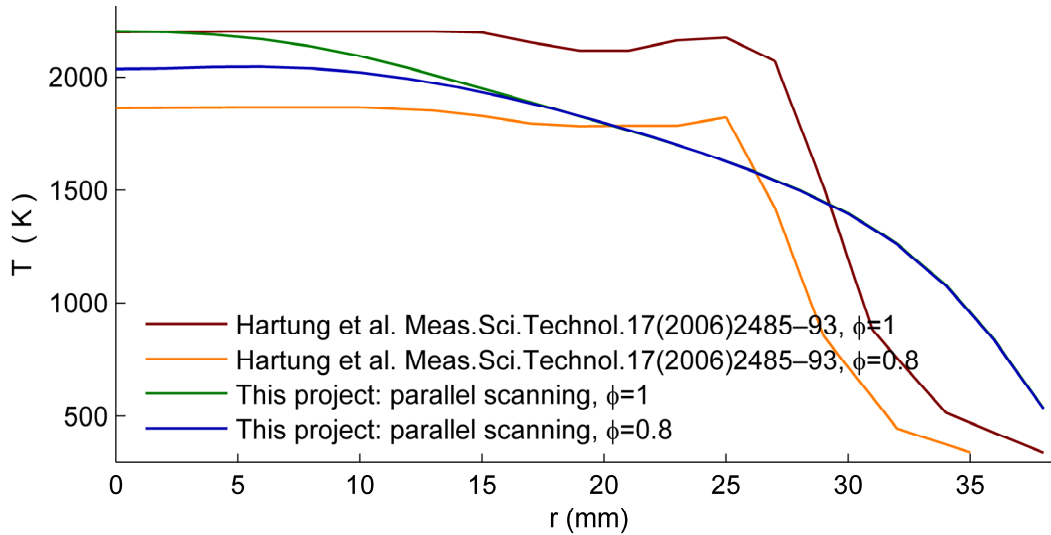
Despite the flame fluctuations and the sensitivity of tomographic algorithms to minor deviations in the input data, some remarkable consistency between the temperature profiles obtained in the different measurement sessions using similar schemes and at similar conditions can be observed in Fig. 37. The temperature profiles obtained from the parallel scanning measurements performed in September 2011 and those performed recently in July 2012 (the layout, parameters and conditions for which are given in Sec. 2.3.6, Table 3) are in good agreement at  $r > 4$  mm despite all the discussed above reasons for possible deviations. The strong deviation of the temperature profile obtained in the older session (September 2011) at  $r < 3$  mm is due to the line-of-sights in that session were covering only the half of the burner and the misalignment error was amplified by the tomographic reconstruction algorithm. A similar deviation at  $r < 3$  mm was also observed in the temperature profile obtained in the recent session (July 2012) if the line-of-sights covering only a half of the burner were used, i.e. without averaging the data for the respective line-of-sights as described in Sec. 2.3.6, Table 3.



**Figure 37.** Burner temperature profile as a function of the distance from the center of the burner plate,  $r$  [mm] ( $r = 30$  mm corresponds to the edge of the burner plate [74]), obtained from the parallel scanning measurements performed in September 2011 (olive) and those performed recently in July 2012 (blue) (the layout, parameters and conditions for which are given in Sec. 2.3.6, Table 3) and compared to the measurements conducted in the work of Ref. [74] (red) (Sec. 2.2.6). The temperature profiles correspond to the height above the burner plate of 12 mm ( $HAB = 12$  mm) (Sec. 2.2.4) and to the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) (Sec. 2.2.4).

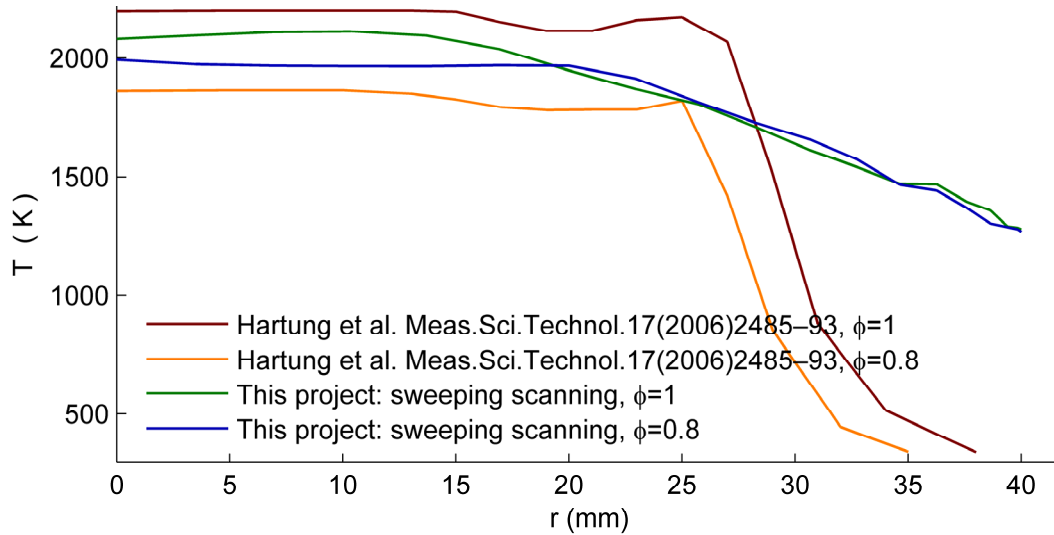
#### 4.1.3. Results for Lean Combustion ( $\phi = 0.8$ )

The temperature profiles obtained from the parallel scanning measurements (Sec. 2.3.6) for two different equivalence ratios  $\phi = 1$  and  $\phi = 0.8$  (Sec. 2.2.4) are compared in Fig. 38.



**Figure 38.** Burner temperature profile as a function of the distance from the center of the burner plate,  $r$  [mm] ( $r = 30$  mm corresponds to the edge of the burner plate [74]), corresponding to the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) obtained from the parallel scanning measurements (olive) (Sec. 2.3.6, Table 3) and to the lean combustion ( $\phi = 0.8$ ) also obtained from the parallel scanning measurements (blue) (Sec. 2.3.6) and compared to the measurements conducted in the work of Ref. [74] for  $\phi = 1$  (red) and  $\phi = 0.8$  (orange) (Sec. 2.2.4, Sec. 2.2.6). The temperature profiles correspond to the height above the burner plate of 12 mm (HAB = 12 mm) (Sec. 2.2.4).

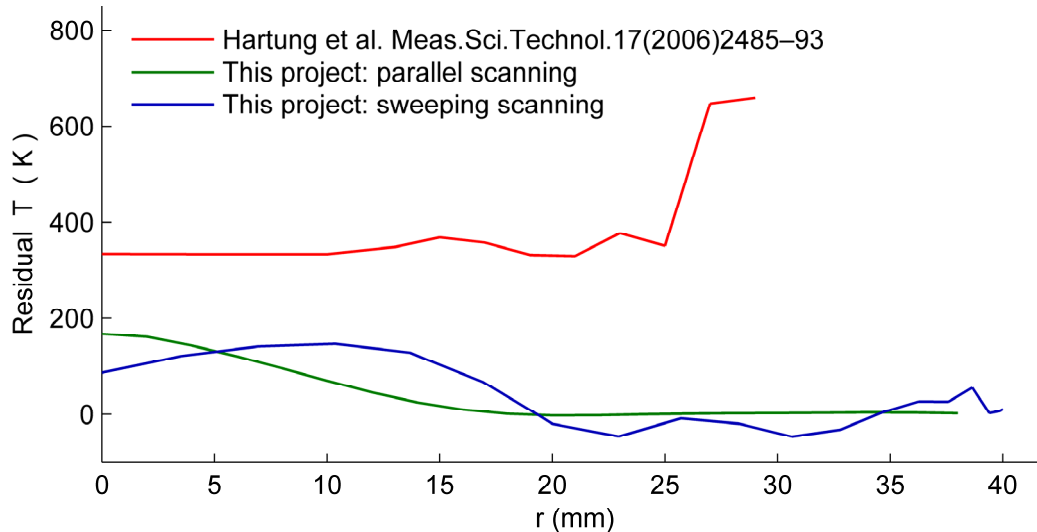
The temperature profiles obtained from the sweeping scanning measurements (Sec. 2.3.8) also for  $\phi = 1$  and  $\phi = 0.8$  (Sec. 2.2.4) are compared in Fig. 39. It is interesting to note that a good agreement between the temperature profiles for  $\phi = 1$  and  $\phi = 0.8$  at  $r > 15$  mm (for parallel scanning curves) and at  $r > 20$  mm (for sweeping scanning curves) can be observed for the two independent measurements with the two different schemes: the parallel scanning and the sweeping scanning.



**Figure 39.** Burner temperature profile as a function of the distance from the center of the burner plate,  $r$  [mm] ( $r = 30$  mm corresponds to the edge of the burner plate [74]), corresponding to the stoichiometric combustion (equivalence ratio  $\phi = 1$ ) obtained from the sweeping scanning measurements (olive) (Sec. 2.3.8, Table 4) and to the lean combustion ( $\phi = 0.8$ ) also obtained from the sweeping scanning measurements (blue) (Sec. 2.3.8) and compared to the measurements conducted in the work of Ref. [74] for  $\phi = 1$  (red) and  $\phi = 0.8$  (orange) (Sec. 2.2.4, Sec. 2.2.6). The temperature profiles correspond to the height above the burner plate of 12 mm (HAB = 12 mm) (Sec. 2.2.4).

Moreover the temperature difference between the profiles for  $\phi = 1$  and  $\phi = 0.8$  is about 70-167 K at  $r < 10$  mm for both measurements performed in this PhD project, whereas the difference between the temperature profiles for  $\phi = 1$  and  $\phi = 0.8$  from Ref. [74] is about 335 K. These observations can more clearly be seen in Fig. 40 where the residuals between the respective temperature profiles for  $\phi = 1$  and  $\phi = 0.8$  are shown.





**Figure 40.** The residuals between the temperature profiles for  $\varphi = 1$  and  $\varphi = 0.8$  from Fig. 38 and Fig. 39 obtained from the parallel scanning measurements (olive) (Sec. 2.3.6) (i.e. the olive curve is the profile for  $\varphi = 1$  minus the profile for  $\varphi = 0.8$  both obtained from the parallel scanning measurements in this PhD project (Sec. 2.3.6) and shown in Fig. 38), the sweeping scanning measurements (blue) (Sec. 2.3.8) and those obtained in the work of Ref. [74] (red).

Also, as can be seen from both Fig. 38 and Fig. 39 the temperature profiles obtained in this PhD project for lean combustion ( $\varphi = 0.8$ ) from both parallel and sweeping scanning measurements (both sessions) overestimate the temperature profile from Ref. [74] at almost all positions  $r$ . It should be noted that the flow rates at  $\varphi = 0.8$  for both methane and air during the measurements in this PhD project were not two times higher than those in the work of Ref. [74] (like in case of  $\varphi = 1$  discussed above, Sec. 4.1.2, Sec. 2.2.4) but were almost the same (more precisely, about 1.1 times higher than those of [74], Sec. 2.2.4).

#### 4.1.4. Discussion on the Deviation from the Results of Ref. [74]

The above observations especially as regards the agreement between the residuals between the temperature profiles (Fig. 40) obtained in the independent sessions in this PhD project where different schemes of measurements and calculations were applied allow to conclude that the deviation of the temperature profiles of this work from those of work [74] can be due to the operation of the burner was probably different here with respect to that in work [74].

Another consideration is that the calculations of the temperature profiles in this work were performed from the experimental spectral values at the wave number of  $2270.05 \text{ cm}^{-1}$  (Sec. 2.3.6, Sec. 2.3.8) which belongs to the  $4.3 \text{ }\mu\text{m}$  ( $2350 \text{ cm}^{-1}$ ) band of  $\text{CO}_2$ . As mentioned in Sec. 2.3.6, that wave number was chosen from the following considerations: the cold  $\text{CO}_2$  (from the ambient air) has negligible absorption at that wave number but at the same time the absorption from hot  $\text{CO}_2$  (from the flame) is sufficiently strong. Also, as can be seen from Fig. 24 (Sec. 2.3.6), the emission from the  $\text{CO}_2$  (in the flame) at that wave number is very strong. Anyway, the presence of cold

CO<sub>2</sub> from the ambient air in the field of view of the optical setup has probably significant effect on the results of the temperature profile calculations taking into account the fact that the tomographic algorithm was used to obtain the profiles whereas tomographic algorithms in general are known to possess an intrinsic property of being sensitive to minor uncertainties in the experimental data.

The wide field of view of the optical setup ( $\varnothing$  6 mm, Sec. 2.3.3) compared to the size of the burner ( $\varnothing$  60 mm) causes overestimation of the gas temperature at  $r > 30$  mm (Fig. 36, Fig. 37, Fig. 38, Fig. 39) due to “catching” the hot CO<sub>2</sub> emission from the neighboring flame layers.

Also, some oscillations in flame emission were observed in this work (Sec. 2.2.5, Fig. 16). The oscillations have the amplitude of about 5% from the mean value and the frequency of about 10 Hz and may influence to a certain degree the results of the measurements on the burner.

As well-known, tomographic reconstruction techniques require as many line-of-sights as possible and the field of view of each line-of-sight should be as narrow as possible. This requirement is often satisfied in medical tomographic systems where the number of line-of-sights reaches 100-1000 and the reconstruction can be considered to take place on a semi-continuum line-of-sight scale. However in the tomography of gases, flames or plasmas this number is usually limited to significantly lower values, typically 5-20, and therefore the reconstruction takes place on a discrete scale. In this case a tomographic reconstruction algorithm gives not exactly expected temperature profile, on the contrary, it gives only an approximation to it. The quality of approximation can be improved significantly by taking into account all possible a priori information about the system (i.e. the burner) like, e.g., highest/lowest temperature limits, temperature fluctuations or noise in the experimental data.

It appears that the 6 mm field of view of the optical setup (Sec. 2.3.3) is not narrow enough. Narrowing the field of view would require a more sophisticated experimental setup. Also, as mentioned above, only one wave number in the CO<sub>2</sub> (hot) band was used in the algorithm. The use of a broader spectral range covering the “hot” and “cold” parts in the CO<sub>2</sub> band would be beneficial. The use of an OH band in the UV range could also be an advantage because that would give the flame temperature and not the temperature of the cold/hot CO<sub>2</sub>. These challenges have to be considered for the future developments in the project. On the other hand, considering the large scale (e.g., a power plant boiler) the 6 mm field of view is quite small and the number of line-of-sights can be very large (typically, the dimensions of power plant boilers are of the order  $10 \times 10$  m). Therefore the developed IR tomography system and methodology is expected to be feasible and promising for application on the large industrial combustion systems rather than on the small lab-scale burners.

## 4.2. Simultaneous Fast Exhaust Gas Temperature Measurements on the Large Diesel Engine

### 4.2.1. Introduction and Remarks

The **multichannel IR spectrometer system** developed in this PhD project (Sec. 2.1) was applied for simultaneous fast time-resolved transient IR spectral measurements in the three optical ports of the exhaust duct of the large Diesel engine (Sec. 2.4) as stated in the objectives of the PhD project (Sec. 1.3.2).

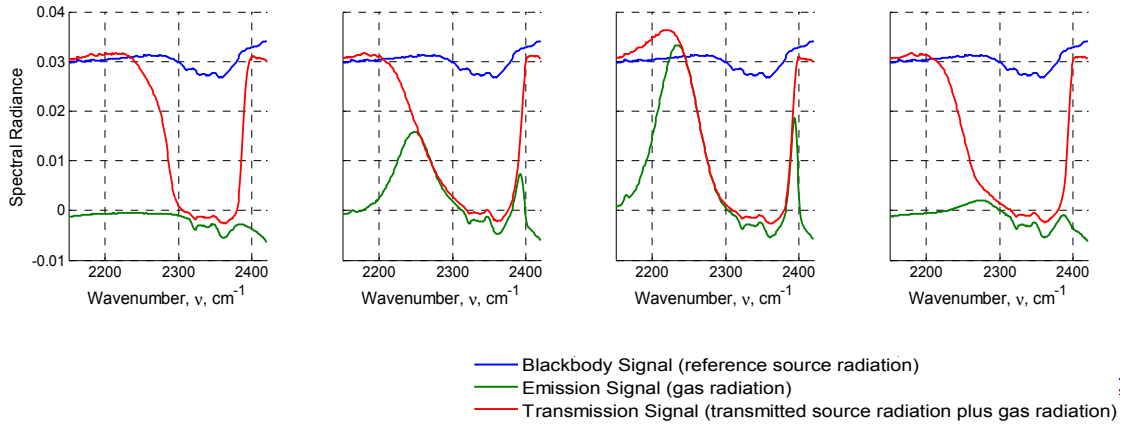
The measurements on the large Diesel engine were also relevant to the EU-funded HERCULES-B project “High-efficiency engine with ultra-low emissions for ships” carried out under 7<sup>th</sup> Framework Program “Sustainable Surface Transport” (grant agreement SCP7-GA-217878) presented in reports [9].

It should be noted that the simultaneous fast time-resolved transient IR spectral measurements in the three optical ports of the exhaust duct of the large Diesel engine were performed for the first time. The gas temperature was obtained simultaneously at the three ports from the simultaneous IR measurements as the brightness temperature of the intensity of the 4.3  $\mu\text{m}$   $\text{CO}_2$  band (Sec. 3.2).

The gas temperature must be known to measure, e.g., NO concentration by optical spectroscopy as well as to measure other pollutant species concentrations or other quantities characterizing combustion phenomena. The gas temperature in the three ports of the exhaust duct was also obtained from the structure of the UV NO 226 nm absorption band using a different system [9, 10]. A reasonable agreement between the IR  $\text{CO}_2$  measurements of the gas temperature (performed using the **multichannel IR spectrometer system**) and UV NO measurements was observed [9, 10] (Sec. 4.2.3).

### 4.2.2. Emission and Transmission IR Spectra of the Exhaust Gas

Examples of the emission and transmission IR spectra of the exhaust gas at the middle port of the exhaust duct of the large Diesel engine (Fig. 27 and Fig. 28, Sec. 2.4.3) at four different time points (within one stroke) obtained using the **multichannel IR spectrometer system** (Sec. 2.1) are shown in Fig. 41. It is possible to calculate the  $\text{CO}_2$  brightness temperature from those emission and transmission IR spectra as described in Sec. 3.2.



**Figure 41.** Examples of the emission (olive) and transmission (red) IR spectra of the exhaust gas at the middle port of the exhaust duct of the large Diesel engine (Fig. 27 and Fig. 28, Sec. 2.4.3) at four different time points within one stroke (see also Fig. 31, Sec. 3.2) and the spectrum of the reference source radiation (blue).

The range from about 2300 to 2400  $\text{cm}^{-1}$  corresponds to the strong absorption by  $\text{CO}_2$  [17, 69, 75]. For this reason both the emission and transmission signals have a sharp dip in that range and a consequent low signal-to-noise ratio leads to negative values in that region. The negative values can also be partially attributed to the reflections in the optical setup of the system and to thermal instability of the InSb array in the IR camera of the **multichannel IR spectrometer system** which is sensitive to even minor variations in the ambient temperature (Sec. 2.1.9). The latter can be improved by placing the system into the temperature chamber. The effect from the reflections can be reduced by covering the optics surfaces with a special antireflection coating.

It can also be seen in Fig. 41 that the transmission signal is higher than the reference signal in the range from about 2150 to 2250  $\text{cm}^{-1}$ . The absorption by  $\text{CO}_2$  is negligible in that range and it means that the transmission and reference signals should be equal. The higher values of the transmission signal are due to misalignments in the experiment on the transmission measurements compared to the one on the reference measurements. The reference source was held in hands in front of the corresponding optical access ports of the exhaust duct (Sec. 2.4.3) during both experiments. As mentioned in Sec. 2.4.3, that can be improved by making a special setup with three blackbodies mounted together such that all three blackbodies would be fixed in their positions in front of the optical ports.

#### 4.2.3. Exhaust Gas Temperature in the Three Optical Ports of the Exhaust Duct

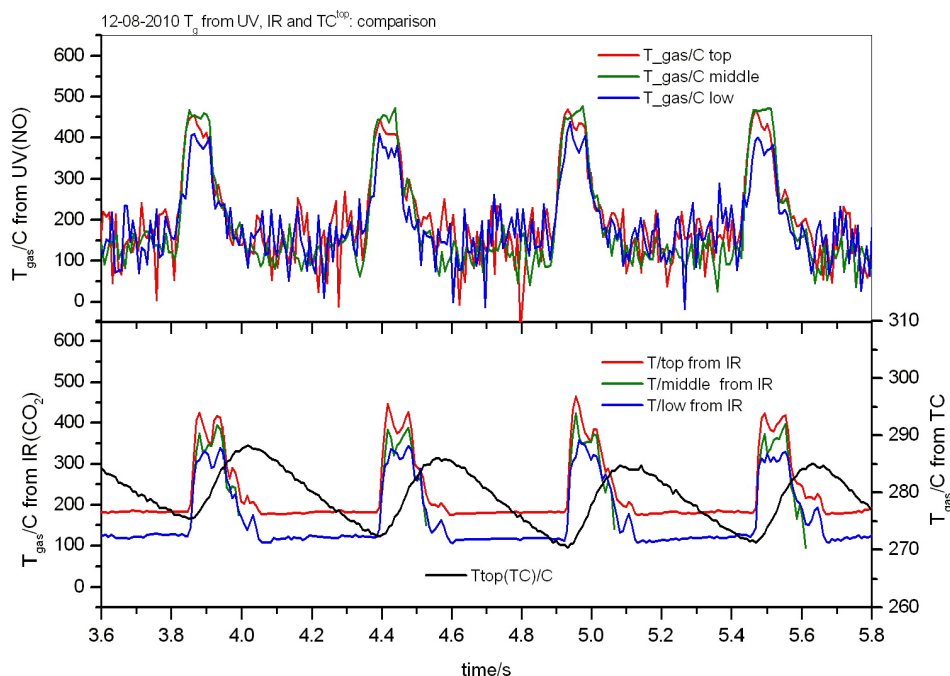
The exhaust gas temperature was obtained simultaneously at the three optical ports from the simultaneous IR measurements conducted using the **multichannel IR spectrometer system** (Sec. 2.1) as the brightness temperature of the intensity of the 4.3  $\mu\text{m}$   $\text{CO}_2$  band (Sec. 3.2).

The spectral range 2270-2290  $\text{cm}^{-1}$  was used for the gas temperature calculation as described in Sec. 3.2 since in that region the emission and transmission signals (Fig. 41,

Fig. 31) have maximum signal-to-noise ratio, the absorption by cold  $\text{CO}_2$  present in the optical path of the system is weak and the spectral absorptance is close to unity in that region (Fig. 32, Sec. 3.2.3).

In the range  $2270\text{--}2290\text{ cm}^{-1}$  the assumption of spectral absorptance being equal to 0.9 was found to be a reasonable approximation based on the analysis of the emission and transmission spectra (Sec. 3.2.3). The exhaust gas temperature was then calculated as the brightness temperature from the simultaneous emission spectra using the spectral emission-absorption method (Sec. 3.2.2, Sec. 3.2.4).

The exhaust gas temperature in the three optical ports of the exhaust duct was also obtained from the structure of the UV NO 226 nm absorption band using a different system [9, 10]. A reasonable agreement between the IR  $\text{CO}_2$  measurements of the exhaust gas temperature (performed using the **multichannel IR spectrometer system**) and the UV NO measurements was observed [9, 10] as can be seen in Fig. 42 taken from Ref. [10].



**Figure 42.** (The figure is taken from Ref. [10].)  
 Upper panel. Exhaust gas temperature as a function of time at the top (red), middle (olive) and bottom (blue) optical ports of the exhaust duct calculated from the NO UV absorption band at 226 nm [9, 10].  
 Lower panel. Exhaust gas temperature as a function of time at the top (red), middle (olive) and bottom (blue) optical ports calculated from the  $\text{CO}_2$  IR absorption band at  $4.3\ \mu\text{m}$ . The exhaust gas temperature measured by the thermocouple (TC) (which was mounted inside the exhaust duct close to the top optical port) is shown by the black curve.

The temperature behavior shows periodicity in accordance with the stroke period of the engine (which was 0.54 s, corresponding to a frequency of 1.9 Hz). As for the temperature curves obtained from the IR  $\text{CO}_2$  measurements, the number of temperature points per stroke was 64 and that was defined by the IR camera frame rate which in the

experiment was 119 Hz (the camera was synchronized with the encoder of the main shaft of the engine).

It should be noted that the UV NO and IR CO<sub>2</sub> measurements were performed at the same engine running settings but at different time points [9, 10]. It can be seen that the UV NO and IR CO<sub>2</sub> measurements agree within 9% at the time points of the exhaust valve openings (corresponding to the hills on the temperature curves) that can be considered as good agreement since the relative error of the UV NO measurements was estimated to  $\pm 10\%$  [9, 10]. The UV NO and IR CO<sub>2</sub> measurements both give lower gas temperature at the bottom optical port compared to that at the middle and top optical ports when the exhaust valve is opened. This result fits with the expected convective paths of the hot exhaust gas in the exhaust duct.

As discussed in Sec. 2.1.9, the relative temperature measurement error of the IR CO<sub>2</sub> measurements (conducted using the **multichannel IR spectrometer system**) was estimated to  $\pm 3\%$ .

The results corresponding to the time points between the subsequent exhaust valve openings (the depressions on the temperature curves) were strongly affected by noise due to very low signal level and are therefore disregarded in the discussion. The UV NO curves exhibit noisy behavior at those time points whereas the IR CO<sub>2</sub> curves for the top and bottom optical ports exhibit an offset with respect to each other and the IR CO<sub>2</sub> curve for the middle port was contaminated by noise to such an extent that it is not shown at those time points. It should be noted that the offset in the IR CO<sub>2</sub> curves at the time points between the valve openings cannot be used for correction of the values at the time points of the valve openings since the offsets are just due to the low signal-to-noise ratio at the time points between the valve openings which is not the case at the time points of the valve openings.

It can be seen in Fig. 42 that the gas temperature varies quickly upon the exhaust valve opening. When the valve opens a fast expanding and cooling down of the exhaust gas takes place. The heat is then transferred to the thermocouple (TC) (which was mounted inside the exhaust duct close to the top optical port) that shows an increase in its own temperature (Fig. 42, lower panel, black curve).

As mentioned above, it should be noted that in the range 2270-2290 cm<sup>-1</sup> (used in the exhaust gas temperature calculation from the IR CO<sub>2</sub> measurements, Sec. 3.2) the absorption by cold CO<sub>2</sub> is weak. Cold CO<sub>2</sub> is present, e.g., near the windows of the optical ports of the exhaust duct and in the optical path of the **multichannel IR spectrometer system**. The path should be flushed with an IR-inactive gas (e.g., N<sub>2</sub>) in order to remove cold CO<sub>2</sub> (as well as H<sub>2</sub>O) from the path for best results. The absorption by cold CO<sub>2</sub> leads to a decrease in the calculated temperature values. This can be an explanation of the fact that the temperatures obtained from the UV NO measurements are on average higher by about 50 °C than those obtained from the IR CO<sub>2</sub> measurements (Fig. 42).

## 5. Summary

### 5.1. Tomographic Reconstruction of the Lab Flame Temperature Profile

The lab-scale gas burner producing a flame with known temperature profiles was bought and the experimental setup (the burner, the gas flow system and the optical setup) was established (Sec. 2.2, Sec. 2.3) in order to validate the developed tomographic algorithms for gas temperature profile reconstruction (Sec. 3.3).

The burner is a flat flame burner (Sec. 2.2.2, Fig. 10) producing a laminar pre-mixed methane ( $\text{CH}_4$ )/air flame with known axisymmetric temperature profiles. The choice of the burner was based on Ref. [74] where the temperature profiles at different methane/air ratios and heights above the burner plate are presented. In the work of Ref. [74] the temperature profiles were measured by the CARS and Na-line reversal techniques and compared with the modeling results. The burner is reported to be “a well-characterized high temperature source which other researchers in the field of laser-based thermometry can simply reconstruct and apply for their calibration and validation purposes without having to characterize the high temperature source themselves” [74]. The burner was made in Germany under the supervision of G. Hartung (the first author of Ref. [74]).

Some oscillations in the burner flame emission were observed in this work (Sec. 2.2.5, Fig. 16). The oscillations have the amplitude of about 5% from the mean value and the frequency of about 10 Hz and may influence to a certain degree the results of the measurements on the burner. The oscillations are unlikely to be due to some instabilities or eigenfrequencies in the flow controllers or other elements of the gas supply system but are caused by the flame propagation dynamics, buoyancy effects and convection.

The tomographic algorithms for gas temperature profile reconstruction (Sec. 3.3) were developed by an external collaborator Prof. Valery Sizikov from National Research University of Information Technologies, Mechanics and Optics (St Petersburg, Russia) who is a well-recognized expert in solution of inverse ill-posed problems (see, e.g., Ref. [88]: Petrov Yu. P., Sizikov V. S. *Well-Posed, Ill-Posed, and Intermediate Problems with Applications*. VSP, Leiden-Boston, 2005, 234 pp.).

The algorithms were developed for the simplest case of an axisymmetric temperature field of a flame/hot flue gas as a first step in the development (Sec. 3.3). Also, as mentioned above, the laboratory burner gives the flame having the axisymmetric temperature profile [74] (Sec. 2.2).

The algorithms were designed for the two schemes of measurements: the parallel scanning scheme (Sec. 2.3.6) and the sweeping scanning scheme (Sec. 2.3.8).

The parallel scanning scheme provides the most efficient distribution of the line-of-sights and hence satisfies the condition of minimal reconstruction error. The tomographic reconstruction algorithms possess an essential property of being sensitive to minor

uncertainties in the input data. Hence parallel scanning measurements provide a means to test the algorithms at their best. However, the application of the parallel scanning scheme on industrial scale might be difficult due to the requirement of many optical access ports to the measurement area (e.g. in a boiler of a power plant).

The method of sweeping scanning developed in this PhD project, is expected, on the contrary, to be promising for applications on industrial scale since it was designed with a view to a limited optical access. The sweeping scanning scheme allows using a low number of access ports. By means of rotating the detectors installed in the access ports, the “sweeping” of the measurement area “with the line-of-sights” can be performed thus implementing the sweeping scanning scheme, e.g., in a boiler of a power plant.

Series of measurements on the lab-scale burner implementing the parallel and sweeping scanning schemes were performed (Sec. 2.3). The measurements were conducted with the **multichannel IR spectrometer system** (the first prototype of the two-dimensional time-dependent (2Dt) IR tomography system which was developed in this PhD project, Sec. 2.1, Fig 1, see also the summary on it in Sec. 5.2) and with the FTIR spectrometer equipped with an InSb detector (as the IR camera of the multichannel system, Sec. 2.1.2) and the same optics as used in the multichannel system (Sec. 2.3.3).

The FTIR spectrometer was used to cover a broader spectral range compared to that of the multichannel system (Sec. 2.1.4, Fig. 3). Also, the multichannel system is sensitive to temperature variations of the housing of the IR camera employed in the system (Sec. 2.1.9). The FTIR spectrometer provides more thermal stability and a higher spectral resolution. It was decided to use the FTIR spectrometer for tomographic measurements on the burner. The results of the spectral measurements are input data to the developed tomographic algorithms for gas temperature profile reconstruction (Sec. 3.3). It is important for testing of the developed algorithms to have the spectral measurements with minimum uncertainties since tomographic reconstruction algorithms are known to be very sensitive to uncertainties in the input data (Sec. 3.3).

The goal of the tomographic measurements on the lab-scale burner was to compare the temperature profiles obtained from the parallel and sweeping scanning measurements in this PhD project with those published in Ref. [74]. The two equivalence ratios  $\varphi = 1$  (stoichiometric combustion) and  $\varphi = 0.8$  (lean combustion) were used.

Qualitatively the temperature profiles obtained from these measurements (Fig. 36, Fig. 37, Fig. 38, Fig. 39) are in agreement with those of Ref. [74] in that the temperature of the flame is at about its highest value at around the centre of the burner plate and is decreasing towards the edges.

Quantitatively, however, the obtained temperature profiles at  $\varphi = 1$  significantly underestimate the temperature profile of Ref. [74] at  $7 \text{ mm} < r < 30 \text{ mm}$  ( $r = 30 \text{ mm}$  corresponds to the edge of the burner plate and  $r = 0$  corresponds to the centre of the burner plate) and overestimate it at  $r > 30 \text{ mm}$ . The deviation from the results of Ref. [74] is greater than the standard deviation of the temperature points in Ref. [74] (Sec. 2.2.6). Also, the obtained temperature profiles do not exhibit a sharp temperature gradient at  $25 \text{ mm} < r < 35 \text{ mm}$  which was observed in Ref. [74]. Between  $r = 0$  and  $r = 7 \text{ mm}$ , however, there is a very good agreement between the measurements here and in Ref. [74].

The underestimation at  $7 \text{ mm} < r < 30 \text{ mm}$  can be due to the flow rates of methane and air in this work were about two times lower than those of Ref. [74] at  $\varphi = 1$  (Sec. 2.2.4). It follows, however, from Ref. [74] that the temperature profile is inde-



pendent of the actual values of the flow rates but that it depends only on the height above the burner plate and the equivalence ratio  $\phi$  which were the same in this work as in work [74] (Sec. 2.2.4). Also, higher flow rates would, probably, produce a flame having a temperature profile with a sharper gradient at about  $25 \text{ mm} < r < 35 \text{ mm}$ . Hence that can also partially explain the small gradient in the profiles of this work at about  $25 \text{ mm} < r < 35 \text{ mm}$ .

The wide field of view of the optical setup ( $\varnothing 6 \text{ mm}$ , Sec. 2.3.3) compared to the size of the burner ( $\varnothing 60 \text{ mm}$ ) can be the reason for the overestimation of the gas temperature at  $r > 30 \text{ mm}$  (Fig. 36, Fig. 37, Fig. 38, Fig. 39) due to “catching” the emission from the neighboring hotter flame layers.

The temperature profiles obtained in this PhD project for lean combustion ( $\phi = 0.8$ ) from both parallel and sweeping scanning measurements overestimate the temperature profile from Ref. [74] at almost all positions  $r$  (Fig. 38, Fig. 39). It should be noted that the flow rates at  $\phi = 0.8$  for both methane and air during the measurements in this PhD project were not two times higher than those in the work of Ref. [74] (like in case of  $\phi = 1$  discussed above, Sec. 4.1.2, Sec. 2.2.4) but were almost the same (more precisely, about 1.1 times higher than those of [74], Sec. 2.2.4).

It was interesting to note that a good agreement between the temperature profiles for  $\phi = 1$  and  $\phi = 0.8$  at  $r > 15 \text{ mm}$  (for parallel scanning curves, Fig. 38) and at  $r > 20 \text{ mm}$  (for sweeping scanning curves, Fig. 39) was observed for the two independent measurements with the two different schemes: the parallel scanning and the sweeping scanning.

Moreover the temperature difference between the profiles for  $\phi = 1$  and  $\phi = 0.8$  was about 70-167 K at  $r < 10 \text{ mm}$  for both measurements performed in this PhD project, whereas the difference between the temperature profiles for  $\phi = 1$  and  $\phi = 0.8$  from Ref. [74] was about 335 K (Fig. 40).

Some remarkable consistency between the temperature profiles obtained in this PhD project in the different measurement sessions using similar schemes and at similar conditions was observed. E.g., the temperature profiles obtained from the parallel scanning measurements performed in September 2011 and those performed recently in July 2012 were in excellent agreement (Fig. 37).

The above observations allow to conclude that the deviation of the temperature profiles of this work from those of work [74] can be due to some parameters of the burner operation were different here with respect to that in work [74].

Another consideration is that the calculations of the temperature profiles in this work were performed from the experimental spectral values at the wave number of  $2270.05 \text{ cm}^{-1}$  (Sec. 2.3.6, Sec. 2.3.8) which belongs to the  $4.3 \mu\text{m}$  ( $2350 \text{ cm}^{-1}$ ) band of  $\text{CO}_2$ . The presence of even minor quantities of cold  $\text{CO}_2$  from the ambient air in the field of view of the optical setup has probably significant effect on the results of the temperature profile calculations taking into account the fact that the tomographic algorithm was used to obtain the profiles whereas tomographic algorithms in general are known to possess an intrinsic property of being sensitive to minor uncertainties in the experimental data.

The above fact should also be taken into account since the profiles of Ref. [74] were obtained from the point measurements whereas in this work they were obtained from the line-of-sight measurements with subsequent application of the developed tomographic algorithm. It should be stressed that the application on the industrial scale of the

techniques employed in work [74], which are laser-based techniques (Sec. 2.2.6), could probably be more difficult and expensive since they are known to be critical to the alignment and require extensive optical access. On the contrary, the method of sweeping scanning (Sec. 2.3.8) developed in this project is expected to be promising for applications on industrial scale since it was designed with a view to a limited optical access. Also, line-of-sight measurements are much more time-efficient than the point measurements when the whole temperature field has to be imaged.

As well-known, tomographic reconstruction techniques require as many line-of-sights as possible and the field of view of each line-of-sight should be as narrow as possible. This requirement is often satisfied in medical tomographic systems where the number of line-of-sights reaches 100-1000 and the reconstruction can be considered to take place on a semi-continuum line-of-sight scale. However in the tomography of gases, flames or plasmas this number is usually limited to significantly lower values, typically 5-20, and therefore the reconstruction takes place on a discrete scale. In this case a tomographic reconstruction algorithm gives not exactly expected temperature profile, but on the contrary, it gives only an approximation to it. The quality of approximation can be improved significantly by taking into account all possible a priori information about the system (i.e. the burner) like, e.g., highest/lowest temperature limits, temperature fluctuations or noise in the experimental data.

It appears that the 6 mm field of view of the optical setup (Sec. 2.3.3) is not narrow enough. Narrowing the field of view would require a more sophisticated experimental setup. Also, as mentioned above, only one wave number in the CO<sub>2</sub> (hot) band was used in the algorithm. The use of a broader spectral range covering the “hot” and “cold” parts in the CO<sub>2</sub> band would be beneficial. The use of an OH band in the UV range could also be an advantage because that would give the flame temperature and not the temperature of the cold/hot CO<sub>2</sub>. These challenges have to be considered for the future developments in the project. On the other hand, considering the large scale (e.g., a power plant boiler) the 6 mm field of view is quite small and the number of line-of-sights can be very large (typically, the dimensions of power plant boilers are of the order 10 × 10 m).

Therefore the developed IR tomography system and methodology is expected to be feasible and promising for application on the large industrial combustion systems rather than on the small lab-scale burners. First step in further developments will require establishing the access ports and that should be done in collaboration with an industrial partner. Based on the results obtained in this PhD project as well as in the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” [8], a possible continuation of the project in terms of a new project application in collaboration with an industrial partner (e.g., Vattenfall A/S and/or DONG Energy) is under consideration.

## 5.2. Simultaneous Fast Exhaust Gas Temperature Measurements on the Large Diesel Engine

The **multichannel IR spectrometer system** as the first prototype of the two-dimensional time-dependent (2Dt) IR tomography system was developed in this part of the PhD project for simultaneous fast time-resolved transient IR spectral measurements at several line-of-sights (Sec. 2.1, Fig 1).

The system was successfully applied on industrial scale for simultaneous fast exhaust gas temperature measurements in the three optical ports of the exhaust duct of the large Diesel engine (Sec. 2.4).

The development and the application of the system were also relevant to the EU-funded HERCULES-B project “High-efficiency engine with ultra-low emissions for ships” carried out under 7<sup>th</sup> Framework Program “Sustainable Surface Transport” (grant agreement SCP7-GA-217878) presented in reports [9], to the project Energinet.dk ForskEL projektnr. 2008-1-0079 “Fast optical measurements and imaging of flow mixing” reported in Ref. [11] and to the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8].

It should be noted that the simultaneous fast time-resolved transient IR spectral measurements in the three optical ports of the exhaust duct of the large Diesel engine were performed for the first time. The gas temperature was obtained simultaneously at the three ports from the simultaneous IR emission measurements as the brightness temperature of the intensity of the 4.3  $\mu\text{m}$   $\text{CO}_2$  band (Sec. 3.2).

The negative values in the 2300-2400  $\text{cm}^{-1}$  range were observed in the emission and transmission spectra of the exhaust gas at the three optical ports of the exhaust duct (Fig. 41, Fig. 31). That is partially due to the low signal level in that range which, in turn, is due to the strong absorption by  $\text{CO}_2$  in that range [17, 69, 75]. The negative values can also be partially attributed to the reflections in the optical setup of the system and to some thermal drift of the InSb array of the IR camera used in the **multichannel IR spectrometer system**. The array was observed to be sensitive to variations in the ambient temperature (Sec. 2.1.9). The latter can be improved by placing the system into a temperature chamber. The effects caused by the reflections can be reduced by covering the optics surfaces with a special antireflection coating.

It was also observed (Fig. 41) that in the range 2150-2250  $\text{cm}^{-1}$  the transmission signal was higher than the reference signal. The absorption by  $\text{CO}_2$  is negligible in that region which means that the transmission and reference signals are expected to be equal. The higher values of the transmission signal are due to some differences in the alignment of the optical elements in the experiment on the transmission measurements compared to that on the reference measurements. The reference source was held in hands in front of the corresponding optical ports of the exhaust duct (Sec. 2.4.3) during both experiments. That can be improved by making a special setup with three reference sources mounted together such that all the three sources would be fixed in their positions in front of the optical ports.

The exhaust gas temperature was obtained simultaneously at the three optical ports (top, middle, bottom) from the simultaneous IR emission measurements as the brightness temperature of the intensity of the 4.3  $\mu\text{m}$   $\text{CO}_2$  band (Sec. 3.2).

The spectral range  $2270\text{-}2290\text{ cm}^{-1}$  was used for the gas temperature calculation (Sec. 3.2) since in that region the emission and transmission signals (Fig. 41, Fig. 31) have maximum signal-to-noise ratio, the absorption by cold  $\text{CO}_2$  present in the optical path of the system is weak and the spectral absorptance is close to unity in that region (Fig. 32, Sec. 3.2.3).

In the range  $2270\text{-}2290\text{ cm}^{-1}$  the assumption of spectral absorptance being equal to 0.9 was found to be a reasonable approximation based on the analysis of the emission and transmission spectra (Sec. 3.2.3). The exhaust gas temperature was then calculated as the brightness temperature from the simultaneous IR emission spectra using the spectral emission-absorption method (Sec. 3.2.2, Sec. 3.2.4).

The gas temperature must be known to measure, e.g., NO concentration by optical spectroscopy as well as to measure other pollutant species concentrations or other quantities characterizing combustion phenomena. The exhaust gas temperature in the three optical ports (top, middle, bottom) of the exhaust duct was also obtained from the structure of the UV NO 226 nm absorption band using a different system [9, 10]. A reasonable agreement between the IR  $\text{CO}_2$  measurements of the exhaust gas temperature (performed using the **multichannel IR spectrometer system**) and the UV NO measurements was observed [9, 10] (Fig. 42).

The temperature behavior of the exhaust gas in the three ports of the exhaust duct showed the expected periodicity in accordance with the stroke period of the engine.

The UV NO and IR  $\text{CO}_2$  measurements were performed at the same engine running settings but at different time points [9, 10]. The UV NO and IR  $\text{CO}_2$  measurements agree within 9% at the time points of the exhaust valve openings that can be considered as good agreement since the relative error of the UV NO measurements was estimated to  $\pm 10\%$  [9, 10]. The relative error of the IR  $\text{CO}_2$  measurements was estimated to  $\pm 3\%$  (Sec. 2.1.9).

The UV NO and IR  $\text{CO}_2$  measurements both gave lower gas temperature at the bottom optical port compared to that at the middle and top optical ports when the exhaust valve is opened. This result is in agreement with the expected convective paths of the hot exhaust gas in the exhaust duct.

Also, the exhaust gas temperature varied quickly upon the exhaust valve opening. When the valve opened a fast expanding and cooling down of the exhaust gas took place. The heat was then transferred to the thermocouple (which was mounted inside the exhaust duct close to the top optical port) that showed an increase in its own temperature (Fig. 42).

In the range  $2270\text{-}2290\text{ cm}^{-1}$  (used for the exhaust gas temperature calculation, as mentioned above, Sec. 3.2) the absorption by cold  $\text{CO}_2$  is weak. Cold  $\text{CO}_2$  is present, e.g., near the windows of the optical ports of the exhaust duct and in the optical path of the **multichannel IR spectrometer system**. The whole optical path should be flushed with an IR-inactive gas (e.g.,  $\text{N}_2$ ) in order to remove cold  $\text{CO}_2$  (as well as  $\text{H}_2\text{O}$ ) from the path for best results. The absorption by cold  $\text{CO}_2$  leads to a decrease in the calculated temperature values. That can be an explanation of the fact that the temperatures obtained from the UV NO measurements were on average higher by about  $50\text{ }^\circ\text{C}$  than the temperatures obtained from the IR  $\text{CO}_2$  measurements (Fig. 42).

The application of the **multichannel IR spectrometer system** (the first prototype of the 2Dt IR spectroscopic tomography system) on the large Diesel engine under real industrial conditions, when approach to a lot of practical challenges has to be found and the time is extremely limited, was a first industrial test of the optical tomography meth-

odologies developed in this PhD project. Although it was not reasonable to apply the tomographic algorithms to the data obtained from the measurements on the engine due to a much greater (than three) number of line-of-sights is required for the tomographic reconstruction of the temperature profiles (Sec. 5.1, Sec. 3.3), a valuable practical experience of the application on industrial scale was gained and the objectives for the future work were identified.

Based on the results obtained in this PhD project as well as in the project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” [8], a possible continuation of the project in terms of a new project application in collaboration with an industrial partner (e.g., Vattenfall A/S and/or DONG Energy) is under consideration.

### 5.3. Line-by-line Modeling of Gas Spectra

In the context of the PhD project, it was also important to investigate the spectral properties of major combustion species (such as CO<sub>2</sub> and CO) in the IR range at high temperatures to provide the theoretical background for the development of the optical tomography methods (Sec. 1.3.3).

The new software was developed for the line-by-line calculations of the transmission spectra of the CO<sub>2</sub>/CO mixture which is able to use within reasonable time the most recent but huge CDSD-4000 database [16] containing updated high-temperature spectroscopic line-by-line data.

The software was used for the line-by-line calculations of the transmission spectra of the CO<sub>2</sub>/CO mixture at high temperatures and the results were compared to the measurements in the high-temperature flow gas cell [2, 17, 18, 19, 20] which were carried out before the PhD project. The results of the calculations and the discussion are presented in a journal article [17] (*Evseev et al. JQSRT 113 (2012) 2222, 10.1016/j.jqsrt.2012.07.015*) included in the PhD thesis as an attachment (Appendix B).

The line-by-line modeling procedure, used for the calculations in this PhD project, is presented in detail in Sec. 3.1 and implemented in the mentioned above software developed using the C++ Programming Language (Microsoft Visual Studio 2008). The software was developed in this PhD project and the code is given in Appendix A. It should be noted that the detailed description of the line-by-line procedure as well as the software code are not given in Ref. [17] (which is included in the PhD thesis as an attachment, Appendix B).

# Acknowledgements

The author is grateful to Sønnik Clausen and Alexander Fateev for many help, valuable advice, guidance and consultancy in the work on the PhD project, to Karsten Lindorff Nielsen for technical support and to Prof. Valery S. Sizikov for the essential contribution to the work on the tomographic methodology and algorithms for gas temperature profile reconstruction and for the valuable consultancy on mathematical and computational issues in the PhD project.

The development of the multichannel IR spectrometer system and the IR tomography methodologies was funded by Energinet.dk (project Energinet.dk ForskEL projektnr. 2009-1-10246 “IR tomography in hot gas flows” reported in Ref. [8] and project Energinet.dk ForskEL projektnr. 2008-1-0079 “Fast optical measurements and imaging of flow mixing” reported in Ref. [11]).

The author is also grateful to MAN Diesel & Turbo, Denmark, for giving the exceptional opportunity of measurements on the large Diesel engine which were performed under the EU-funded HERCULES-B project “High-efficiency engine with ultra-low emissions for ships” carried out under 7<sup>th</sup> Framework Program “Sustainable Surface Transport” (grant agreement SCP7-GA-217878) presented in Ref. [9].

# References

- [1] Clausen, S.; Fateev, A.; Hvid, S.L.; Beutler, J.; Evseev, V. (2011): Combustion zone investigation in fuel flexible suspension fired boilers, Experimental, Report Risø-R-1751(EN), Risø National Laboratory for Sustainable Energy, Technical University of Denmark.
- [2] Fateev A, Clausen S. On-line non-contact gas analysis. Roskilde (Denmark): Risø National Laboratory, Technical University of Denmark; 2008 Mar. Report No.: Risø-R-1636(EN). Contract No.: Energinet.dk nr. 2006 1 6382. [http://orbit.dtu.dk/fedora/objects/orbit:82519/datastreams/file\\_4998276/content](http://orbit.dtu.dk/fedora/objects/orbit:82519/datastreams/file_4998276/content)
- [3] Sønnik Clausen. Infrarøde temperatur- og gasmålinger Haderslev Kraftvarmeværk. Roskilde (Denmark): Forskningscenter Risø, Danmarks Tekniske Universitet; April 2007. Report No.: Risø-R-1566(DA). Kontrakt nr.: PSO-projekt 5727 Online driftoptimering af affaldsfyrede anlæg. [http://orbit.dtu.dk/fedora/objects/orbit:79894/datastreams/file\\_7703121/content](http://orbit.dtu.dk/fedora/objects/orbit:79894/datastreams/file_7703121/content)
- [4] Bak J, Clausen S. FTIR emission spectroscopy methods and procedures for real time quantitative gas analysis in industrial environments. Meas Sci Technol 2002;13:150-6. <http://dx.doi.org/10.1088/0957-0233/13/2/302>
- [5] Bak, Jimmy; Clausen, Sønnik. FTIR transmission-emission spectrometry of gases at high temperatures: Demonstration of Kirchhoff's law for a gas in an enclosure. J. Quant. Spectrosc. Radiat. Transfer, vol: 61, pages: 687-694, 1999. <http://orbit.dtu.dk/en/publications/ftir-transmissionemission-spectrometry-of-gases-at-high-temperatures-demonstration-of-kirchhoffs-law-for-a-gas-in-an-enclosure%284a964044-5fc2-4fa0-9cd8-984533cff4e1%29.html>
- [6] Clausen, Sønnik; Bak, Jimmy. FTIR transmission emission spectroscopy of gases at high temperatures: Experimental set-up and analytical procedures. J. Quant. Spectrosc. Radiat. Transfer, vol: 61, pages: 131-141, 1999. <http://orbit.dtu.dk/en/publications/ftir-transmission-emission-spectroscopy-of-gases-at-high-temperatures-experimental-setup-and-analytical-procedures%28d08682a7-7958-4bd5-a309-bc32f6d229c2%29.html>
- [7] Clausen, S. (1996): Local measurement of gas temperature with an infrared fibre-optic probe, Measurement Science & Technology 7 (6), pages 888-896. <http://orbit.dtu.dk/en/publications/local-measurement-of-gas-temperature-with-an-infrared-fibreoptic-probe%28aba4cc07-977f-465c-a701-afde4876ddd1%29.html>
- [8] Vadim Evseev, Alexander Fateev, Valery Sizikov and Sønnik Clausen. IR Tomography in Hot Gas Flows. Published by: Technical University of Denmark, Department of Chemical and Biochemical Engineering, 2800 Kgs. Lyngby, Denmark (November 2012).









- [75] Oberly R., Narahari Rao K., Hahn Y.H., McCubbin T.K., Jr. Bands of Carbon Dioxide in the Region of 4.3 Microns, *Journal of Molecular Spectroscopy*, Vol. 25, pp. 138–165, 1968.
- [76] Fleckl T., Jäger H., Oberberger I. Experimental verification of gas spectra calculated for high temperatures using the HITRAN/HITEMP database // *J. Phys. D: Appl. Phys.* 2002. Vol. 35. P. 3138–3144.
- [77] Rothman L.S., et al. The HITRAN molecular spectroscopic database and HAWKS (HITRAN atmospheric workstation): 1996 edition // *J. Quant. Spectrosc. Radiat. Transfer*. 1998. Vol. 60. No. 5. P. 665–710.
- [78] J. Fischer, R.R. Gamache, A. Goldman, L.S. Rothman, and A. Perrin. Total internal partition sums for molecular species in the 2000 edition of the HITRAN database. *Journal of Quantitative Spectroscopy & Radiative Transfer* 82 (2003) 401–412. Fortran programs for Partition Sums are available on [ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global\\_Data/Fortran%20programs%20for%20Partition%20Sums/](ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Sums/)
- [79] Philip W. Morrison, Jr. and Oranut Taweechokesupsin. Calculation of Gas Spectra for Quantitative Fourier Transform Infrared Spectroscopy of Chemical Vapor Deposition. *J. Electrochem. Soc.* 145, 3212 (1998), DOI:10.1149/1.1838788
- [80] Franz Schreier. Optimized implementations of rational approximations for the Voigt and complex error function. *Journal of Quantitative Spectroscopy & Radiative Transfer* 112 (2011) 1010–1025.
- [81] S.M. Abrarov, B.M. Quine. Efficient algorithmic implementation of the Voigt/complex error function based on exponential series approximation. *Applied Mathematics and Computation* 218 (2011) 1894–1902
- [82] L.S. Rothman et al. The HITRAN 2004 molecular spectroscopic database. *Journal of Quantitative Spectroscopy & Radiative Transfer* 96 (2005) 139–204.
- [83] Tourin R.H., Krakow B. Applicability of Infrared Emission and Absorption Spectra to Determination of Hot Gas Temperature Profiles. *Applied Optics*. 1965. Vol. 4. No. 2. P. 237–242.
- [84] Heland J., Haus R., Schafer K. Remote sensing and analysis of trace gases from hot aircraft engine plumes using FTIR-emission-spectroscopy. *The Science of the Total Environment* 158 (1994) 85–91.
- [85] Bronshtein I.N., Semendyaev K.A. *Reference Book on Mathematics for Engineers and Students of Institutes of Technology*. Ed. 13th. – Nauka: Moscow, 1986. 544 pp. (in Russian).
- [86] Natterer F. *The Mathematics of Computerized Tomography*. – Wiley: Chichester, 1986. 288 pp.
- [87] Sizikov V.S. *Mathematical Methods for Processing the Results of Measurements*. – Politekhnik: St.-Petersburg, 2001. 240 pp. (in Russian).
- [88] Petrov Yu. P., Sizikov V.S. *Well-Posed, Ill-Posed, and Intermediate Problems with Applications*. – Leiden–Boston: VSP, 2005. 234 pp.
- [89] Sizikov V.S., Smirnov A.V., Fedorov B.A. Numerical solution of the Abelian singular integral equation by the generalized quadrature method // *Russian Mathematics (Iz. VUZ)*. 2004. Vol. 48. No. 8. P. 59–66.
- [90] Åkesson E.O., Daun K.J., Parameter selection methods for axisymmetric flame tomography through Tikhonov regularization. *Applied Optics*. 2008. Vol. 47. No. 3. P. 407–416.
- [91] Engl H.W., Hanke M., Neubauer A. *Regularization of Inverse Problems*. – Dordrecht: Kluwer, 1996. 328 pp.
- [92] Verlan' A.F., Sizikov V.S. *Integral Equations: Methods, Algorithms, Programs*. – Nauk. dumka: Kiev, 1986. 544 pp. (in Russian).

## Appendix A.

# The C++ Code of the Software for the Line-by-line Calculations

As mentioned in the objectives of the PhD project (Sec. 1.3.3), this appendix contains the C++ code of the software developed in this PhD project for the line-by-line calculations of the effective transmittance spectrum of the mixture of CO<sub>2</sub> and CO using the line-by-line parameters from the HITEMP-1995 [12], HITEMP-2010 [13], CDSD-

HITEMP [14] or CDS-4000 [16] databases. The software follows the procedure described in Sec. 3.1. The results of the calculations are presented in Ref. [17] (Ref. [17] is included in this PhD thesis, Appendix B). It should be noted that the software code is not given in Ref. [17].

File "CDSHITEMP\_MixtureOfCO2andCO.ini"

DTU Kemiteknik 2012

-----  
-----  
This is the main input file for the CDS-4000 application.

NB:

You can not change the name of this file.

It must be located in the same directory as the executable \*.exe.  
-----  
-----

#####  
#####

Special Features Summary:

- Calculates effective transmissivity and absorbance for the mixture of CO<sub>2</sub> and CO.

- Calculates true values (Cross-Sec [cm<sup>2</sup>/molecule], True Absorbance, True

Transmissivity) for CO<sub>2</sub>, CO, and the mixture of those.

- Some minor coefficients, constants and reference data points have now improved values.

#####  
#####

:::  
::::::::::::

NB:

You can only change the values inside the quotes.

You can not change the order of the entries.

You can not change the units.

You can not add more quote symbols than there are already here.

:::  
::::::::::::

:::  
::::::::::::

NB:

Regarding disc space:

The free space available on your working disc must be at least equal to the space occupied by the original database files.

It is because the program will create database files in the binary format from the original files

in order to speed up the calculations. If the binary database files already exist (they have been created sometime before) then they will be used.

And in this case the program anyway needs some space in order to be able to at least create

the file with the results.

:::  
::::::::::::

Entry

1 Working directory:

"C:\Documents and Settings\vaev\My Documents\1500C"

Copy it from the windows explorer address bar and paste here. It must not contain any extra spaces.

2 Output file which will contain all the results:

"cpp\_2000\_2030\_CDSDHITEMP\_NoCO2\_OnlyCO\_014\_CO2\_100%\_1500C\_0125.dat"  
 cpp50\_500\_850\_CDSDHITEMP\_ILSTriang2cm1\_exp\_1200C\_Fig7.dat  
 It will be created in the working directory. The file name must not contain any extra spaces.

3 File with a wave number axis for transmissivity calculation:  
 "CO2\_100%\_1500C\_0125.txt" exp\_1200C\_Fig7.txt  
 Only the first column is processed. Subsequent columns (if any) are ignored. The first column must contain the wave numbers for transmissivity calculation. The file name must not contain any extra spaces.  
 This file must be located in the working directory (see entry 1).

4 Directory containing the CDSD-HITEMP files:  
 "C:\Documents and Settings\vaev\My Documents\HITRAN"  
 Copy it from the windows explorer address bar and paste here. It must not contain any extra spaces.

5 Range for the CO2 (Carbon Dioxide) CDSD-HITEMP files:  
 "1500 2000" 3000 3750 3500 3500 2250 2250 0 750  
 These are the STARTING wave numbers XXXX for the first and the last CDSD-HITEMP files to be processed.  
 CDSD-HITEMP files for CO2 (Carbon Dioxide) follow the name pattern  
 "cdsd\_hitemp\_XXXX\_YYYY",  
 where XXXX is the initial wave number and YYYY is the final wave number. XXXX and YYYY represent the spectral range of a given file.  
 You should specify only the STARTING wave numbers XXXX for the FIRST and the LAST CDSD-HITEMP files to be processed.  
 E.g. 1500 2500 cm-1 means the following files will be used:  
 cdsd\_hitemp\_1500\_2000,  
 cdsd\_hitemp\_2000\_2125,  
 cdsd\_hitemp\_2125\_2250,  
 cdsd\_hitemp\_2250\_2500,  
 cdsd\_hitemp\_2500\_3000.

NB:  
 The data for only the 8 most abundant isotopes is used. The data for the other isotopes is ignored.

NB:  
 If the binary database files do not exist then the program will create them.  
 The binary database files have the ending \_b10. E.g. cdsd\_hitemp\_1500\_2000\_b10, cdsd\_hitemp\_2000\_2125\_b10 and so on.

NB:  
 The CO database file is defined in the code of the software (see the output file for the CO database file name).

6 Spectral range for the transmissivity calculation:  
 "2000 2030" cm-1 3100 3900 3585 3665 2340 2440 1800 2500 2340 2440 500 850  
 It should be in reasonable correspondence with the above range for the CDSD-HITEMP files.  
 E.g. for the above example of the range for the CDSD-HITEMP it can be 1500 3000 cm-1.

7 The reference temperature of ALL database files (for CO2 and CO):  
 "296" K 296  
 Must be in the range 135 to 2935 K (the actual limits built in the software may differ slightly depending on the version)

8 The gas temperature (assumed to be the same for CO2 and CO):  
 "1773.15" K 1773.15 1550 1473.15  
 Must be in the range 135 to 2935 K (the actual limits built in the software may differ slightly depending on the version)

9 The total pressure:  
 "1" atm  
 NB: It must not differ much from 1 atm. The calculations are designed for the pressure around 1 atm.

10 Path length:  
 "53.3" cm 53.3 50

11 The CO2 (Carbon Dioxide) mole fraction:  
 "10" % (per cent) 99.86 100

NOTE: The CO (Carbon Monoxide) mole fraction does NOT necessarily equals to: 100 % minus CO2 mole fraction; because the total mole fraction of CO2+CO can be < 100 %.

The CO (Carbon Monoxide) mole fraction is defined in the code of the software (see the output file for the value).

12 The ILS parameter:

"0.125" cm-1 0.125 8 4 2

The ILS function is sinc or Triangular which is defined in the code of the software (see the output file for the function type).

For sinc, the ILS parameter is the full width of the main maximum at zero. sinc is normally used for smaller values of the ILS parameter.

For triangular, the ILS parameter is the length of the base of the isosceles triangle (which represents the triangular ILS function).

Triangular is normally used for higher values of the ILS parameter.

```
// File "Define.h"

// Contains
// #define directives.

// !!! NB:
// !!! Many of these values are actually physical, chemical or mathematical constants;
// !!! A special care must be taken when editing this file.

// #define Directives.

// System Values.

////////////////////////////////////
//
// The database type:
//
// #define defDBType (defDBType_HITEMP2010) // HITEMP/HITRAN/CSD-1000
// #define defDBType (defDBType_CSDSFourk) // CSD-4000
// #define defDBType (defDBType_CSDHITEMP) // CSD-HITEMP ftp://ftp.iao.ru/pub/CSD-
// HITEMP/Readme%20for%20cdsd-hitemp.pdf
//
//
// The database name:
//
// #define defDBName L"HITEMP2010"
// #define defDBName L"CSD-4000"
// #define defDBName L"CSD-HITEMP"
//
//
// The database file name pattern:
//
// #define defDBFileNamePattern L"02_%04u-%04u_HITEMP2010.par" // HITEMP-2010
// #define defDBFileNamePattern L"cdsd_%05u_%05u" // CSD-4000
// #define defDBFileNamePattern L"cdsd_hitemp_%04u_%04u" // CSD-HITEMP
// ftp://ftp.iao.ru/pub/CSD-HITEMP/Readme%20for%20cdsd-hitemp.pdf
// #define defDBFileNamePattern L"cdsd_hitemp_%u_%u" // CSD-HITEMP
// ftp://ftp.iao.ru/pub/CSD-HITEMP/Readme%20for%20cdsd-hitemp.pdf
// #define defDBFileNamePattern L"test1PairMol_v1_cdsd_hitemp_%04u_%04u" // CSD-HITEMP
// ftp://ftp.iao.ru/pub/CSD-HITEMP/Readme%20for%20cdsd-hitemp.pdf
//
//
// The number of parameters (which are used in the calculations) given in the database:
//
// #define conDBNumOfParams (conDBNumOfParams_Standard) // HITEMP/HITRAN/CSD-1000
// #define conDBNumOfParams (conDBNumOfParams_CSDPlus) // CSD-4000/CSD-HITEMP
```

```

//
//
// ILS Type
//
#define defILSType (defILSType_sinc)
// #define defILSType (defILSType_Triangular)
//
// The field lengths of the parameters specified above (!!! NOT IN USE !!!):
//
// #define conDBFieldLengths (conDBFieldLengths_Standard) // !!! NOT IN USE !!!
HITEMP/HITRAN/CDSD-1000/CDSD-4000
// #define conDBFieldLengths (conDBFieldLengths_CDSDHITEMP) // !!! NOT IN USE !!! CDSD-
HITEMP ftp://ftp.iao.ru/pub/CDSD-HITEMP/Readme%20for%20cdsd-hitemp.pdf
//
//
// The name of the main input file (*.ini):
//
// #define defMainInputFileName (_T("HITEMP2010Mix_CO_004.ini"))
// #define defMainInputFileName (_T("CDSD4000CutoffMix_v1_CO_014pc.ini"))
// #define defMainInputFileName (_T("CDSDHITEMPPairMol_v1.ini"))
#define defMainInputFileName (_T("CDSDHITEMP_NoCO2_OnlyCO_ILSsinc.ini"))
//
//
//
//
// CO (Carbon Monoxide) Parameters
//
// The mole fraction of CO (Carbon Monoxide):
//
#define defMoleFract_CMonox (0.14) // in [%] 0.14 0.04
//
//
// The name of the database file for CO (Carbon Monoxide)
//
#define defDBFileName_CMonox (_T("05_HITEMP2010new.par"))
//
//
//
//
#define defRemTimeDisplayIntervalInSec (60) // in seconds
#define defRemTimeDisplayAdvanceInSec (60) // in seconds

#define defBinaryDBFileNameSuffix L"_b10"

#define defMainInputFileNumOfEntries (12)

#define defRadix (10)

// #define defBufVerySmall 64
#define defBufSmall (128)
#define defBufLarge (1024) // The following snippet contains 1067 characters (with spac-
es)

#define defNumOfElems_Static(Array) (sizeof Array / sizeof Array[0])
#define defNumOfElems_Dynamic(Array) (_msize(Array) / sizeof Array[0])
//
// _msize

#define defMaxNumOfDataBFiles (ULONG_MAX)

#define defMaxArrSize (4096) // (UINT_MAX - UCHAR_MAX) // 25000 allowed; 30000 - stack over-
flow
// #define defMaxAlloc (1048576) // (1048576) // 2^20 elements = 8 MB = 8388608 Bytes for
double (8 bytes)
#define defMaxAlloc (8388608) // (16777216) // 2^23 elements = 64 MB = 67108864 Bytes for
double (8 bytes)
// Verif.: 8388608

#define defWNStopMax (99999+1)

#define defProfileType_VoiWhi (0x0147)

#define defILSType_sinc (0x1102) // sinc, the ILSParam is the full width of the main
maximum at zero

```

```

#define defILSType_Triangular (0x18B) // Triangular, the ILSParam is the length of the
base of the isosceles triangle

#define defDBType_HITEMP2010 (0x27B)
#define defDBType_CDSDOnek (0x1508) // the first edition of CDSD-1000 where Tref=1000 K.
The parameter is NOT in use so far.
#define defDBType_CDSDFourk (0x0902) // CDSD-4000
#define defDBType_CDSDHITEMP (0x2401) // CDSD-HITEMP ftp://ftp.iao.ru/pub/CDSD-
HITEMP/Readme%20for%20cdsd-hitemp.pdf

// Constants
// relating to spectroscopy, physics, chemistry or mathematics.

#define conDBNumOfParams_Standard (10) // HITEMP/HITRAN/CDSD-1000
//#define conDBNumOfParams (10) // HITEMP/HITRAN/CDSD-1000
#define conDBNumOfParams_CDSDPlus (11) // CDSD-4000/CDSD-HITEMP
//#define conDBNumOfParams (11) // CDSD-4000

#define conDBNumOfParamsInUse (10)
#define conDBFieldLengths_Standard 2, 1, 12, 10, 10, 5, 5, 10, 4, 8, 5 //
HITEMP/HITRAN/CDSD-1000/CDSD-4000
// Verification: 2 1 12 10 10 5 5 10 4 8 5
// HITEMP-2010: 2 1 12 10 10 5 5 10 4 8 // !!! There
is no last field // ftp://ftp.iao.ru/pub/CDSD-HITEMP/HITEMP_paper.pdf
// CDSD-4000: 2 1 12 10 10 5 5 10 4 8 5 //
ftp://ftp.iao.ru/pub/CDSD-4000/Readme%20for%20CDSD-4000.pdf
// CDSD-1000: 2 1 12 10 10 5 5 10 4 8 // The
format of databank corresponds to the HITRAN96 format
// HITRAN-2004: 2 1 12 10 10 5 5 10 4 8
// HITRAN-1986-2001: 2 1 12 10 10 5 5 10 4 8
#define conDBFieldLengths_CDSDHITEMP 2, 1, 12, 10, 10, 5, 5, 10, 4, 8, 5
// Verification: 2 1 12 10 10 5 5 10 4 8 4 // !!! A
mistake in ftp://ftp.iao.ru/pub/CDSD-HITEMP/Readme%20for%20cdsd-hitemp.pdf
// CDSD-HITEMP: 2 1 12 10 10 5 5 10 4 8 4 // !!! A
mistake in ftp://ftp.iao.ru/pub/CDSD-HITEMP/Readme%20for%20cdsd-hitemp.pdf

#define conDataBaseFieldLenDeclar const USHORT DataBaseFieldLen[] =
{conDBFieldLengths_Standard}

#define conDataBaseLineLen (defBufLarge)

#define conILSHalfXCoeffDefault (1.1)//(1.1)
#define conXUser2ModelExtCoeff (2.)//(sqrt(2.))
#define conFactorForMinGamma (4.)
#define conXModelMinStep (0.001)

#define conLineProfileWingCutoffCoeff (1e-4)
#define conMaxLProfWingCutoffValuecm1 (50) // cm-1 LProfWingCutoffValuecm1=50cm-1 is
used in Depraz, Perrin, Rivière, Soufiani, JQSRT, Available online 21 September 2011,
10.1016/j.jqsrt.2011.09.013
http://www.sciencedirect.com.globalproxy.cvt.dk/science/article/pii/S0022407311003359
#define conILSWingCutoffCoeff (1e-3)
#define conMinWaveNumberCoeff (4.)

// Physical Constants

// All units are SI units unless otherwise is denoted in the field name:

// The following values are taken from
// http://en.wikipedia.org

#define con_c (299792458) // m/s
// verif.: 299792458 m/s
#define con_h (6.62606896e-34) // J*s
// verif.: 6.62606896e-34 J*s
#define con_kB (1.3806504e-23) // J/K
// verif.: 1.3806504e-23 J/K
#define con_K (273.15) // (used to convert from Celsius degrees to Kelvin and vice ver-
sa)
// verif.: 273.15
#define con_atm (101325) // Pa/atm
// verif.: 101325
#define con_NA (6.02214179e23) // 1/mol

```

```

// verif.:      6.02214179e23      1/mol
#define con_Ru (8.314472) // J/(K*mol)
// verif.:      8.314472      J/(K*mol)

// The following value is calculated as given in the comment to formula (A3) in the
HITRAN 1996 Article
#define con_c2_HITRunits (1.43877516016792) // cm*K
// verif.:      1.43877516016792      cm*K

// Chemical Constants

// The following atomic masses (weights) are taken from http://www.ptable.com/
#define con_C12 (12e-3) // [kg/mol]
//      12
#define con_O16 (15.99491461956e-3) // [kg/mol]
//      15.99491461956
#define con_C13 (13.00335483778e-3) // [kg/mol]
//      13.00335483778
#define con_O18 (17.999161001e-3) // [kg/mol]
//      17.999161001
#define con_O17 (16.999131703e-3) // [kg/mol]
//      16.999131703

// The following fractional abundances of isotopes are taken from
// ftp://ftp.iao.ru/pub/CDS-1000/readme-CDS-1000
// (see also the HITRAN 2004 article, table 6, p.148)

// A lot of useful molecular and spectroscopic data can be found on
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/molparam.txt

#define defNameShort_CDiox (_T("CO2"))
#define defNameLong_CDiox (_T("Carbon Dioxide"))
#define defNameShort_CMonox (_T("CO"))
#define defNameLong_CMonox (_T("Carbon Monoxide"))

#define conMolNum_CDiox (2)
#define conMolNum_CMonox (5)

#define conNumOfIsotopes_CDiox (8)
#define conNumOfIsotopes_CMonox (6)

#define conFracAbund_CDiox \
    .984204E+00,\
    1.10574E-02,\
    3.94707E-03,\
    7.33989E-04,\
    4.43446E-05,\
    8.24623E-06,\
    3.95734E-06,\
    1.47180E-06
#define defDeclar_FracAbund_CDiox const double FracAbund_CDiox[] = {conFracAbund_CDiox}

#define conFracAbund_CMonox \
    .986544E+00,\
    1.10836E-02,\
    1.97822E-03,\
    3.67867E-04,\
    2.22250E-05,\
    4.13292E-06
#define defDeclar_FracAbund_CMonox const double FracAbund_CMonox[] =
{conFracAbund_CMonox}

#define conMolW_CDiox \
    43.989830e-3,\
    44.993185e-3,\
    45.994076e-3,\
    44.994045e-3,\
    46.997431e-3,\
    45.997400e-3,\

```



```

47.998322e-3,\
46.998291e-3 // [kg/mol]
#define defDeclar_MolW_CDiox const double MolW_CDiox[] = {conMolW_CDiox}

#define conMolW_CMonox \
27.994915e-3,\
28.998270e-3,\
29.999161e-3,\
28.999130e-3,\
31.002516e-3,\
30.002485e-3 // [kg/mol]
#define defDeclar_MolW_CMonox const double MolW_CMonox[] = {conMolW_CMonox}

// Mathematical Constants

#define pi (3.1415926535897932384626433832795028841971) //
http://en.wikipedia.org/wiki/Pi
// 3.14159265358979323846264338327950288
// 3.14159265358979
// 3.14159265358979 MATLAB

// Some constants from the FORTRAN routine:
// ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Sums/T
IPS_2011.zip
// or more specifically from this code:
// ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Sums/B
D_TIPS_2011_v1p0.for

// or from
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat
// see also special MATLAB script C:\Documents and Settings\vaev\My Docu-
ments\EvsFatCla_Revisions1\parsum_dat_transpose_and_select_1.m
// and file C:\Documents and Settings\vaev\My Documents\HITRAN\parsum1_transposed.dat

//#define conparsum_NofTemps (119)
#define conparsum_NofTemps (2931) // ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat

#define defparsum_Temps_Declar const double constparsum_Temps[] =
{70,71,72,...,2998,2999,3000}
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat

// THE VALUES FOR THE FOLLOWING ARRAY SHOULD BE TAKEN FROM
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat
#define defparsum_CDiox_Declar const double constparsum_CDiox[] [conparsum_NofTemps] = {\
{...},\
{...},\
{...},\
{...},\
{...},\
{...},\
{...},\
{...},\
{...},\
{...}\
}

// THE VALUES FOR THE FOLLOWING ARRAY SHOULD BE TAKEN FROM
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat
#define defparsum_CMonox_Declar const double constparsum_CMonox[] [conparsum_NofTemps] =
{\
{...},\
{...},\
{...},\
{...},\
{...},\
{...}\
}
// ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat

```

```

// File "Header.h"

// Contains
//      function prototypes and class, struct and union definitions.

// Function Prototypes.

//void ErrorMessage(PCHAR);
void ErrorMessage(TCHAR*);
//void WelcomeMessage(PCHAR);
void WelcomeMessage(TCHAR*);
void MainExitPrompt();
void EmergencyTerminationPrompt();
BOOL PrintOpeningSeparatorToFileAndStdout(FILE*);
BOOL PrintClosingSeparatorToFileAndStdout(FILE*);

// Class Prototypes.
class CSpectralCalcs;

// Class Definitions.

class CTextBuffer{
public:
    size_t StrLen, StrLenTot, NumOfCopiedChars;
    ULONG Num;
};

class CTextBufferLarge{
public:
    static const size_t ArrLen=defBufLarge;//75+3+20-1;97; defBufLarge;
    TCHAR tBuf[ArrLen];

    CTextBufferLarge();

    void Initialise();
};

class CTextBufferLargeEx: public CTextBufferLarge, public CTextBuffer{
};

class CTextBufferSmall{
public:
    static const size_t ArrLen=defBufSmall;
    TCHAR tBuf[ArrLen];

    CTextBufferSmall();

    void Initialise();
    //friend void CTextBufferLarge::Initialise();
};

class CTextBufferSmallEx: public CTextBufferSmall, public CTextBuffer{
};

class CFile{
public:
    CTextBufferLargeEx FileName, Path, FullName;
    FILE* Stream;
    BOOL IsFullNameLong();
    BOOL CreateFullName();
    BOOL OpenFile(TCHAR*);
    BOOL CloseFile();
};

class CFileMultiple:public CFile{
public:

```

```

//TCHAR tExtension[defExtLenInclNull];
CTextBufferSmall Extension;
CTextBufferLargeEx FullNamePattern, FullNameCurrent, FileNameCurrent;

//BOOL CreateFullNamePattern();
//errno_t OpenFirstFile();
BOOL OpenCurrentFile(ULONG, ULONG, CSpectralCalcs*);

ULONG WaveNumStart, WaveNumStartStop;
};

class CFileMultipleBinary:public CFileMultiple{
public:
    BOOL CreateFullName(ULONG, ULONG, CSpectralCalcs*);
    BOOL CreateFileForWriting(CFileMultiple);
    BOOL OpenFileForReading(ULONG, ULONG, CSpectralCalcs*);
};

class CValue{
public:
    CTextBufferSmall ValTxt;
    double Val;
    BOOL GetVal(HWND, double*);
};

class CCounter{
//private:
public:
    ULONG Files, FileLines, LineSymbols, FileTrans, FileTransInUse, TransTotal,
    TransTotalInUse;
    double LinesTotal;

    CCounter();
    void Initialize();
};

class CTimeIntervalDD{
public:
    double DD, HH, MM;
    double SSSS;

    void ct_ConvFromSec(double);
};

class CClcTime{
public:
    clock_t GlobalStart, TimeIntervalStart, TimeIntervalInClocks;
    CTimeIntervalDD RemainingTime;
    double
        GlobalTimeElapsed, // [seconds]
        RemainingTimeInSec; // [seconds]
    time_t lTime;
    CTextBufferSmall BufFor_ctime;
    BOOL IsPossible;

    CClcTime();

    double ct_Clocks2Sec(clock_t);
    clock_t ct_Sec2Clocks(double);
    BOOL ct_SetGlobalTimer();
    BOOL ct_SetTimeIntervalTimer();
    BOOL ct_HasTimeIntervalElapsed();
    BOOL ct_GlobalTimeElapsed(); // [seconds]
    //void ct_ddhhmmsssss(double*, double*, double*, double*, double);
    //void ct_ddhhmmsssss(CTimeIntervalDD*, double);
    //void ct_ddhhmmsssss(double);
    BOOL ct_RemainingTimeForOutput();
};

class CClcTimeRemaining:public CClcTime{
public:
    //void ct_Clock();
    //defRemainingTimeDisplayInterval

```

```
};
```

```
// File "Spectr.h"

// Contains
//   global variables declarations,
//   function prototypes,
//   class, struct and union definitions
//   relating to physics, chemistry and mathematics, and in particular to spectroscopy

static conDataBaseFieldLenDeclar;

static defparsum_Temps_Declar;
static defparsum_CDiox_Declar;
static defparsum_CMonox_Declar;

static BOOL IsTimeToDestruct=FALSE;

// Class Definitions

class DataBaseField{
public:
    USHORT len, start, end;
};

class CDataBaseParameters{
public:
    ULONG M, Isot;
    double v0, S, g_air, g_self, E, n_air, delta_air, n_self;
};

class CMolecularWeight{
public:
    double CDiox[_max(conNumOfIsotopes_CDiox,conNumOfIsotopes_CMonox)]; // [kg/mol]
    double CMonox[_max(conNumOfIsotopes_CDiox,conNumOfIsotopes_CMonox)]; // [kg/mol]
    void CalcMolW();
};

class CConstants{
public:
    CMolecularWeight MolW; // [kg/mol]
    // Mathematical Constants
    double ln2, sqrt_ln2, sqrt_ln2bypi;

    void CalcCon();
};

class CAxis{
//private:
//   bool IsMemoryAllocated;
public:
    CAxis();
    static const size_t MaxAlloc=defMaxAlloc;//defMaxArrSize;
    double* Ax;
    size_t ActualLen;
    double Range[2];
    BOOL IsMemoryAllocated;
    double
        Start,
        Step,
```

```

        Finish;

        size_t ActualLenClcCeil();
        size_t ActualLenClcCeil(double param_start, double param_step, double param_finish);
        size_t ActualLenClcFloor();
        size_t ActualLenClcFloor(double param_start, double param_step, double
param_finish);
        BOOL AxCreate();
        double GetAxEnd();
        BOOL AllocateMemory(size_t);
        BOOL AllocMemFromActualLenAndMaxAlloc();
        ~CAXis();
};
//
class CXAxisUser:public CAxis{
public:
    static const size_t Size=defMaxArrSize;
    double Axis[Size], MainInputFileSpecifiedRange[2]; // [cm-1]
};

class CXAxisModel:public CAxis{
public:
    double
        ILS_HalfX, // [cm-1]
        FactorForMinGamma, // [1]
        XUser2ModelExt, // [cm-1]
        ProfileXThres; // [cm-1]

    LONG iiLeft, iiRight;

    //double ii_2_cm1(size_t);
    double cm1_2_ii(double);
    LONG iiLeftClc(double);
    LONG iiRightClc(double);

    //~CXAxisModel();
};

class CTauAbs{
public:
    CAxis Meas, True, Conv;
    double ta_TauTrue(double, double, double);
    double ta_TauTrue(double Absorbance);
    double ta_Absorbance(double, double, double);
    double ta_Absorbance(double);
    double ta_CrossSecFromTau(double tau, double n_molec_per_cm3, double path_cm); //
[cm2/molecule]
    double ta_CrossSecFromAbs(double Abs, double n_molec_per_cm3, double path_cm); //
[cm2/molecule]
};

class CSpectralCalcs{
public:
    CXAxisUser XUser; // [cm-1]
    CXAxisModel XModel; // [cm-1]

    CAxis kSum; // [1/(molecule cm-2)]
    CTauAbs Tau, Absorbance;

    double Tref, // [K]
        T, // [K]
        TRange[2], // [K]
        p, // [atm]
        ps, // [atm]
        path, // [cm]
        c, // [%]
        ILSParam; // [cm-1]
    BOOL TOutOfRange;

    size_t MolNum, NumOfIsotopes;

    double
        g_p, // [cm-1]

```

```

g_p_min, // [cm-1]
g_p_max, // [cm-1]
Tref_g_self, // [K]
v0sh, // [cm-1]
v_minus_v0sh, // [cm-1]
g_Dop, // [cm-1]
g_Voi, // [cm-1]
g_Voi_min, // [cm-1]
g_Voi_max, // [cm-1]
n_molec_per_cm3, // [molecule/cm3]
S_T, // [1/(molecule cm-2)]
Q_T[__max(conNumOfIsotopes_CDiox,conNumOfIsotopes_CMonox)], // [unitless]
// Q_TIntensCutoff[conNumOfIsotopes_CDiox], // [unitless]
Q_Tref[__max(conNumOfIsotopes_CDiox,conNumOfIsotopes_CMonox)], // [unitless]
QTref_by_QT[__max(conNumOfIsotopes_CDiox,conNumOfIsotopes_CMonox)]; //
[unitless]
//QTref_by_QT4000KCutoff[conNumOfIsotopes_CDiox], // [unitless]
//QTref_by_QT2000KCutoff[conNumOfIsotopes_CDiox], // [unitless]
//QTref_by_QT1000KCutoff[conNumOfIsotopes_CDiox]; // [unitless]

CTextBufferSmallEx
DataBaseType,
ProfileType,
ILSType,
MolNameLong,
MolNameShort;

CConstants Con;

CSpectralCalcs();
//~CSpectralCalcs();

double sp_ps();
double sp_g_p(CDataBaseParameters);
double sp_g_p(double n_air, double g_air, double n_self, double g_self);
double sp_v0sh(CDataBaseParameters);
double sp_v0sh(double v0, double delta_air);
double sp_g_Dop(double v0sh, double T, double MolW);
double sp_g_Dop(size_t Mol, size_t Isot);
//BOOL sp_g_Dop(size_t Mol, size_t Isot, CFile);
double sp_g_Voi(CDataBaseParameters);
double sp_g_Voi();
BOOL sp_Q(CFile);
double sp_AtoB(size_t Isot, double aa); // SUBROUTINE AtoB(aa,bb,A,B,npt) from
ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Sums/B
D_TIPS_2011_vlp0.for
double sp_InterpLagrange(size_t Isot, double x);
double sp_S_T(double QTrefByQT, double TrefK, double TK, double v0, double S, double
E_lower);
double sp_S_T(size_t Isot, double v0, double S, double E_lower);
//BOOL sp_IsAboveIntensCutoff(size_t Isot, double v0, double S, double E_lower);
double sp_ILS(double v);
BOOL sp_ILS(double* ILS, double* v, size_t N);
BOOL sp_ILS(double* ILS, double* v, double v0, size_t N);
BOOL sp_ConvolutionWithILS(CFile);

double sp_Trapz(double*, double*, size_t);
double sp_Trapz(double x1, double y1, double x2, double y2);
BOOL sp_ILS_HalfX(CFile);
BOOL XModelCreate(CFile);
BOOL LineProfileWingCutoff(CFile);
double sp_v_minus_v0sh(double);

double sp_Lor_Common(double, double);
double sp_Lor_Common();
double sp_VoiWhi(double, double, double);
double sp_VoiWhi(double);
double sp_VoiWhi();
double sp_VoiWhi_WithoutAssignment();
double sp_VoiWhi_WithoutAssignmentAndPow(double);
//double sp_Profile_Common(USHORT, double, double, double);

```

```

    BOOL DBFileFirstProc(double*, CFile);
    BOOL DBFileSecondProc(double*, CFile);

    double sp_n_molec_per_cm3();
    BOOL sp_TauTrue_AbsTrue_Clc();
    // BOOL Print_Tau_etc(CFile);

    BOOL DeallocMem();//
};

//class CCMonoxSpectralCalcs: public CSpectralCalcs{
//public:
//    CCMonoxSpectralCalcs();
//};

class CFileProcessing{
private:

    USHORT BinaryDBFileProcAgain, WNFileProcAgain, DBNumOfParamsTrimmed;
    ULONG FileLine, TotalNumOfTrans; // TotalNumOfTransInUse;
    //UINT MB_IconSoundType;
    CCounter Cntr;
    TCHAR
        tch,

    DataBStrWithSpaces [conDataBaseLineLen], //DataBStrWithSpaces [conDataBaseLineLen+const
NumOfDataBaseParams],
        *nptr,
        *endptr;
    CTextBufferSmall Num2TCHAR, BufSmall;
    CTextBufferLarge LabelLarge, BufLarge, Folder;

    DataBaseField DataBField[conDBNumOfParams];

    double ParamVal[conDBNumOfParams], ParamValInUse[conDBNumOfParamsInUse];
    CDataBaseParameters Params;
    CClcTime RemainingTimeEstimation;

public:

    CFile WaveNumberFile, OutputFile;
    CFileMultiple TextDBFile;
    CFileMultipleBinary BinaryDBFile;

    CFileProcessing();

    BOOL InitializationOfSpecClc(CSpectralCalcs* SpecClc);

    BOOL MainInputFileProcessing(CSpectralCalcs*);
    BOOL PreliminaryOutput(CSpectralCalcs, CSpectralCalcs);
    BOOL WaveNumberFileProcessing(CSpectralCalcs*);

    void DBFileGetFormat(CSpectralCalcs*);
    BOOL TextDataBaseFileProcessing(CSpectralCalcs*);
    BOOL BinaryDataBaseFileProcessing(CSpectralCalcs*);
    void FReadBinaryDBFileErrorMessage();
    void FWriteBinaryDBFileErrorMessage();
    void DBFileError_InvalidFormat();
    BOOL Print_Tau_etc(CSpectralCalcs* SpecClc, CSpectralCalcs CMonoxSpecClc);

    friend class CCMonoxFileProcessing;

};
//
class CCMonoxFileProcessing: public CFileProcessing{
public:
    CCMonoxFileProcessing();//
    BOOL InitializationOfCMonoxSpecClc(CSpectralCalcs* CMonoxSpecClc, CSpectralCalcs
SpecClc);
    BOOL AxesCopyForCMonoxSpecClc(CSpectralCalcs* CMonoxSpecClc, CSpectralCalcs SpecClc,
CFileProcessing FileProc);
    BOOL AbsorbanceTrueSummation(CSpectralCalcs* SpecClc, CSpectralCalcs CMonoxSpecClc);
};

```

```

//

// File "Spectr_Funcs.cpp"

// Contains
//     the definitions of functions which are used by the user (programmer)
//     and relating to physics, chemistry and mathematics, and in particular to spec-
//     troscopy

#include <tchar.h>
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
// #include <string.h>
#include <errno.h>
// #include <complex>
#include <math.h>
#include <windows.h>
#include <float.h>
#include <time.h>
#include <malloc.h>

#include "Define.h"
#include "Header.h"
#include "Spectr.h"

//using namespace std;

double CSpectralCalcs::sp_ps(){
    return ps=c/100*p; // [atm]
}

double CSpectralCalcs::sp_g_p(CDataBaseParameters DB){
    // returns the pressure broadened line halfwidth gamma(p,T) (in cm-1) for a gas at
    // pressure p [atm], temperature T [K], and mole fraction c [%] according to:
    // Formula (A12) in
    // the HITRAN 1996 article

    //return g_p=pow((Tref/T),DB.n_air)*DB.g_air*(p-
    ps)+pow((Tref/T),DB.n_self)*DB.g_self*ps; // [cm-1]
    return g_p=sp_g_p(DB.n_air,DB.g_air,DB.n_self,DB.g_self); // [cm-1]
}

double CSpectralCalcs::sp_g_p(double n_air, double g_air, double n_self, double g_self){
    // returns the pressure broadened line halfwidth gamma(p,T) (in cm-1) for a gas at
    // pressure p [atm], temperature T [K], and mole fraction c [%] according to:
    // Formula (A12) in
    // the HITRAN 1996 article
    return g_p=pow((Tref/T),n_air)*g_air*(p-ps)+pow((Tref_g_self/T),n_self)*g_self*ps;
// [cm-1]
//return g_p=pow((Tref/T),n_air)*g_air*(p-ps)+pow((Tref/T),n_self)*g_self*ps; //
[cm-1]
//Tref_g_self;
}

double CSpectralCalcs::sp_v0sh(CDataBaseParameters DB){
    return v0sh=DB.v0+DB.delta_air*p; // [cm-1]
}

```



```

}

double CSpectralCalcs::sp_v0sh(double v0, double delta_air){
    return v0sh=v0+delta_air*p; // [cm-1]
}

double CSpectralCalcs::sp_g_Dop(double v0sh, double T, double MolW){
    // returns the half-width (in cm-1) at half-height (hwhm) of the Doppler feature
    // according to:
    // Formula (3) in
    // Philip W. Morrison, Jr.* and Oranut Taweechokesupsin. Calculation of Gas Spec-
    // tra for Quantitative Fourier Transform
    // Infrared Spectroscopy of Chemical Vapor Deposition.
    // J Electrochem. Soc., Vol. 145, No. 9, September 1998

    return v0sh/con_c*sqrt(2.*con_Ru*T*Con.ln2/MolW); // [cm-1]
    //return g_Dop=v0sh/con_c*sqrt(2.*con_Ru*T*Con.ln2/Con.MolW.CDiox[Isot-1]); // [cm-
    // 1]
    //
    M.g_Dop=M.v0shifted/Const.c_Light*sqrt(2*Const.R_u*Spec.T.K*log(2)/Const.MW.CDiox.Isotop
    (1)); % [cm^{-1}]
}

double CSpectralCalcs::sp_g_Dop(size_t MolNum, size_t Isot){
    switch(MolNum){
        case conMolNum_CDiox:
            if(Isot>conNumOfIsotopes_CDiox || Isot<1)Isot=1;
            g_Dop=sp_g_Dop(v0sh,T,Con.MolW.CDiox[Isot-1]);
            break;
        case conMolNum_CMonox:
            if(Isot>conNumOfIsotopes_CMonox || Isot<1)Isot=1;
            g_Dop=sp_g_Dop(v0sh,T,Con.MolW.CMonox[Isot-1]);
            break;
        default:
            g_Dop=0;
    }
    return g_Dop;
}

double CSpectralCalcs::sp_g_Voi(CDataBaseParameters DB){
    // returns the half-width (in cm-1) of a Voigt line according to:
    // formula (15) in
    // Franz Schreier. Optimized implementations of rational approximations for the
    // Voigt and
    // complex error function.
    // Journal of Quantitative Spectroscopy & Radiative Transfer
    // 112(2011)1010-1025
    return g_Voi=0.5*(1.0692*g_p+sqrt(0.86639*g_p*g_p+4*g_Dop*g_Dop)); // [cm-1]
    // M.g_Voi=0.5*(1.0692*M.g_Lor+sqrt(0.86639*M.g_Lor.^2+4*M.g_Dop.^2));
}

double CSpectralCalcs::sp_g_Voi(){
    // returns the half-width (in cm-1) of a Voigt line according to:
    // formula (15) in
    // Franz Schreier. Optimized implementations of rational approximations for the
    // Voigt and
    // complex error function.
    // Journal of Quantitative Spectroscopy & Radiative Transfer
    // 112(2011)1010-1025
    return g_Voi=0.5*(1.0692*g_p+sqrt(0.86639*g_p*g_p+4*g_Dop*g_Dop)); // [cm-1]
    // M.g_Voi=0.5*(1.0692*M.g_Lor+sqrt(0.86639*M.g_Lor.^2+4*M.g_Dop.^2));
}

double CSpectralCalcs::sp_Trapz(double* X, double* Y, size_t N){
    double q=0;
    if(N>=2 && N<UINT_MAX){
        for (size_t ii=1; ii<N; ii++){
            //q+=(Y[ii-1]+Y[ii])/2*(X[ii]-X[ii-1]);
            q+=sp_Trapz(X[ii-1],Y[ii-1],X[ii],Y[ii]);
        }
    }
    return q;
}

```

```

}

double CSpectralCalcs::sp_Trapz(double x1, double y1, double x2, double y2){
    return (y2+y1)/2*(x2-x1);
}

BOOL CSpectralCalcs::sp_ILS_HalfX(CFile OutputFile){
    size_t ii;
    double
        R=ILSParm/2, // [cm-1]
        ILS_HalfX_MaxAllowed=XUser.Ax[0]/conMinWaveNumberCoeff;
    XModel.ILS_HalfX=conILSHalfXCoeffDefault*R;

    switch(ILSType.Num){
        //case 1:
        //    XModel.ILS_HalfX=conILSHalfXCoeffDefault*ILSParm/2;
        //    break;
        case defILSType_sinc:

            CAxis dv, ILS, dvend;

            double
                dvend_thresY=conILSWingCutoffCoeff*sp_ILS(0),
                nn;
            for(nn=1.5, ii=0;ii<UINT_MAX && abs(sp_ILS(R*nn))>=dvend_thresY;
                nn++, ii++){
                dvend.Start=R*(ceil(nn));
                dvend.Finish=dvend.Start+3*2*R;
                dvend.Step=XModel.Step/2;
                dvend.ActualLenClcCeil(dvend.Start,dvend.Step,dvend.Finish);
                //ILSInt.ActualLen=dvend.ActualLen;

                _ftprintf_s(OutputFile.Stream, _T("
                    L"\n\nILS Half X, case 2 (sinc ILS) - Details:\n"
                    L"\ndefMaxAlloc - dvend.ActualLen = %-d"
                    L"\nXModel.Step = %-.18g cm-1"
                    L"\ndvend.Start = %-.18g"
                    L"\ndvend.Finish = %-.18g"
                    L"\nnn = %-.18g"
                    L"\nILS(R*nn)/ILS(0) = %-.18g %%"
                    L"\n"
                    )
                    ,defMaxAlloc-dvend.ActualLen
                    ,XModel.Step
                    ,dvend.Start
                    ,dvend.Finish
                    ,nn
                    ,sp_ILS(R*nn)/sp_ILS(0)*100
                    );

                if(
                    !dvend.AllocMemFromActualLenAndMaxAlloc()
                    |||
                    //!ILSInt.AllocMemFromActualLenAndMaxAlloc()
                    ||
                    !dvend.AxCreate()
                    )
                    return FALSE;
                //for (ii=0;ii<dvend.ActualLen;ii++){
                //    dvend.Ax[ii]=dvend.Start+ii*dvend.Step;
                //}

                dv.ActualLenClcCeil(-1.*dvend.GetAxEnd(),XModel.Step,dvend.GetAxEnd());
                ILS.ActualLen=dv.ActualLen;
                if(!dv.AllocMemFromActualLenAndMaxAlloc()
                    ||
                    !ILS.AllocMemFromActualLenAndMaxAlloc())
                    return FALSE;

                double ILSInt, ILSIntHowCloseToUnity, ILSIntMinDifFromUnity;
                size_t ILSIntHowCloseToUnityii;
                ILSIntMinDifFromUnity=1;

```

```

    //_ftprintf_s(OutputFile.Stream, _T("
    // L"\n\nii dvend.Ax[ii] sp_Trapz(dv.Ax,ILS.Ax) HowCloseToUnity
    MinDifFromUnity corresp.ii XModel.ILS_HalfX\n\n"
    // ));

    for(ii=0;ii<dvend.ActualLen;ii++){
        dv.ActualLenClcFloor(-1.*dvend.Ax[ii],XModel.Step,dvend.Ax[ii]);
        ILS.ActualLen=dv.ActualLen;
        dv.AxCreate();
        sp_ILS(ILS.Ax,dv.Ax,dv.ActualLen);
        ILSIntHowCloseToUnity=fabs(sp_Trapz(dv.Ax,ILS.Ax,dv.ActualLen)-1.);
        if(ILSIntHowCloseToUnity<ILSIntMinDifFromUnity){
            ILSInt=sp_Trapz(dv.Ax,ILS.Ax,dv.ActualLen);
            ILSIntMinDifFromUnity=ILSIntHowCloseToUnity;
            ILSIntHowCloseToUnityii=ii;
            XModel.ILS_HalfX=dvend.Ax[ii];
        }
        //_ftprintf_s(OutputFile.Stream, _T("
        // L"%5u %14.7g %14.7g %14.7g %14.7g %5u %14.7g\n"
        // )
        // ,ii
        // ,dvend.Ax[ii]
        // ,sp_Trapz(dv.Ax,ILS.Ax,dv.ActualLen)
        // ,ILSIntHowCloseToUnity
        // ,ILSIntMinDifFromUnity
        // ,ILSIntHowCloseToUnityii
        // ,XModel.ILS_HalfX
        // );
    }

    _ftprintf_s(OutputFile.Stream, _T("
    L"\n\nILS Half X, case 2 (sinc ILS) - Details:\n"
    L"\nXModel.ILS_HalfX = %-.18g"
    L"\nILSIntCloseToUnity = %-.18g"
    L"\nILSIntMinDifFromUnity = %-.18g"
    L"\nILSIntCloseToUnityii = %-u"
    L"\nnn = %-.18g"
    L"\nILS(ILS_HalfX)/ILS(0) = %-.18E"
    L"\nconILSWingCutoffCoeff = %-.18E"
    L"\n"
    )
    ,XModel.ILS_HalfX
    ,ILSInt
    ,ILSIntMinDifFromUnity
    ,ILSIntHowCloseToUnityii
    ,nn
    ,sp_ILS(XModel.ILS_HalfX)/sp_ILS(0)
    ,conILSWingCutoffCoeff
    );

    _tprintf_s(_T("
    L"\n\nILS Half X, case 2 (sinc ILS) - Details:\n"
    L"\nXModel.ILS_HalfX = %-.18g"
    L"\nILSIntCloseToUnity = %-.18g"
    L"\nILSIntMinDifFromUnity = %-.18g"
    L"\nILSIntCloseToUnityii = %-u"
    L"\nnn = %-.18g"
    L"\nILS(ILS_HalfX)/ILS(0) = %-.18E"
    L"\nconILSWingCutoffCoeff = %-.18E"
    L"\n"
    )
    ,XModel.ILS_HalfX
    ,ILSInt
    ,ILSIntMinDifFromUnity
    ,ILSIntHowCloseToUnityii
    ,nn
    ,sp_ILS(XModel.ILS_HalfX)/sp_ILS(0)
    ,conILSWingCutoffCoeff
    );

    //XModel.ILS_HalfX=74.5322000998149*ILSParm/2;//74.5322000998149//65.5219116339485

```

```

if (dvend.IsMemoryAllocated) { free (dvend.Ax) ; dvend.IsMemoryAllocated=FALSE; }
if (dv.IsMemoryAllocated) { free (dv.Ax) ; dv.IsMemoryAllocated=FALSE; }
if (ILS.IsMemoryAllocated) { free (ILS.Ax) ; ILS.IsMemoryAllocated=FALSE; }

break;

//VERIFICATION by MATLAB "sp_ILS_HalfRange.m":
//R=ILS_Paramset(1,1); % [cm^-1]
//
//dv_end=10*R:X_Step/2:100*R;
//ILS_Int=zeros(size(dv_end));
//for k=1:length(dv_end)
//    dv=-dv_end(k):X_Step:dv_end(k);
//    if length(dv)>1
//
ILS_Int(k)=trapz(dv,sp_InstrumentLineShape(dv,0,ILS_Paramset,ILS_Type));
//    elseif length(dv)==1
//        ILS_Int(k)=0;
//    else
//        ILS_Int=zeros(0);
//        break;
//    end
//end
//[~,ILS_Int_1_Ind]=min(abs(ILS_Int-1));
//ILS_HalfRange=dv_end(ILS_Int_1_Ind); % [cm^-1]

//default:
}

XModel.ILS_HalfX=__min(XModel.ILS_HalfX,ILS_HalfX_MaxAllowed);

_ftprintf_s(OutputFile.Stream, _T("
L"\nILS Half X (final value used everywhere in the calculations):"
L"\nXModel.ILS_HalfX = %-.18g"
L"\nILS(ILS_HalfX)/ILS(0) = %-.18E"
L"\nconILSWingCutoffCoeff = %-.18E"
L"\n\n"
)
,XModel.ILS_HalfX
,sp_ILS(XModel.ILS_HalfX)/sp_ILS(0)
,conILSWingCutoffCoeff
);
_tprintf_s(_T("
L"\nILS Half X (final value used everywhere in the calculations):"
L"\nXModel.ILS_HalfX = %-.18g"
L"\nILS(ILS_HalfX)/ILS(0) = %-.18E"
L"\nconILSWingCutoffCoeff = %-.18E"
L"\n\n"
)
,XModel.ILS_HalfX
,sp_ILS(XModel.ILS_HalfX)/sp_ILS(0)
,conILSWingCutoffCoeff
);

return TRUE;
}

double CSpectralCalcs::sp_ILS(double v){
// The instrument line shape (ILS) function.
// Types:
// default - Triangular, the ILSParam is the full width at zero (or the base of
the isosceles triangle)
// defILSType_sinc - sinc, the ILSParam is the full width of the main maximum at
zero

double
ILS=0,
R=ILSParam/2,
ILS0=1/R;

switch(ILSType.Num){
case defILSType_sinc: // sinc, the ILSParam is the full width of the main maxi-
mum at zero

```

```

double x;
x=pi*v/R;
if (-DBL_EPSILON<=x && x<=DBL_EPSILON) {
    ILS=ILS0;
} else {
    ILS=ILS0*sin(x)/x;
}
break;
//VERIFICATION:
// R=paramset(1,1);
// x=pi*(vi-v0)/R;
// ILS=1/R*sin(x)./x;
// ILS(x==0)=1/R;

default:
if (-R<=v && v< 0) ILS=( v+R)*ILS0/R;
if ( 0<=v && v<=R) ILS=(-v+R)*ILS0/R;

//VERIFICATION:
// a=paramset(1,1);
// f_0_a=1/a;
//% ILS=0;
//% if vi>=v0-a && vi<v0, ILS=(vi-v0+a)*f_0_a/a; end
//% if vi>=v0 && vi<=v0+a, ILS=(v0+a-vi)*f_0_a/a; end
// ILS=zeros(size(vi));
// cond1=vi>=v0-a & vi<v0;
// ILS(cond1)=(vi(cond1)-v0+a)*f_0_a/a;
// cond2=vi>=v0 & vi<=v0+a;
// ILS(cond2)=(v0+a-vi(cond2))*f_0_a/a;
}
return ILS;
}

BOOL CSpectralCalcs::sp_ILS(double* ILS, double* v, size_t N){
for(size_t ii=0; ii<N; ii++){
    ILS[ii]=sp_ILS(v[ii]);
}
return TRUE;
}

BOOL CSpectralCalcs::sp_ILS(double* ILS, double* v, double v0, size_t N){
for(size_t ii=0; ii<N; ii++){
    ILS[ii]=sp_ILS(v[ii]-v0);
}
return TRUE;
}

BOOL CSpectralCalcs::sp_ConvolutionWithILS(CFile OutputFile){
//
_tprintf_s(_T("
L"\n\nConvolution of the true transmissivity of the CO2+CO mixture with the
ILS function is in progress...\n"));

Tau.Conv.ActualLen=XUser.ActualLen;
Absorbance.Conv.ActualLen=XUser.ActualLen;
if(
!Tau.Conv.AllocMemFromActualLenAndMaxAlloc()
||
!Absorbance.Conv.AllocMemFromActualLenAndMaxAlloc()
){
    _ftprintf_s(OutputFile.Stream, _T("
L"\n\nERROR:\nWhile calculating the convolution with the instrument line
shape function:\n"
L"Failed to allocate memory for some variables. The application will
terminate.\n\n"
));
    _tprintf_s(_T("
L"\n\nERROR:\nWhile calculating the convolution with the instrument line
shape function:\n"
L"Failed to allocate memory for some variables. The application will
terminate.\n\n"
));
return FALSE;
}
}

```

```

}

LONG ii, jj;
for(ii=0;ii<(LONG)XUser.ActualLen && ii<LONG_MAX;ii++){
    Tau.Conv.Ax[ii]=0;
    XModel.iiLeftClc(XUser.Ax[ii]-XModel.ILS_HalfX);
    XModel.iiRightClc(XUser.Ax[ii]+XModel.ILS_HalfX);
    for(jj=XModel.iiLeft+1;jj<=XModel.iiRight && jj<LONG_MAX;jj++){
        Tau.Conv.Ax[ii]+=sp_Trapz(
            XModel.Ax[jj-1],
            Tau.True.Ax[jj-1]*sp_ILS(XModel.Ax[jj-1]-XUser.Ax[ii]),
            XModel.Ax[jj ],
            Tau.True.Ax[jj ]*sp_ILS(XModel.Ax[jj ]-XUser.Ax[ii])
        );
    }
    Absorbance.Conv.Ax[ii]=Absorbance.ta_Absorbance(Tau.Conv.Ax[ii]);
}

//for(ii=0; ii<UINT_MAX && ii<SpecClc.XUser.ActualLen; ii++){
// //Tau.True.Ax[ii]=Tau.ta_TauTrue(kSum.Ax[ii],n_molec_per_cm3,path);
//
// SpecClc.Absorbance.Conv.Ax[ii]=SpecClc.Absorbance.ta_Absorbance(SpecClc.Tau.Conv.Ax[
ii]);
// _ftprintf_s(OutputFile.Stream, _T("%#28.18g %#28.18g %#28.18g\n"),
// SpecClc.XUser.Ax[ii], SpecClc.Tau.Conv.Ax[ii],
SpecClc.Absorbance.Conv.Ax[ii]);
//}

_tprintf_s(_T("
L" Done.\n\n"));

return TRUE;

//VERIFICATION:
//if nargin==5
// X_Step_Vec=zeros(1,length(Modelling_X)-1);
// for ii=2:length(Modelling_X)
// X_Step_Vec(ii-1)=Modelling_X(ii)-Modelling_X(ii-1);
// end
// X_Step=mean(X_Step_Vec);
//end
//
//Conv_Span=sp_ILS_HalfRange(ILS_Paramset,ILS_Type,X_Step); % [cm^-1]
//convolved=zeros(size(Conv_X));
//for k=1:length(Conv_X)
// Conv_Border=[Conv_X(k)-Conv_Span Conv_X(k)+Conv_Span]; % [cm^-1]
// Conv_Indices=sp_IndexFromTheValue(Modelling_X,Conv_Border(1)):...
// sp_IndexFromTheValue(Modelling_X,Conv_Border(2)); % [indices]
// Current_Modelling_X=Modelling_X(Conv_Indices); % [cm^-1]
// convolved(k)=trapz(Current_Modelling_X,...
// to_convolve(Conv_Indices).*...
//
sp_InstrumentLineShape(Current_Modelling_X,Conv_X(k),ILS_Paramset,ILS_Type));
//end
}

CAxis::CAxis() {
    Ax=NULL;
    IsMemoryAllocated=FALSE;
    ActualLen=0;//MaxAlloc;

    //_tprintf_s(_T("
// L"\n\nSystem message:\nConstructor\nConstructor CAxis() has been execut-
ed.\nConstructor\n\n"
// )
// );
}

size_t CAxis::ActualLenClcCeil(){
    return ActualLen=(size_t)ceil((Finish-Start)/(Step)+1);
}

```

```

size_t CAxis::ActualLenClcCeil(double param_start, double param_step, double
param_finish){
    Start=param_start;
    Step=param_step;
    Finish=param_finish;
    return ActualLenClcCeil();
}

size_t CAxis::ActualLenClcFloor(){
    return ActualLen=(size_t)floor((Finish-Start)/(Step)+1);
}

size_t CAxis::ActualLenClcFloor(double param_start, double param_step, double
param_finish){
    Start=param_start;
    Step=param_step;
    Finish=param_finish;
    return ActualLenClcFloor();
}

BOOL CAxis::AxCreate(){
    for (size_t ii=0;ii<ActualLen && ii<MaxAlloc;ii++){
        Ax[ii]=Start+ii*Step;
    }
    return TRUE;
}

double CAxis::GetAxEnd(){
    if(ActualLen>=1 && ActualLen<=MaxAlloc){
        return Ax[ActualLen-1];
    }else{
        return 0;
    }
}

BOOL CAxis::AllocateMemory(size_t AllocSize){
    //Ax = new double [AllocSize];
    Ax = (double *)calloc( AllocSize, sizeof( double ) );
    if(Ax==NULL){//0
        IsMemoryAllocated=FALSE;
        return FALSE;
    }else{
        IsMemoryAllocated=TRUE;
        return TRUE;
    }
    //long *buffer;

    // buffer = (long *)calloc( 40, sizeof( long ) );
    // if( buffer != NULL )
    //     printf( "Allocated 40 long integers\n" );
    // else
    //     printf( "Can't allocate memory\n" );
    // free( buffer );

    // \Another EXAMPLE from Help:
    //ms-help://MS.VSCC.v90/MS.MSDNQTR.v90.en/dv_vclang/html/afa859b0-f3bc-4c4d-b250-
    c68b335b6004.htm
    // class String {
    //public:
    //     String( char *ch ); // Declare constructor
    //     ~String(); // and destructor.
    //private:
    //     char *_text;
    //     size_t sizeOfText;
    //};
    //
    ///// Define the constructor.
    //String::String( char *ch ) {
    //     sizeOfText = strlen( ch ) + 1;
    //
    //     // Dynamically allocate the correct amount of memory.
    //     _text = new char[ sizeOfText ];
    //
    //

```

```

// // If the allocation succeeds, copy the initialization string.
// if( _text )
//     strcpy_s( _text, sizeofText, ch );
//}
//
////// Define the destructor.
//String::~String() {
//    // Deallocate the memory that was previously reserved
//    // for this string.
//    if ( _text )
//        delete[] _text;
//}
//
//int main() {
//    String str("The piper in the glen...");
//}

BOOL CAxis::AllocMemFromActualLenAndMaxAlloc() {
    if(ActualLen<=MaxAlloc && AllocateMemory(ActualLen))
        return TRUE;
    else
        return FALSE;
}

CAxis::~CAxis(){
    if (IsTimeToDestruct && IsMemoryAllocated){
        //delete [] Ax;
        free( Ax );
        IsMemoryAllocated=FALSE;
        _tprintf_s(_T("
L"\n\nSystem message:\nDestructor ~CAxis() has been executed within
IsTimeToDestruct==TRUE && IsMemoryAllocated==TRUE.\nDestructor\n"
)
);
    }
}

double CXAxisModel::cm1_2_ii(double v){
    //XModel.Ax[ii]=XModel.Start+ii*XModel.Step;
    return (v-Start)/Step;
}

LONG CXAxisModel::iiLeftClc(double v){
    return iiLeft=(LONG)(__max((double)(0.),floor(cm1_2_ii(v))));
}

LONG CXAxisModel::iiRightClc(double v){
    return iiRight=(LONG)(__min((double)(ActualLen-1),ceil(cm1_2_ii(v))));
}

BOOL CSpectralCalcs::XModelCreate(CFile OutputFile){
    XModel.FactorForMinGamma=conFactorForMinGamma;
    XModel.Step=fabs(g_Voi_min/XModel.FactorForMinGamma);
    XModel.Step=__max(conXModelMinStep,XModel.Step);

    if(!sp_ILS_HalfX(OutputFile)){
        _ftprintf_s(OutputFile.Stream,_T("
L"\n\nERROR:\nThe wave number axis for the TRUE transmissivity has NOT been
created.\n"
L"Failed to allocate memory for some variables. The application will termi-
nate.\n\n"
));
        _tprintf_s(_T("
L"\n\nERROR:\nThe wave number axis for the TRUE transmissivity has NOT been
created.\n"
L"Failed to allocate memory for some variables. The application will termi-
nate.\n\n"
));
        return FALSE;
    }

    XModel.XUser2ModelExt=conXUser2ModelExtCoeff*XModel.ILS_HalfX;
}

```



```

XModel.Start=XUser.Range[0]-XModel.XUser2ModelExt;
XModel.Finish=XUser.Range[1]+XModel.XUser2ModelExt;

XModel.ActualLen=(size_t) floor( fabs( (XModel.Finish-XModel.Start) ) / XModel.Step + 1 );

double CurrentValue=XModel.Start;
register size_t ii=0;
// Counting the number of elements necessary for the modelling X axis
//for( ; CurrentValue<=XModel.Finish && ii<UINT_MAX; ii++,
CurrentValue+=XModel.Step);

//XModel.ActualLen=ii;

kSum.ActualLen=XModel.ActualLen;
Tau.True.ActualLen=XModel.ActualLen;
Absorbance.True.ActualLen=XModel.ActualLen;

if(
  XModel.AllocMemFromActualLenAndMaxAlloc()
  &&
  kSum.AllocMemFromActualLenAndMaxAlloc()
  &&
  Tau.True.AllocMemFromActualLenAndMaxAlloc()
  &&
  Absorbance.True.AllocMemFromActualLenAndMaxAlloc()
){
  for(ii=0;ii<XModel.ActualLen;ii++){
    XModel.Ax[ii]=XModel.Start+ii*XModel.Step;
    kSum.Ax[ii]=0;
    Tau.True.Ax[ii]=0;
    Absorbance.True.Ax[ii]=0;
  }

  _ftprintf_s(OutputFile.Stream, _T("
ed.\n"
    L"\nThe wave number axis for the TRUE transmissivity has been creat-
    L"    First:  %#28.18g cm-1 \n"
    L"    Step:   %#28.18g cm-1 ( XModel.Step )\n"
    L"           %#28.18g cm-1 ( XModel.Ax[1]-XModel.Ax[0] )\n"
    L"    Last:   %#28.18g cm-1 ( XModel.GetAxEnd() )\n"
    L"           %#28.18g cm-1 ( XModel.Finish )\n"
    L"    Number of elements: %-u\n"
    L"           %-u\n"
    )
    ,XModel.Ax[0]
  ,XModel.Step
    ,XModel.Ax[1]-XModel.Ax[0]
  ,XModel.GetAxEnd()
    ,XModel.Finish
    ,XModel.ActualLen
    ,kSum.ActualLen
  );
  _tprintf_s(_T("
ed.\n"
    L"\nThe wave number axis for the TRUE transmissivity has been creat-
    L"    First:  %#28.18g cm-1 \n"
    L"    Step:   %#28.18g cm-1 ( XModel.Step )\n"
    L"           %#28.18g cm-1 ( XModel.Ax[1]-XModel.Ax[0] )\n"
    L"    Last:   %#28.18g cm-1 ( XModel.GetAxEnd() )\n"
    L"           %#28.18g cm-1 ( XModel.Finish )\n"
    L"    Number of elements: %-u\n"
    )
    ,XModel.Ax[0]
  ,XModel.Step
    ,XModel.Ax[1]-XModel.Ax[0]
  ,XModel.GetAxEnd()
    ,XModel.Finish
    ,XModel.ActualLen
  );

  return TRUE;
}else{

```

```

        _ftprintf_s(OutputFile.Stream,_T("
        //L"The range of values in the wave number file is too wide! Try to use a
more narrow range of the wave numbers."
        L"\n\nERROR:\nThe wave number axis for the TRUE transmissivity has NOT been
created.\n"
        L"    The range of values in the wave number file is so wide that\n"
        L"    the TRUE transmissivity wave number axis has to have %-d elements
which is too many. Try to specify a more narrow spectral range.\n\n"
        )
        ,XModel.ActualLen);
        _tprintf_s(_T("
        //L"The range of values in the wave number file is too wide! Try to use a
more narrow range of the wave numbers."
        L"\n\nERROR:\nThe wave number axis for the TRUE transmissivity has NOT been
created.\n"
        L"The range of values in the wave number file is so wide that "
        L"the TRUE transmissivity wave number axis has to have %-d elements which is
too many. Try to specify a more narrow spectral range.\n\n"
        )
        ,XModel.ActualLen);
        return FALSE;
    }
}

double CSpectralCalcs::sp_v_minus_v0sh(double v){
    return v_minus_v0sh=v-v0sh;
}

double CSpectralCalcs::sp_Lor_Common(double v_v0shifted, double gamma_p_T){
    // returns a Lorentz profile:
    // formula (A14) in
    // the HITRAN 1996 Article

    return 1/pi*gamma_p_T/(gamma_p_T*gamma_p_T+v_v0shifted*v_v0shifted);
}

double CSpectralCalcs::sp_Lor_Common(){
    // returns a Lorentz profile:
    // formula (A14) in
    // the HITRAN 1996 Article

    return 1/pi*g_p/(g_p*g_p+v_minus_v0sh*v_minus_v0sh);
}

double CSpectralCalcs::sp_VoiWhi(double v_v0shifted,double gamma_p, double gamma_V){
    // returns a Voigt profile according to Whiting's approximation:
    // formulae (6) and (7) in
    // Philip W. Morrison, Jr.* and Oranut Taweekhesupsin.
    // Calculation of Gas Spectra for Quantitative Fourier Transform
    // Infrared Spectroscopy of Chemical Vapor Deposition.
    // J Electrochem. Soc., Vol. 145, No. 9, September 1998

    double
        gP_gV=gamma_p/gamma_V,
        v_v0_2gV=fabs(v_v0shifted)/2/gamma_V;
    return
        ((1-gP_gV)*
        exp(-2.772*v_v0_2gV*v_v0_2gV)+
        (gP_gV)/(1+4*v_v0_2gV*v_v0_2gV)+0.016*(1-gP_gV)*gP_gV*
        (exp(-0.4*pow(v_v0_2gV,2.25))-10/(10+pow(v_v0_2gV,2.25))))*
        1/(2*gamma_V*(1.065+0.447*gP_gV+0.058*gP_gV*gP_gV));
    // VERIFICATION:
    // gP_gV=gamma_p/gamma_V;
    //v_v0_2gV=v_v0shifted/2/gamma_V;
    //profile=((1-gP_gV)...
    // *exp(-2.772*v_v0_2gV.^2)...
    // +(gP_gV)*1./(1+4*v_v0_2gV.^2)+0.016*(1-gP_gV)*gP_gV...
    // *(exp(-0.4*v_v0_2gV.^2.25)-10./(10+v_v0_2gV.^2.25))...
    // *1/(2*gamma_V*(1.065+0.447*gP_gV+0.058*gP_gV^2));
}

double CSpectralCalcs::sp_VoiWhi(double v_v0sh){
    // returns a Voigt profile according to Whiting's approximation:

```

```

// formulae (6) and (7) in
// Philip W. Morrison, Jr.* and Oranut Taweechokesupsin.
// Calculation of Gas Spectra for Quantitative Fourier Transform
// Infrared Spectroscopy of Chemical Vapor Deposition.
// J Electrochem. Soc., Vol. 145, No. 9, September 1998

//return sp_VoiWhi(v_v0sh,g_p,g_Voi); // Doing so doesn't produce .exe working fast-
est
// The below approach seem to produce .exe working fastest

double
  gP_gV=g_p/g_Voi,
  v_v0_2gV=fabs(v_v0sh)/2/g_Voi;

return
  ((1-gP_gV)*
  exp(-2.772*v_v0_2gV*v_v0_2gV)+
  (gP_gV)/(1+4*v_v0_2gV*v_v0_2gV)+0.016*(1-gP_gV)*gP_gV*
  (exp(-0.4*pow(v_v0_2gV,2.25))-10/(10+pow(v_v0_2gV,2.25))))*
  1/(2*g_Voi*(1.065+0.447*gP_gV+0.058*gP_gV*gP_gV));
}

double CSpectralCalcs::sp_VoiWhi(){
// returns a Voigt profile according to Whiting's approximation:
// formulae (6) and (7) in
// Philip W. Morrison, Jr.* and Oranut Taweechokesupsin.
// Calculation of Gas Spectra for Quantitative Fourier Transform
// Infrared Spectroscopy of Chemical Vapor Deposition.
// J Electrochem. Soc., Vol. 145, No. 9, September 1998

return sp_VoiWhi(v_minus_v0sh,g_p,g_Voi);

//double
// gP_gV=g_p/g_Voi,
// v_v0_2gV=fabs(v_minus_v0sh)/2/g_Voi;

//return
// ((1-gP_gV)*
// exp(-2.772*v_v0_2gV*v_v0_2gV)+
// (gP_gV)/(1+4*v_v0_2gV*v_v0_2gV)+0.016*(1-gP_gV)*gP_gV*
// (exp(-0.4*pow(v_v0_2gV,2.25))-10/(10+pow(v_v0_2gV,2.25))))*
// 1/(2*g_Voi*(1.065+0.447*gP_gV+0.058*gP_gV*gP_gV));

//VERIFICATION 1:
//double
// gP_gV=gamma_p/gamma_V,
// v_v0_2gV=v_v0shifted/2/gamma_V;
//return
// ((1-gP_gV)*
// exp(-2.772*v_v0_2gV*v_v0_2gV)+
// (gP_gV)/(1+4*v_v0_2gV*v_v0_2gV)+0.016*(1-gP_gV)*gP_gV*
// (exp(-0.4*pow(v_v0_2gV,2.25))-10/(10+pow(v_v0_2gV,2.25))))*
// 1/(2*gamma_V*(1.065+0.447*gP_gV+0.058*gP_gV*gP_gV));

// VERIFICATION 2 (MATLAB):
// gP_gV=gamma_p/gamma_V;
//v_v0_2gV=v_v0shifted/2/gamma_V;
//profile=((1-gP_gV)...
// *exp(-2.772*v_v0_2gV.^2)...
// +(gP_gV)*1./(1+4*v_v0_2gV.^2)+0.016*(1-gP_gV)*gP_gV...
// *(exp(-0.4*v_v0_2gV.^2.25)-10./(10+v_v0_2gV.^2.25)))...
// *1/(2*gamma_V*(1.065+0.447*gP_gV+0.058*gP_gV.^2));
}

double CSpectralCalcs::sp_VoiWhi_WithoutAssignment(){
// returns a Voigt profile according to Whiting's approximation:
// formulae (6) and (7) in
// Philip W. Morrison, Jr.* and Oranut Taweechokesupsin.
// Calculation of Gas Spectra for Quantitative Fourier Transform
// Infrared Spectroscopy of Chemical Vapor Deposition.
// J Electrochem. Soc., Vol. 145, No. 9, September 1998

//double

```

```

// gP_gV=(g_p/g_Voi),
// v_v0_2gV=(v_minus_v0sh/2/g_Voi);

return
((1-(g_p/g_Voi))*
exp(-2.772*(fabs(v_minus_v0sh)/2/g_Voi)*(fabs(v_minus_v0sh)/2/g_Voi))+

((g_p/g_Voi)/(1+4*(fabs(v_minus_v0sh)/2/g_Voi)*(fabs(v_minus_v0sh)/2/g_Voi))+0.016*
(1-(g_p/g_Voi))*(g_p/g_Voi)*
(exp(-0.4*pow((fabs(v_minus_v0sh)/2/g_Voi),2.25))-
10/(10+pow((fabs(v_minus_v0sh)/2/g_Voi),2.25)))))*
1/(2*g_Voi*(1.065+0.447*(g_p/g_Voi)+0.058*(g_p/g_Voi)*(g_p/g_Voi)));
}

double CSpectralCalcs::sp_VoiWhi_WithoutAssignmentAndPow(double v_v0sh){
// returns a Voigt profile according to Whiting's approximation:
// formulae (6) and (7) in
// Philip W. Morrison, Jr.* and Oranut Taweechokesupsin.
// Calculation of Gas Spectra for Quantitative Fourier Transform
// Infrared Spectroscopy of Chemical Vapor Deposition.
// J Electrochem. Soc., Vol. 145, No. 9, September 1998

//double
// gP_gV=(g_p/g_Voi),
// v_v0_2gV=(v_minus_v0sh/2/g_Voi);

//pow((fabs(v_minus_v0sh)/2/g_Voi),2.25)=pow((v_v0sh/2/g_Voi),2.25)=((v_v0sh/2/g_Voi
)*(v_v0sh/2/g_Voi)*sqrt(sqrt(fabs(v_v0sh)/2/g_Voi)))
return
((1-(g_p/g_Voi))*
exp(-2.772*((v_v0sh)/2/g_Voi)*((v_v0sh)/2/g_Voi))+
((g_p/g_Voi)/(1+4*((v_v0sh)/2/g_Voi)*((v_v0sh)/2/g_Voi))+0.016*(1-
(g_p/g_Voi))*(g_p/g_Voi)*
(exp(-0.4*((v_v0sh)/2/g_Voi)*((v_v0sh)/2/g_Voi)*sqrt(sqrt(fabs(v_v0sh)/2/g_Voi))))-
10/(10+((v_v0sh)/2/g_Voi)*((v_v0sh)/2/g_Voi)*sqrt(sqrt(fabs(v_v0sh)/2/g_Voi)))))*
1/(2*g_Voi*(1.065+0.447*(g_p/g_Voi)+0.058*(g_p/g_Voi)*(g_p/g_Voi)));
}

BOOL CSpectralCalcs::LineProfileWingCutoff(CFile OutputFile){
double RefProfVal=conLineProfileWingCutoffCoeff*sp_VoiWhi(0,g_p_max,g_Voi_max); //
[unitless]
ULONG ii;
XModel.ProfileXThres=XModel.Step; // [cm-1]
for(ii=1;ii<ULONG_MAX &&
sp_VoiWhi(XModel.ProfileXThres,g_p_max,g_Voi_max)>=RefProfVal;
ii++, XModel.ProfileXThres+=XModel.Step);

XModel.ProfileXThres=__min(XModel.ProfileXThres,conMaxLProfWingCutoffValuecm1);

_ftprintf_s(OutputFile.Stream, _T(
"\nLine profile wing cutoff has been calculated.\n"
L" Number of steps taken: %-d\n"
L" Value: %-.18g cm-1\n"
L" It means that line profiles are calculated only at the wave numbers from -
Value to +Value only.\n"
L" Profile(Value,g_p_max,g_Voi_max)/Profile(0,g_p_max,g_Voi_max) = %-.18E\n"
L" conLineProfileWingCutoffCoeff = %-.18E\n"
L"\nRecomended minimum wave number range for the database files:\n"
L" [%-.6g %-.6g] cm-1\n\n"
),ii
,XModel.ProfileXThres
,(sp_VoiWhi(XModel.ProfileXThres,g_p_max,g_Voi_max)/sp_VoiWhi(0,g_p_max,g_Voi_max))
,conLineProfileWingCutoffCoeff
,XUser.Range[0]-XModel.ILS_HalfX-
XModel.ProfileXThres,XUser.Range[1]+XModel.ILS_HalfX+XModel.ProfileXThres
);

_tprintf_s(_T(
"\nLine profile wing cutoff has been calculated.\n"
L" Number of steps taken: %-d\n"
L" Value: %-.18g cm-1\n"

```

```

        L"    It means that line profiles are calculated only at the wave numbers from -
Value to +Value only.\n"
        L"    Profile(Value,g_p_max,g_Voi_max)/Profile(0,g_p_max,g_Voi_max) = %-.18E\n"
        L"    conLineProfileWingCutoffCoeff = %-.18E\n"
        L"\nRecommended minimum wave number range for the database files:\n"
        L"    [%-.6g %-.6g] cm-1\n\n"
    )
    ,ii
    ,XModel.ProfileXThres

    , (sp_VoiWhi(XModel.ProfileXThres,g_p_max,g_Voi_max)/sp_VoiWhi(0,g_p_max,g_Voi_max))
    ,conLineProfileWingCutoffCoeff
    ,XUser.Range[0]-XModel.ILS_HalfX-
XModel.ProfileXThres,XUser.Range[1]+XModel.ILS_HalfX+XModel.ProfileXThres
    );
//
    return TRUE;
}

void CMolecularWeight::CalcMolW() {
    defDeclar_MolW_CDiox; // [kg/mol]
    for (USHORT ii=0;ii<conNumOfIsotopes_CDiox;ii++)
        CDiox[ii]=MolW_CDiox[ii]; // [kg/mol]
    defDeclar_MolW_CMonox; // [kg/mol]
    for (USHORT ii=0;ii<conNumOfIsotopes_CMonox;ii++)
        CMonox[ii]=MolW_CMonox[ii]; // [kg/mol]
}

void CConstants::CalcCon() {
    MolW.CalcMolW();

    ln2=log(2.);
    sqrt_ln2=sqrt(ln2);
    sqrt_ln2bypi=sqrt(ln2/pi);
}

CFileProcessing::CFileProcessing() {
    BinaryDBFileProcAgain=WNFileProcAgain=0;
    FileLine=0;
    Cntr.Initialize();
}

CCMonoxFileProcessing::CCMonoxFileProcessing() {
    BinaryDBFileProcAgain=WNFileProcAgain=0;
    FileLine=0;
    Cntr.Initialize();
}

void CFileProcessing::DBFileGetFormat(CSpectralCalcs* SpecClc) {
    // Calculating the starting and ending element number of the fields
    DataBField[0].start=1;
    for (USHORT k=0; k<conDBNumOfParams; k++) {
        DataBField[k].len=DataBaseFieldLen[k];
        if (k>0) DataBField[k].start=DataBField[k-1].end+1;
        DataBField[k].end=DataBField[k].start+DataBField[k].len-1;
    }

    switch (SpecClc->DataBaseType.Num) {
        case defDBType_HITEMP2010:
            case defDBType_CDSDOnek:
                DBNumOfParamsTrimmed=conDBNumOfParams_Standard;
                break;
        default:
            DBNumOfParamsTrimmed=conDBNumOfParams;
    }
}

BOOL CFileProcessing::TextDataBaseFileProcessing(CSpectralCalcs* SpecClc) {
    register ULONG k=0, m=0, n=0;
    //size_t FWriteNumOfElems;

    PrintOpeningSeparatorToFileAndStdout (OutputFile.Stream);
    switch (SpecClc->MolNum) {

```

```

case conMolNum_CDiox:
    _ftprintf_s(OutputFile.Stream, _T("
        L"\n%-s (%-s) DATABASE FILES\n"
        L"\n"
        L"Folder with the database files:\n\"%-s\"\n"
        L"Spectral range for the database files: from %-u to %-u.\n\n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TextDBFile.Path.tBuf
        ,TextDBFile.WaveNumStart, TextDBFile.WaveNumStartStop
        );
    _tprintf_s(_T("
        L"\n%-s (%-s) DATABASE FILES\n"
        L"\n"
        L"Folder with the database files:\n\"%-s\"\n"
        L"Spectral range for the database files: from %-u to %-u.\n\n"
        L"Searching for database files in the binary format...\n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TextDBFile.Path.tBuf
        ,TextDBFile.WaveNumStart, TextDBFile.WaveNumStartStop
        );
break;
case conMolNum_CMonox:
    _ftprintf_s(OutputFile.Stream, _T("
        L"\n%-s (%-s) DATABASE FILE:\n"
        L"\n"
        L"\"%-s\\%-s\"\n\n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TextDBFile.Path.tBuf
        ,defDBFileName_CMonox
        );
    _tprintf_s(_T("
        L"\n%-s (%-s) DATABASE FILE:\n"
        L"\n"
        L"\"%-s\\%-s\"\n\n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TextDBFile.Path.tBuf
        ,defDBFileName_CMonox
        );
break;
default:
    _ftprintf_s(OutputFile.Stream, _T("
        L"\n\n\nERROR:\n\n"
        L"Internal error occured before processing the text data base files:
        MolNum not recognized. The application will terminate\n"
        )
        );
    _tprintf_s(_T("
        L"\n\n\nERROR:\n\n"
        L"Internal error occured before processing the text data base files:
        MolNum not recognized. The application will terminate\n"
        )
        );
return FALSE;
}

ULONG WNStart, WNStop, MainFORIterationNumber;
MainFORIterationNumber=1;
switch(SpecClc->MolNum){
case conMolNum_CDiox:
    WNStart=TextDBFile.WaveNumStart;
    WNStop=WNStart+1;
break;
case conMolNum_CMonox:
    WNStart=WNStop=defWNStopMax-1;
break;
}

```

```

default:
    _ftprintf_s(OutputFile.Stream, _T("
        L"\n\n\nERROR:\n\n"
        L"Internal error ocurred before processing the text data base files:
MolNum not recognized. The application will terminate\n"
    )
    );
    _tprintf_s(_T("
        L"\n\n\nERROR:\n\n"
        L"Internal error ocurred before processing the text data base files:
MolNum not recognized. The application will terminate\n"
    )
    );
    return FALSE;
}

for(
;
WNStop<defWNStopMax && MainFORIterationNumber<ULONG_MAX;
WNStop++, MainFORIterationNumber++
){
    if(BinaryDBFile.OpenFileForReading(WNStart, WNStop, SpecClc)){
        if(!BinaryDBFile.CloseFile()) return FALSE;
        _tprintf_s(_T("
            L"\nOne %-s (%-s) database file in the binary format has been
found: \n\"%-s\"\n"
            L"It is assumed that all the subsequent files also exist in the
binary format "
            L"so the conversion from the text format to binary is not need-
ed.\n\n"
        )
        , SpecClc->MolNameShort.tBuf
        , SpecClc->MolNameLong.tBuf
        , BinaryDBFile.FileNameCurrent.tBuf);
        return TRUE;
    }
}
//_tprintf_s(_T("
// L"\n\nBinary database file full name: \n\"%-s\"\n\n"
// )
// , BinaryDBFile.FullNameCurrent.tBuf);

switch(SpecClc->MolNum) {
case conMolNum_CDiox:
    _tprintf_s(_T("
        L"There were no %-s (%-s) databse files found in the binary format.\n"
        L"The database files will be converted from the text format to bina-
ry.\n\n"
        L"Database files with a starting wave number in the range from %-u to %-
u are being converted to the binary format...\n\n"
        L"Progress...\nDatabase file and the number of transitions in it:\n"
    )
    , SpecClc->MolNameShort.tBuf
    , SpecClc->MolNameLong.tBuf
    , TextDBFile.WaveNumStart, TextDBFile.WaveNumStartStop
    );
    _ftprintf_s(OutputFile.Stream, _T("
        L"%-s (%-s) database files with a starting wave number in the range from
%-u to %-u have been converted to the binary format.\n\n"
        L"Summary.\nDatabase file and the number of transitions in it:\n"
    )
    , SpecClc->MolNameShort.tBuf
    , SpecClc->MolNameLong.tBuf
    , TextDBFile.WaveNumStart, TextDBFile.WaveNumStartStop
    );
    break;
case conMolNum_CMonox:
    _tprintf_s(_T("
        L"The %-s (%-s) databse file was not found in the binary format.\n"
        L"It will be converted from the text format to binary.\n\n"
        L"the database file is being converted to the binary format...\n\n"
        L"Progress...\nThe database file and the number of transitions in it:\n"
    )
    );
}

```

```

        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
    );
    _ftprintf_s(OutputFile.Stream, _T("
L"%-s (%-s) database file have been converted to the binary format.\n\n"
L"Summary.\nThe database file and the number of transitions in it:\n"
)
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
    );
    break;
default:
    _ftprintf_s(OutputFile.Stream, _T("
L"\n\n\nERROR:\n\n"
L"Internal error ocured before processing the text data base files:
MolNum not recognized. The application will terminate\n"
)
    );
    _tprintf_s(_T("
L"\n\n\nERROR:\n\n"
L"Internal error ocured before processing the text data base files:
MolNum not recognized. The application will terminate\n"
)
    );
    return FALSE;
}

DBFileGetFormat(SpecClc);
Cntr.Initialize();

Cntr.TransTotal=
Cntr.TransTotalInUse=
Cntr.Files=0;

MainFORIterationNumber=1;
switch(SpecClc->MolNum){
case conMolNum_CDiox:
    WNStart=TextDBFile.WaveNumStart;
    WNStop=WNStart+1;
    break;
case conMolNum_CMonox:
    WNStart=TextDBFile.WaveNumStartStop;
    //WNStart=WNStop;
    WNStop=WNStart+1;
    //WNStop=WNStart+1;
    break;
default:
    _ftprintf_s(OutputFile.Stream, _T("
L"\n\n\nERROR:\n\n"
L"Internal error ocured before processing the text data base files:
MolNum not recognized.\n"
)
    );
    _tprintf_s(_T("
L"\n\n\nERROR:\n\n"
L"Internal error ocured before processing the text data base files:
MolNum not recognized. The application will terminate\n"
)
    );
    return FALSE;
}

for(
;
    WNStart<=TextDBFile.WaveNumStartStop && MainFORIterationNumber<ULONG_MAX;
    MainFORIterationNumber++
){
    // Opening the current database file:
    if(TextDBFile.OpenCurrentFile(WNStart, WNStop, SpecClc)){
        Cntr.Files++;
        //if (!BinaryDBFileProcAgain && Cntr.Files==1)

        if(!BinaryDBFile.CreateFileForWriting(TextDBFile)) return FALSE;
    }
}

```



```

        _ftprintf_s(OutputFile.Stream, _T("\n%-u.  \\"%-s\\"      "), Cntr.Files,
TextDBFile.FileNameCurrent.tBuf);
        _tprintf_s(
TextDBFile.FileNameCurrent.tBuf);

        //Processing the database file:

        //Run over the lines:
        for ( FileLine=Cntr.FileTrans=Cntr.FileTransInUse=0;
!feof(TextDBFile.Stream) && FileLine<ULONG_MAX; ){
            FileLine++;
            Cntr.FileTrans++;
            Cntr.TransTotal++;
            //Reading the line and writing it into a string:
            for(m=0, n=0; (tch=_gettc(TextDBFile.Stream))!=WEOF && tch!='\n' &&
m<ULONG_MAX; ){
                m++;
                if(n<conDataBaseLineLen-3-conDBNumOfParams){ // defBufLarge
                    DataBStrWithSpaces[n]=tch;
                    for(k=0; k<conDBNumOfParams; k++){
                        if(m==DataBField[k].end){
                            DataBStrWithSpaces[+n]=' ';
                            break;
                        }
                    }
                    n++;
                }

                DataBStrWithSpaces[n]=0;
                // _ftprintf_s( OutputFile.Stream, _T("\n%-s\n"),
DataBStrWithSpaces );

                if(m==0 && (tch=='\n' || tch==WEOF)){
                    Cntr.FileTrans--;
                    Cntr.TransTotal--;
                    continue;
                }
                if(m==ULONG_MAX && tch!='\n' && tch!=WEOF){
                    DBFileError_InvalidFormat();
                    return FALSE;
                }

                //Extracting the spectroscopic parameter values from the string:
                endptr=DataBStrWithSpaces;
                /*PossibleToContinue=true;

                for(k=0; k<DBNumOfParamsTrimmed; k++){ // DBNumOfParamsTrimmed
                    nptr=endptr;
                    ParamVal[k]=_tcstod(nptr, &endptr);
                    if(nptr==endptr){
                        DBFileError_InvalidFormat();
                        return FALSE;
                    }
                }
                //for(k=0; k<conDBNumOfParams; k++){ // DBNumOfParamsTrimmed
                //    nptr=endptr;
                //    ParamVal[k]=_tcstod(nptr, &endptr);
                //    if(nptr==endptr){
                //        DBFileError_InvalidFormat();
                //        return FALSE;
                //    }
                // }
                // }

                // _ftprintf_s( OutputFile.Stream,
                // _T("%2u;%1u;%.6f;%.3E;%.3E;%.4f;%.4f;%.4f;%.2f;%.6f\n"),
                // //Verific.:      %2d %1d %12.6f %10.3E %10.3E %5.4f %5.4f %10.4f
                //4.2f %8.6f
                // // (int)ParamVal[k--], (int)ParamVal[k--], ParamVal[k--
                ], ParamVal[k--], ParamVal[k--], ParamVal[k--], ParamVal[k--], ParamVal[k--], ParamVal[k--
                ], ParamVal[k--] );

```

```

//
(size_t)ParamVal[0], (size_t)ParamVal[1], ParamVal[2], ParamVal[3], ParamVal[4], ParamVal
[5], ParamVal[6], ParamVal[7], ParamVal[8], ParamVal[9] );

//Writing the parameters into the class object:

//Params.M      =   (ULONG)ParamVal[1-1];
//Params.Isot   =   (ULONG)ParamVal[2-1];
//if (Params.M==conMolNum_CDiox &&
Params.Isot<=conNumOfIsotopes_CDiox && Params.Isot>=1){
    ParamValInUse[9] =   ParamVal[1-1]; // Molecule number

    ParamValInUse[0] =   ParamVal[2-1];
    ParamValInUse[1] =   ParamVal[3-1];
    ParamValInUse[2] =   ParamVal[4-1];

    ParamValInUse[3] =   ParamVal[6-1];
    ParamValInUse[4] =   ParamVal[7-1];
    ParamValInUse[5] =   ParamVal[8-1];
    ParamValInUse[6] =   ParamVal[9-1];
    ParamValInUse[7] =   ParamVal[10-1];

    switch (SpecClc->DataBaseType.Num) {
        case defDBType_CDSDFourk:
        case defDBType_CDSDHITEMP:
            ParamValInUse[8]=ParamVal[11-1]; // NB: n_self which is
ParamVal[11-1] is only for CDS4-4000 and CDS4-HITEMP
            break;
        default:
            ParamValInUse[8]=ParamVal[9-1]; // NB: n_self which is
ParamVal[11-1] is only for CDS4-4000 and CDS4-HITEMP
            // Otherwise n_self=n_air which is ParamVal[9-1]
    }

    //Params.Isot      =
(ULONG)ParamValInUse[conDBNumOfParamsInUse-9]; //0
    //Params.v0       =   ParamValInUse[conDBNumOfParamsInUse-8]; //1
    //Params.S        =   ParamValInUse[conDBNumOfParamsInUse-7]; //2
    //Params.g_air    =   ParamValInUse[conDBNumOfParamsInUse-6]; //3
    //Params.g_self   =   ParamValInUse[conDBNumOfParamsInUse-5]; //4
    //Params.E        =   ParamValInUse[conDBNumOfParamsInUse-4]; //5
    //Params.n_air    =   ParamValInUse[conDBNumOfParamsInUse-3]; //6
    //Params.delta_air =   ParamValInUse[conDBNumOfParamsInUse-2]; //7
    //Params.n_self   =   ParamValInUse[conDBNumOfParamsInUse-1]; //8
// NB: n_self ParamVal[11-1] is only for CDS44000

    //_ftprintf_s( OutputFile.Stream,
//_T("%lu;% .6f;% .3E;
%.4f;% .4f;% .4f;% .2f;% .6f;% .4f;\n\n"),
// //Verific.:      %2d %1d %12.6f %10.3E %10.3E %5.4f %5.4f %10.4f
%4.2f %8.6f
// // (int) ParamVal[k--], (int) ParamVal[k--], ParamVal[k--
], ParamVal[k--], ParamVal[k--], ParamVal[k--], ParamVal[k--], ParamVal[k--
], ParamVal[k--] );
//
(UINT)ParamValInUse[0], ParamValInUse[1], ParamValInUse[2], ParamValInUse[3], ParamValIn
Use[4], ParamValInUse[5], ParamValInUse[6], ParamValInUse[7], ParamValInUse[8] );

    //_tprintf_s(_T("
// L"\n\nINFO:\n"
// L"FileLine = %-u\n"
// L"Cntr.FileTrans = %-u\n"
// L"Cntr.TransTotal = %-u\n"
// L"SpecClc->MolNameShort.tBuf = %-s\n"
// L"SpecClc->MolNameLong.tBuf = %-s\n"
// L"SpecClc->MolNum = %-u\n"
// L"(size_t)ParamValInUse[9] = %-u\n"
// L"ParamValInUse[9] = %-.15g\n"
// L"SpecClc->NumOfIsotopes = %-u\n"
// L"(size_t)ParamValInUse[0] = %-u\n"
// L"ParamValInUse[0] = %-.15g\n"
// )

```

```

// ,FileLine
// ,Cntr.FileTrans
// ,Cntr.TransTotal
// ,SpecClc->MolNameShort.tBuf
// ,SpecClc->MolNameLong.tBuf
// ,SpecClc->MolNum
// ,(size_t)ParamValInUse[9]
// ,ParamValInUse[9]
// ,SpecClc->NumOfIsotopes
// ,(size_t)ParamValInUse[0]
// ,ParamValInUse[0]
// );

//_ftprintf_s( OutputFile.Stream, _T("
// L"\n\nINFO:\n"
// L"FileLine = %-u\n"
// L"Cntr.FileTrans = %-u\n"
// L"Cntr.TransTotal = %-u\n"
// L"SpecClc->MolNameShort.tBuf = %-s\n"
// L"SpecClc->MolNameLong.tBuf = %-s\n"
// L"SpecClc->MolNum = %-u\n"
// L"(size_t)ParamValInUse[9] = %-u\n"
// L"ParamValInUse[9] = %-.15g\n"
// L"SpecClc->NumOfIsotopes = %-u\n"
// L"(size_t)ParamValInUse[0] = %-u\n"
// L"ParamValInUse[0] = %-.15g\n"
// )
// ,FileLine
// ,Cntr.FileTrans
// ,Cntr.TransTotal
// ,SpecClc->MolNameShort.tBuf
// ,SpecClc->MolNameLong.tBuf
// ,SpecClc->MolNum
// ,(size_t)ParamValInUse[9]
// ,ParamValInUse[9]
// ,SpecClc->NumOfIsotopes
// ,(size_t)ParamValInUse[0]
// ,ParamValInUse[0]
// );

if (SpecClc->MolNum==(size_t)ParamValInUse[9] &&
1<=(size_t)ParamValInUse[0] && (size_t)ParamValInUse[0]<=SpecClc->NumOfIsotopes) {
//if (1<=ParamValInUse[0] &&
ParamValInUse[0]<=conNumOfIsotopes_CDiox){
//FWriteNumOfElems=;
Cntr.FileTransInUse++;
Cntr.TransTotalInUse++;
if(

fwrite (ParamValInUse, sizeof(double), conDBNumOfParamsInUse, BinaryDBFile.Stream) !=conD
BNumOfParamsInUse

||
ferror (BinaryDBFile.Stream)
){
//FReadBinaryDBFileErrorMessage();
FWriteBinaryDBFileErrorMessage();
//if(
//OutputFile.CloseFile();
//||
//TextDBFile.CloseFile();
//||
//BinaryDBFile.CloseFile();
//)
//return FALSE;
return FALSE;
}
}
//_tprintf_s(_T("\n fwrite = %-u\n"), FWriteNumOfElems);
}

_ftprintf_s(OutputFile.Stream, _T("%-u out of %-u"),
Cntr.FileTransInUse, Cntr.FileTrans);

```

```

        _tprintf_s(
Cntr.FileTransInUse,Cntr.FileTrans);
        if(
            !TextDBFile.CloseFile()
            ||
            !BinaryDBFile.CloseFile())
            return FALSE;
        WNStart=WNStop;
        WNStop=WNStart+1;
    }else{
        switch(SpecClc->MolNum){
            case conMolNum_CDiox:
                WNStop++;
                if(WNStop==defWNStopMax){
                    _ftprintf_s(OutputFile.Stream, _T("
files in the range\nfrom \"
                    defDBFileNamePattern
                    L\" to \"
                    defDBFileNamePattern
                    L\"
\nThe calculations were continued.\n")
                    ,SpecClc->MolNameShort.tBuf
                    ,SpecClc->MolNameLong.tBuf
                    ,WNStart,WNStart+1
                    ,WNStart,defWNStopMax-1
                    );
                    _tprintf_s(_T("
files in the range\nfrom \"
                    defDBFileNamePattern
                    L\" to \"
                    defDBFileNamePattern
                    L\"
\nThe calculations were continued.\n")
                    ,SpecClc->MolNameShort.tBuf
                    ,SpecClc->MolNameLong.tBuf
                    ,WNStart,WNStart+1
                    ,WNStart,defWNStopMax-1);
                    WNStart++;
                    WNStop=WNStart+1;
                }
                break;
            case conMolNum_CMonox:
                _ftprintf_s(OutputFile.Stream, _T("
base file:\n\"%-s\"
                L\"
\nThe application will terminate.\n")
                ,SpecClc->MolNameShort.tBuf
                ,SpecClc->MolNameLong.tBuf
                ,TextDBFile.FullNameCurrent.tBuf
                );
                _tprintf_s(_T("
base file:\n\"%-s\"
                L\"
\nThe application will terminate.\n")
                ,SpecClc->MolNameShort.tBuf
                ,SpecClc->MolNameLong.tBuf
                ,TextDBFile.FullNameCurrent.tBuf
                );
                return FALSE;
                break;
            default:
                _ftprintf_s(OutputFile.Stream, _T("
file: MolNum not recognized. The application will terminate\n")
                L\"
\nERROR:\n\n\"
                L\"Internal error ocured while processing a text data base
                )
                );
                _tprintf_s(_T("
file: MolNum not recognized. The application will terminate\n")
                L\"
\nERROR:\n\n\"
                L\"Internal error ocured while processing a text data base
                )
                );
        }
    }
}

```

```

        return FALSE;
    }
}
}
if(Cntr.Files==0){
    _ftprintf_s(OutputFile.Stream, _T("\n\nERROR:\nNo database files found.\n\n"));
    _tprintf_s(
        _T("\n\nERROR:\nNo database files found.\n\n"));
    return FALSE;
}
//TotalNumOfTrans =Cntr.TransTotal ;
//TotalNumOfTransInUse=Cntr.TransTotalInUse;
_ftprintf_s(OutputFile.Stream, _T("
L"\n\nFinished converting the %-s (%-s) database files to the binary format:\n"
L" The total number of transitions: %-u out of %-u\n"
)
,SpecClc->MolNameShort.tBuf
,SpecClc->MolNameLong.tBuf
,Cntr.TransTotalInUse,Cntr.TransTotal
);
_tprintf_s(_T("
L"\n\nFinished converting the %-s (%-s) database files to the binary format:\n"
L" The total number of transitions: %-u out of %-u\n"
)
,SpecClc->MolNameShort.tBuf
,SpecClc->MolNameLong.tBuf
,Cntr.TransTotalInUse,Cntr.TransTotal
);

//_ftprintf_s( OutputFile.Stream, _T(" g_Voi_min = %-28.18g cm-1\n"), SpecClc-
>g_Voi_min );
return TRUE;
}

```

```

BOOL CFileProcessing::BinaryDataBaseFileProcessing(CSpectralCalcs* SpecClc){

    register ULONG TransTotal=0;

    ULONG WNStart, WNStop, MainFORIterationNumber;
    size_t FReadNumOfElements;

    Cntr.Initialize();
    Cntr.TransTotal=0;
    Cntr.Files=0;

    if (!BinaryDBFileProcAgain){
        switch(SpecClc->MolNum){
            case conMolNum_CDiox:
                _ftprintf_s(OutputFile.Stream, _T("
L"\nThe %-s (%-s) binary database files with a starting wave number
in the range from %-u to %-u have been processed.\n\n"
L"Summary.\nBinary database file and the number of transitions in
it:\n"
)
,SpecClc->MolNameShort.tBuf
,SpecClc->MolNameLong.tBuf
,TextDBFile.WaveNumStart, TextDBFile.WaveNumStartStop
);
                _tprintf_s(_T("
L"\nThe %-s (%-s) binary database files with a starting wave number
in the range from %-u to %-u are being processed...\n\n"
L"Progress...\nBinary database file and the number of transitions
in it:\n"
)
,SpecClc->MolNameShort.tBuf
,SpecClc->MolNameLong.tBuf
,TextDBFile.WaveNumStart, TextDBFile.WaveNumStartStop
);
                break;
            case conMolNum_CMonox:
                _ftprintf_s(OutputFile.Stream, _T("
L"\nThe %-s (%-s) binary database file has been processed.\n\n"
L"Summary.\nThe binary database file and the number of transitions
in it:\n"

```

```

        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        );
    _tprintf_s(_T(
        L"\n\nThe %-s (%-s) binary database file is being processed...\n\n"
        L"Progress...\n\nThe binary database file and the number of transi-
tions in it:\n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        );
        break;
    default:
        _ftprintf_s(OutputFile.Stream, _T(
            L"\n\n\nERROR:\n\n"
            L"Internal error occured before processing binary database files:
MolNum not recognized. The application will terminate\n"
        )
        );
        _tprintf_s(_T(
            L"\n\n\nERROR:\n\n"
            L"Internal error occured before processing binary database files:
MolNum not recognized. The application will terminate\n"
        )
        );
        return FALSE;
    }
}
else{
    // _ftprintf_s(OutputFile.Stream, _T(
    //     L"\n\n\nDATABASE PARAMETERS ("
    //     defDBName
    //     L")\n\n Transition frequency [cm-1] Press.-shift corr.trans.freq.[cm-1]
Line intensity[cm/molecule] Temper.corr.line intensity[cm/molecule]\n"
    // ));

    _tprintf_s(_T(
        L"\n\n\nLine-by-line calculations for %-s (%-s) are in progress...\n\n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        );
    RemainingTimeEstimation.ct_SetTimeIntervalTimer();
    RemainingTimeEstimation.ct_SetGlobalTimer();
}

MainFORIterationNumber=1;
switch(SpecClc->MolNum){
    case conMolNum_CDiox:
        WNStart=TextDBFile.WaveNumStart;
        WNStop=WNStart+1;
        break;
    case conMolNum_CMonox:
        WNStart=TextDBFile.WaveNumStartStop;
        WNStop=WNStart+1;
        //WNStart=WNStop;
        //WNStop=WNStart+1;
        break;
    default:
        _ftprintf_s(OutputFile.Stream, _T(
            L"\n\n\nERROR:\n\n"
            L"Internal error occured before processing the binary data base files:
MolNum not recognized.\n"
        )
        );
        _tprintf_s(_T(
            L"\n\n\nERROR:\n\n"
            L"Internal error occured before processing the binary data base files:
MolNum not recognized. The application will terminate\n"
        )
        );
        return FALSE;
}
}

```

```

for(
;
WNStart<=TextDBFile.WaveNumStartStop && MainFORIterationNumber<ULONG_MAX;
MainFORIterationNumber++
){
    // Opening the current database file:
    if(BinaryDBFile.OpenFileForReading(WNStart,WNStop,SpecClc)){
        Cntr.Files++;
        if (!BinaryDBFileProcAgain){
            _ftprintf_s(OutputFile.Stream, _T("\n%-u. \\"%-s\""),
Cntr.Files, BinaryDBFile.FileNameCurrent.tBuf);
            _tprintf_s(
Cntr.Files, BinaryDBFile.FileNameCurrent.tBuf);

            FileLine=0;
            Cntr.FileTrans=0;

            if(TransTotal==0){
                FileLine++;
                Cntr.FileTrans++;
                TransTotal++;

                //_tprintf_s(_T("\nFile %-u line %-u\n"), Cntr.Files,
FileLine);
                if(
                    //(FReadNumOfElements=

                    fread(ParamValInUse,sizeof(double),conDBNumOfParamsInUse,BinaryDBFile.Stream)!=conDB
NumOfParamsInUse
                    ||
                    ferror(BinaryDBFile.Stream)
                    ){
                        //_ftprintf_s(OutputFile.Stream, _T("\n fread = %-
u\n"), FReadNumOfElements);
                        FReadBinaryDBFileErrorMessage();
                        return FALSE;
                    }
                    //_tprintf_s(_T("\n fread = %-u\n"), FReadNumOfElements);
                    SpecClc->sp_v0sh(ParamValInUse[1],ParamValInUse[7]);
                    SpecClc->sp_g_Dop(SpecClc->MolNum,(size_t)ParamValInUse[0]);
                    //SpecClc-
                    >sp_g_Dop((size_t)ParamValInUse[9],(size_t)ParamValInUse[0]);
                    SpecClc->g_p_min=SpecClc->g_p_max=SpecClc-
                    >sp_g_p(ParamValInUse[6],ParamValInUse[3],ParamValInUse[8],ParamValInUse[4]);
                    SpecClc->g_Voi_min=SpecClc->g_Voi_max=SpecClc->sp_g_Voi();
                    //_ftprintf_s(OutputFile.Stream, _T("
                    // L%-u. \\"%-s\""), TransTotal,
                    SpecClc->g_p, SpecClc->g_p_min, SpecClc->g_p_max);
                }

                //Run over the lines:
                for (;

                    (FReadNumOfElements=fread(ParamValInUse,sizeof(double),conDBNumOfParamsInUse,BinaryD
BFile.Stream))!=0
                    &&
                    !feof(BinaryDBFile.Stream)
                    &&
                    TransTotal<ULONG_MAX;
                ){
                    FileLine++;
                    Cntr.FileTrans++;
                    TransTotal++;
                    //_tprintf_s(_T("\nFile %-u line %-u\n"), Cntr.Files,
FileLine);
                    //_tprintf_s(_T("\n fread = %-u\n"), FReadNumOfElements);
                    if(
                        ferror(BinaryDBFile.Stream)
                        ||
                        FReadNumOfElements!=conDBNumOfParamsInUse
                        ){
                            FReadBinaryDBFileErrorMessage();

```

```

        return FALSE;
    }

    //Params.Isot =
    (ULONG) ParamValInUse [conDBNumOfParamsInUse-9] ;//0
    //Params.v0 = ParamValInUse [conDBNumOfParamsInUse-
8] ;//1
    //Params.S = ParamValInUse [conDBNumOfParamsInUse-
7] ;//2
    //Params.g_air = ParamValInUse [conDBNumOfParamsInUse-
6] ;//3
    //Params.g_self = ParamValInUse [conDBNumOfParamsInUse-
5] ;//4
    //Params.E = ParamValInUse [conDBNumOfParamsInUse-
4] ;//5
    //Params.n_air = ParamValInUse [conDBNumOfParamsInUse-
3] ;//6
    //Params.delta_air = ParamValInUse [conDBNumOfParamsInUse-
2] ;//7
    //Params.n_self = ParamValInUse [conDBNumOfParamsInUse-
1] ;//8

    //_ftprintf_s( OutputFile.Stream,
    //_T("\n %1u;%.6f;%.3E;
%.4f;%.4f;%.4f;%.2f;%.6f;%.4f;\n"),
    // //Verific.: %2d %1d %12.6f %10.3E %10.3E %5.4f %5.4f
%10.4f %4.2f %8.6f
    // //(int)ParamVal [k--], (int)ParamVal [k--], ParamVal [k--
], ParamVal [k--], ParamVal [k--], ParamVal [k--], ParamVal [k--], ParamVal [k--
], ParamVal [k--] );
    //
    (ULONG) ParamValInUse [0], ParamValInUse [1], ParamValInUse [2], ParamValInUse [3], ParamValIn
Use [4], ParamValInUse [5], ParamValInUse [6], ParamValInUse [7], ParamValInUse [8] );

    SpecClc->DBFileFirstProc (ParamValInUse, OutputFile);
    //SpecClc->DBFileSecondProc (Params);
    //_ftprintf_s (OutputFile.Stream, _T("
// L"%-u. %-.18g %-.18g %-.18g\n"), TransTotal,
SpecClc->g_p, SpecClc->g_p_min, SpecClc->g_p_max);
    }
    _ftprintf_s (OutputFile.Stream, _T("%-u"), Cntr.FileTrans);
    _tprintf_s(
        _T("%-u"), Cntr.FileTrans);
    }else{

        //Run over the lines:
        for (;

            fread (ParamValInUse, sizeof (double), conDBNumOfParamsInUse, BinaryDBFile.Stream) == conDB
NumOfParamsInUse

            &&
            TransTotal < ULONG_MAX;
        ){
            TransTotal++;
            SpecClc->DBFileSecondProc (ParamValInUse, OutputFile);

            //if (SpecClc->XUser.Range [0] <= SpecClc->v0sh && SpecClc-
>v0sh <= SpecClc->XUser.Range [1]){
                //_ftprintf_s (OutputFile.Stream, _T("
// L"%#28.18g %#28.18g %#28.18g %#28.18g\n"
// )
// , ParamValInUse [1]
// , SpecClc->v0sh
// , ParamValInUse [2]
// , SpecClc->S_T
// );
                //}

            // remaining time estimation:
            if (RemainingTimeEstimation.IsPossible &&
RemainingTimeEstimation.ct_HasTimeIntervalElapsed ()) {
                RemainingTimeEstimation.ct_GlobalTimeElapsed ();
                if (RemainingTimeEstimation.IsPossible) {

```



```

    RemainingTimeEstimation.RemainingTimeInSec=RemainingTimeEstimation.GlobalTimeElapsed
/TransTotal*(TotalNumOfTrans-TransTotal);
    RemainingTimeEstimation.ct_RemainingTimeForOutput();

    if(_tprintf_s(_T("
time estimation (displayed every %-d seconds)\n"
Minutes\n"
L"\n\nLine-by-line calcs for %-s (%-s)... Remaining
L"Time remaining: %-.4g Days, %-.4g Hours, %-.4g
L"Expected to be completed by: %s"
)
,SpecClc->MolNameShort.tBuf
,SpecClc->MolNameLong.tBuf
,defRemTimeDisplayIntervalInSec

,RemainingTimeEstimation.RemainingTime.DD,RemainingTimeEstimation.RemainingTime.HH,
RemainingTimeEstimation.RemainingTime.MM
,RemainingTimeEstimation.BufFor_ctime.tBuf
)<=0
)
    RemainingTimeEstimation.IsPossible=FALSE;
}
    RemainingTimeEstimation.ct_SetTimeIntervalTimer();
}
}
}
if(!BinaryDBFile.CloseFile())return FALSE;
WNStart=WNStop;
WNStop=WNStart+1;
}else{
switch(SpecClc->MolNum){
case conMolNum_CDiox:
    WNStop++;
    if(WNStop==defWNStopMax){
        if(!BinaryDBFileProcAgain){
            _ftprintf_s(OutputFile.Stream, _T("
database files in the range\nfrom \"
defDBFileNamePattern
defBinaryDBFileNameSuffix
L\" to \"
defDBFileNamePattern
defBinaryDBFileNameSuffix
L\" \n\nThe calculations were continued.\n")
,SpecClc->MolNameShort.tBuf
,SpecClc->MolNameLong.tBuf
,WNStart,WNStart+1
,WNStart,defWNStopMax-1
);
            _tprintf_s(_T("
database files in the range\nfrom \"
defDBFileNamePattern
defBinaryDBFileNameSuffix
L\" to \"
defDBFileNamePattern
defBinaryDBFileNameSuffix
L\" \n\nThe calculations were continued.\n")
,SpecClc->MolNameShort.tBuf
,SpecClc->MolNameLong.tBuf
,WNStart,WNStart+1
,WNStart,defWNStopMax-1);
        }
        WNStart++;
        WNStop=WNStart+1;
    }
}
break;
case conMolNum_CMonox:
    _ftprintf_s(OutputFile.Stream, _T("
base file:\n\"%-s\"
L"\n\n\nERROR:\n\nCould not open the %-s (%-s) binary data-

```

```

        L"\n\n")
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TextDBFile.FullNameCurrent.tBuf
    );
    _tprintf_s(_T("
base file:\n\"%-s\"")
        L"\n\nERROR:\n\nCould not open the %-s (%-s) binary data-
        L"\n\nThe application will terminate.\n")
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TextDBFile.FullNameCurrent.tBuf
    );
    return FALSE;
    break;
default:
    _ftprintf_s(OutputFile.Stream, _T("
file: MolNum not recognized.\n")
        L"\n\nERROR:\n\n"
        L"Internal error occured while processing a binary database
        )
    );
    _tprintf_s(_T("
file: MolNum not recognized. The application will terminate\n")
        L"\n\nERROR:\n\n"
        L"Internal error occured while processing a binary database
        )
    );
    return FALSE;
}
}
}

if(Cntr.Files==0){
    _ftprintf_s(OutputFile.Stream, _T("\n\nERROR:\nNo database files found in binary
format.\n\n"));
    _tprintf_s(
        _T("\n\nERROR:\nNo database files found in binary
format.\n\n"));
    return FALSE;
}

if(BinaryDBFileProcAgain==0){
    TotalNumOfTrans=TransTotal;
    _ftprintf_s(OutputFile.Stream, _T("
mat:\n"
        L"\n\nFinished processing the %-s (%-s) database files in the binary for-
        L"    The total number of transitions: %-u\n"
        L"    g_p   = [ %-.18g   %-.18g ] cm-1 \n"
        L"    g_Voi = [ %-.18g   %-.18g ] cm-1 \n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TotalNumOfTrans
        ,SpecClc->g_p_min, SpecClc->g_p_max
        ,SpecClc->g_Voi_min, SpecClc->g_Voi_max);
    _tprintf_s(_T("
mat:\n"
        L"\n\nFinished processing the %-s (%-s) database files in the binary for-
        L"    The total number of transitions: %-u\n"
        L"    g_p   = [ %-.18g   %-.18g ] cm-1 \n"
        L"    g_Voi = [ %-.18g   %-.18g ] cm-1 \n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        ,TotalNumOfTrans
        ,SpecClc->g_p_min, SpecClc->g_p_max
        ,SpecClc->g_Voi_min, SpecClc->g_Voi_max);

    // _ftprintf_s( OutputFile.Stream, _T("    g_Voi_min = %-28.18g cm-1\n"),
    SpecClc->g_Voi_min );
    BinaryDBFileProcAgain=1;
}
else{
    if(TotalNumOfTrans==TransTotal){

```

```

        _tprintf_s(_T("
            L"\n\nLine-by-line calculations for %-s (%-s) are DONE.\n\n\n"
        )
        ,SpecClc->MolNameShort.tBuf
        ,SpecClc->MolNameLong.tBuf
        );
    }else{
        _ftprintf_s(OutputFile.Stream, _T("
            L"\n\nERROR:\nInvalid format found in one of the binary database files
while scanning them for the second time and performing the line-by-line calcula-
tions.\n\n"));
        _tprintf_s(_T("
            L"\n\nERROR:\nInvalid format found in one of the binary database files
while scanning them for the second time and performing the line-by-line calculations.
The application will terminate.\n\n"));
        return FALSE;
    }
    PrintClosingSeparatorToFileAndStdout(OutputFile.Stream);
    BinaryDBFileProcAgain=0;
}

return TRUE;
}

void CFileProcessing::FReadBinaryDBFileErrorMessage(){
    _tprintf_s(OutputFile.Stream, _T("
        L"\n\nERROR:\nin \"%-s\"\nInvalid format.\n\n"
    ),BinaryDBFile.FullNameCurrent.tBuf
    );
    // _tprintf_s(_T("fread = %-d ERROR\n"), numofreadelements);
    _tprintf_s(_T("
        L"\n\nERROR:\nin \"%-s\"\nInvalid format.\n\nThe application will termi-
nate.\n\n"
    ),BinaryDBFile.FullNameCurrent.tBuf
    );
}

void CFileProcessing::FWriteBinaryDBFileErrorMessage(){
    _tprintf_s(OutputFile.Stream, _T("
        L"\n\nERROR:\nin \"%-s\"\nFailed to write the data into this binary database
file.\n\nThe likely reason is the insufficient disk space.\n\n"
    ),BinaryDBFile.FullNameCurrent.tBuf
    );
    // _tprintf_s(_T("fread = %-d ERROR\n"), numofreadelements);
    _tprintf_s(_T("
        L"\n\nERROR:\nin \"%-s\"\nFailed to write the data into this binary database
file.\n\nThe likely reason is the insufficient disk space.\n\nThe application will termi-
nate.\n\n"
    ),BinaryDBFile.FullNameCurrent.tBuf
    );
}

BOOL CSpectralCalcs::DBFileFirstProc(double DBParamInUse [conDBNumOfParamsInUse], CFile
OutputFile){
    //Params.Isot          = (ULONG) ParamValInUse [conDBNumOfParamsInUse-9]; //0
    //Params.v0            = ParamValInUse [conDBNumOfParamsInUse-8]; //1
    //Params.S             = ParamValInUse [conDBNumOfParamsInUse-7]; //2
    //Params.g_air         = ParamValInUse [conDBNumOfParamsInUse-6]; //3
    //Params.g_self        = ParamValInUse [conDBNumOfParamsInUse-5]; //4
    //Params.E             = ParamValInUse [conDBNumOfParamsInUse-4]; //5
    //Params.n_air         = ParamValInUse [conDBNumOfParamsInUse-3]; //6
    //Params.delta_air     = ParamValInUse [conDBNumOfParamsInUse-2]; //7
    //Params.n_self        = ParamValInUse [conDBNumOfParamsInUse-1]; //8

    //sp_v0sh(DB);
    sp_v0sh(DBParamInUse[1],DBParamInUse[7]);
    sp_g_Dop(MolNum, (size_t)DBParamInUse[0]);
    sp_g_p(DBParamInUse[6],DBParamInUse[3],DBParamInUse[8],DBParamInUse[4]);
}

```

```

    sp_g_Voi();
    // Searching for some minimum and maximum values
    if(g_p<g_p_min) g_p_min=g_p;
    if(g_p>g_p_max) g_p_max=g_p;
    if(g_Voi<g_Voi_min) g_Voi_min=g_Voi;
    if(g_Voi>g_Voi_max) g_Voi_max=g_Voi;
    return TRUE;
}

BOOL CSpectralCalcs::DBFileSecondProc(double DBParamInUse [conDBNumOfParamsInUse], CFile
OutputFile){
    //if(
    //sp_S_T((size_t)DBParamInUse[0],DBParamInUse[1],DBParamInUse[2],DBParamInUse[5])>co
nIntensityCutoff
    //||
    //sp_IsAboveIntensCutoff((size_t)DBParamInUse[0],DBParamInUse[1],DBParamInUse[2],DBP
aramInUse[5])
    //||
    //DBParamInUse[2]>conIntensityCutoff
    //){
    sp_v0sh(DBParamInUse[1],DBParamInUse[7]);
    sp_g_Dop(MolNum,(size_t)DBParamInUse[0]);
    sp_g_p(DBParamInUse[6],DBParamInUse[3],DBParamInUse[8],DBParamInUse[4]);
    sp_g_Voi();

    // Extending the line intensity S:
    //DBParamInUse[2]=DBParamInUse[2]*(1+conSExtension_pc/100); /*DBParamInUse[2];

    // Correcting the line intensity S for temperature:
    sp_S_T((size_t)DBParamInUse[0],DBParamInUse[1],DBParamInUse[2],DBParamInUse[5]);
    //sp_S_T((size_t)DBParamInUse[0],DBParamInUse[1],DBParamInUse[2],DBParamInUse[5]);

    XModel.iiLeftClc(v0sh-XModel.ProfileXThres);
    XModel.iiRightClc(v0sh+XModel.ProfileXThres);

    //_ftprintf_s(OutputFile.Stream, _T("
// L" "%-d %-d"
// )
// ,XModel.iiLeft
// ,XModel.iiRight
// //,ii
// //,kSum.Ax[ii]
//);

    register LONG ii;
    for(ii=XModel.iiLeft; ii<=XModel.iiRight; ii++){ //0<=ii &&
ii<(LONG)XModel.ActualLen && ii<(LONG)kSum.ActualLen
        //if(
        // sp_v_minus_v0sh(XModel.Ax[ii])>=-XModel.ProfileXThres
        // &&
        // v_minus_v0sh<=XModel.ProfileXThres
        // ){

        //sp_v_minus_v0sh(XModel.Ax[ii]);

        //kSum.Ax[ii]=sp_VoiWhi_Common_WithoutAssignment(); // [1/(molecule cm-2)]
        //_ftprintf_s(OutputFile.Stream, _T("
// L" " ii=%-u; v-v0shifted = %-.18g\n"
// )
// ,ii
// ,v_minus_v0sh
// //,ii
// //,kSum.Ax[ii]
//
//
//);

        //kSum.Ax[ii]+=DBParamInUse[2]*sp_VoiWhi(XModel.Ax[ii]-v0sh); // [1/(molecule
cm-2)]

        kSum.Ax[ii]+=S_T*sp_VoiWhi(XModel.Ax[ii]-v0sh); // [1/(molecule cm-2)]

        //v_minus_v0sh,
        //g_p,

```

```

        //g_Voi); // [1/(molecule cm-2)]
        //}
    }
    //}
    return TRUE;
}

//switch(MolNum){
// case conMolNum_CDiox:
//     if (Isot>conNumOfIsotopes_CDiox || Isot<1)Isot=1;
//     g_Dop=sp_g_Dop(v0sh,T,Con.MolW.CDiox[Isot-1]);
//     break;
// case conMolNum_CMonox:
//     if (Isot>conNumOfIsotopes_CMonox || Isot<1)Isot=1;
//     g_Dop=sp_g_Dop(v0sh,T,Con.MolW.CMonox[Isot-1]);
//     break;
// default:
//     g_Dop=0;
//}
//return g_Dop;

BOOL CSpectralCalcs::sp_Q(CFile OutputFile){
    //The code for total internal partition sums here is based on a fortran routine
    downloaded from
    //ftp://cfa-
    ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Sums/T
    IPS_2011.zip

    //for(size_t Isot=1; Isot<=conNumOfIsotopes_CDiox; Isot++){
    // Q_Tref[Isot-1]=sp_InterpLagrange(Isot,Tref);
    // if (fabs(Q_T [Isot-1]=sp_InterpLagrange(Isot,T ))<=DBL_EPSILON) Q_T[Isot-
    1]=DBL_EPSILON;
    // //if (fabs(Q_TIntensCutoff [Isot-
    1]=sp_InterpLagrange(Isot,conIntensityCutoffTemperK ))<=DBL_EPSILON)
    Q_TIntensCutoff[Isot-1]=DBL_EPSILON;
    // QTref_by_QT[Isot-1]=Q_Tref[Isot-1]/Q_T[Isot-1];
    // //QTref_by_QT4000KCutoff[Isot-1]=Q_Tref[Isot-1]/const_Qof4000K[Isot-1];
    // //QTref_by_QT2000KCutoff[Isot-1]=Q_Tref[Isot-1]/const_Qof2000K[Isot-1];
    // //QTref_by_QT1000KCutoff[Isot-1]=Q_Tref[Isot-1]/const_Qof1000K[Isot-1];
    //}

    for(size_t Isot=1; Isot<=NumOfIsotopes; Isot++){
        Q_Tref[Isot-1]=sp_InterpLagrange(Isot,Tref);
        if (fabs(Q_T [Isot-1]=sp_InterpLagrange(Isot,T ))<=DBL_EPSILON) Q_T[Isot-
        1]=DBL_EPSILON;
        //if (fabs(Q_TIntensCutoff [Isot-
        1]=sp_InterpLagrange(Isot,conIntensityCutoffTemperK ))<=DBL_EPSILON)
        Q_TIntensCutoff[Isot-1]=DBL_EPSILON;
        QTref_by_QT[Isot-1]=Q_Tref[Isot-1]/Q_T[Isot-1];
    }

    if(TOutOfRange){
        _fprintf_s(OutputFile.Stream, _T("
        L"\n\nERROR:\nSomething went wrong during the calculation of total internal
        partition sums. Most likely reason: Either T_ref = %-.18g or T_gas = %-.18g K is out of
        the acceptable range which is %-.18g to %-.18g K.\n\n"
        )
        ,Tref,T,TRange[0],TRange[1]
        );
        _tprintf_s(_T("
        L"\n\nERROR:\nSomething went wrong during the calculation of total internal
        partition sums. Most likely reason: Either T_ref = %-.18g or T_gas = %-.18g K is out of
        the acceptable range which is %-.18g to %-.18g K.\n\n"
        )
        ,Tref,T,TRange[0],TRange[1]
        );
        return FALSE;
    }
    return TRUE;
}

double CSpectralCalcs::sp_AtoB(size_t Isot, double aa){

```

```

// SUBROUTINE AtoB(aa,bb,A,B,npt) from
// ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Sums/B
D_TIPS_2011_v1p0.for

```

```

//c
//c
//c*****
// SUBROUTINE AtoB(aa,bb,A,B,npt)
//c*****
//c...LaGrange 3- and 4-point interpolation
//c...arrays A and B are the npt data points, given aa, a value of the
//c...A variable, the routine will find the corresponding bb value
//c
//c...input: aa
//c...output: bb
// implicit DOUBLE PRECISION (a-h,o-z)
// Parameter (Nmax=600)
// dimension A(Nmax),B(Nmax)
//c
//C
//c
// DO 50 I=2,npt
// IF(A(I).GE.aa) THEN
// IF(I.LT.3 .OR. I.EQ.npt) THEN
//C LaGrange three point interpolation
// J = I
// IF(I.LT.3) J = 3
// IF(I.EQ.npt) J = npt
//c....do not divide by zero
// A0D1=A(J-2)-A(J-1)
// IF(A0D1.EQ.0.) A0D1=0.0001
// A0D2=A(J-2)-A(J)
// IF(A0D2.EQ.0.) A0D2=0.0001
// A1D1=A(J-1)-A(J-2)
// IF(A1D1.EQ.0.) A1D1=0.0001
// A1D2=A(J-1)-A(J)
// IF(A1D2.EQ.0.) A1D2=0.0001
// A2D1=A(J)-A(J-2)
// IF(A2D1.EQ.0.) A2D1=0.0001
// A2D2=A(J)-A(J-1)
// IF(A2D2.EQ.0.) A2D2=0.0001
//c
// A0=(aa-A(J-1))*(aa-A(J))/(A0D1*A0D2)
// A1=(aa-A(J-2))*(aa-A(J))/(A1D1*A1D2)
// A2=(aa-A(J-2))*(aa-A(J-1))/(A2D1*A2D2)
//c
// bb = A0*B(J-2) + A1*B(J-1) + A2*B(J)
//c
// ELSE
//C LaGrange four point interpolation
// J = I
//c....do not divide by zero
// A0D1=A(J-2)-A(J-1)
// IF(A0D1.EQ.0.) A0D1=0.0001
// A0D2=A(J-2)-A(J)
// IF(A0D2.EQ.0.) A0D2=0.0001
// A0D3 = (A(J-2)-A(J+1))
// IF(A0D3.EQ.0.) A0D3=0.0001
//c
// A1D1=A(J-1)-A(J-2)
// IF(A1D1.EQ.0.) A1D1=0.0001
// A1D2=A(J-1)-A(J)
// IF(A1D2.EQ.0.) A1D2=0.0001
// A1D3 = A(J-1)-A(J+1)
// IF(A1D3.EQ.0.) A1D3=0.0001
//c
// A2D1=A(J)-A(J-2)
// IF(A2D1.EQ.0.) A2D1=0.0001
// A2D2=A(J)-A(J-1)
// IF(A2D2.EQ.0.) A2D2=0.0001
// A2D3 = A(J)-A(J+1)
// IF(A2D3.EQ.0.) A2D3=0.0001

```

```

//c
//      A3D1 = A(J+1)-A(J-2)
//      IF(A3D1.EQ.0.) A3D1=0.0001
//      A3D2 = A(J+1)-A(J-1)
//      IF(A3D2.EQ.0.) A3D2=0.0001
//      A3D3 = A(J+1)-A(J)
//      IF(A3D3.EQ.0.) A3D3=0.0001
//c
//      A0=(aa-A(J-1))*(aa-A(J))*(aa-A(J+1))
//      A0=A0/(A0D1*A0D2*A0D3)
//      A1=(aa-A(J-2))*(aa-A(J))*(aa-A(J+1))
//      A1=A1/(A1D1*A1D2*A1D3)
//      A2=(aa-A(J-2))*(aa-A(J-1))*(aa-A(J+1))
//      A2=A2/(A2D1*A2D2*A2D3)
//      A3=(aa-A(J-2))*(aa-A(J-1))*(aa-A(J))
//      A3=A3/(A3D1*A3D2*A3D3)
//c
//      bb = A0*B(J-2) + A1*B(J-1) + A2*B(J) + A3*B(J+1)
//      ENDIF
//c
//      GO TO 100
//      ENDIF
//      50 CONTINUE
//      100 CONTINUE
//ccc      write(2,*) 'F1, F2, F3, H1, H2, H3 = ',B(J-2),B(J-1),B(J),
//ccc      + A(J-2), A(J-1), A(J)
//ccc      write(2,*) 'A0, A1, A2, bb = ',A0,A1,A2,bb
//c
//      RETURN
//      END
//c

//const USHORT Nmax=600;

if(Isot<1 || Isot>conNumOfIsotopes_CDiox) Isot=1;

const size_t npt=conparsum_NofTemps;

double A[npt], B[npt],
      A0D1,A0D2,A0D3,
      A1D1,A1D2,A1D3,
      A2D1,A2D2,A2D3,
      A3D1,A3D2,A3D3,
      A0,A1,A2,A3,
      bb;

for (size_t ii=0; ii<npt; ii++){
  A[ii]=constparsum_Temps[ii];
  B[ii]=constparsum_CDiox[Isot-1][ii];
}

size_t I, J;

for(I=2;I<=npt;I++){
  //DO 50 I=2, npt
  if(A[I-1] >= aa)break; // aa is T
  //      IF(A(I) .GE.aa)THEN
}
if(I < 3 || I == npt){
  // IF(I.LT.3 .OR. I.EQ.npt) THEN
  //C      LaGrange three point interpolation
  J = I;
  //      J = I
  if(I < 3) J = 3;
  //      IF(I.LT.3) J = 3
  if(I == npt) J = npt;
  //      IF(I.EQ.npT) J = npt
  //c.....do not devide by zero
  if (fabs(A0D1=A[J-2-1]-A[J-1-1])<=DBL_EPSILON) A0D1=0.0001;
  //      A0D1=A(J-2) -A(J-1)          A0D1=0.0001
  //IF(A0D1.EQ.0.) A0D1=0.0001
  if (fabs(A0D2=A[J-2-1]-A[J-1])<=DBL_EPSILON) A0D2=0.0001;
  //      A0D2=A(J-2) -A(J)          A0D2=0.0001

```

```

//IF (A0D2.EQ.0.) A0D2=0.0001
if (fabs (A1D1=A[J-1-1]-A[J-2-1]) <=DBL_EPSILON) A1D1=0.0001;
//      A1D1=A(J-1)      -A(J-2)      A1D1=0.0001
//IF (A1D1.EQ.0.) A1D1=0.0001
if (fabs (A1D2=A[J-1-1]-A[J-1]) <=DBL_EPSILON) A1D2=0.0001;
//      A1D2=A(J-1)      -A(J)      A1D2=0.0001
//IF (A1D2.EQ.0.) A1D2=0.0001
if (fabs (A2D1=A[J-1]-A[J-2-1]) <=DBL_EPSILON) A2D1=0.0001;
//      A2D1=A(J)      -A(J-2)      A2D1=0.0001
//IF (A2D1.EQ.0.) A2D1=0.0001
if (fabs (A2D2=A[J-1]-A[J-1-1]) <=DBL_EPSILON) A2D2=0.0001;
//      A2D2=A(J)      -A(J-1)      A2D2=0.0001
//IF (A2D2.EQ.0.) A2D2=0.0001
//c
A0=(aa-A[J-1-1])* (aa-A[J-1]) / (A0D1*A0D2);
//      A0=(aa-A(J-1) )*(aa-A(J) ) / (A0D1*A0D2)
A1=(aa-A[J-2-1])* (aa-A[J-1]) / (A1D1*A1D2);
//      A1=(aa-A(J-2) )*(aa-A(J) ) / (A1D1*A1D2)
A2=(aa-A[J-2-1])* (aa-A[J-1-1]) / (A2D1*A2D2);
//      A2=(aa-A(J-2) )*(aa-A(J-1) ) / (A2D1*A2D2)
//c
bb = A0*B[J-2-1] + A1*B[J-1-1] + A2*B[J-1];
//      bb = A0*B(J-2) + A1*B(J-1) + A2*B(J)
}else{
//C      LaGrange four point interpolation
J=I;
//c.....do not devide by zero
if (fabs (A0D1=A[J-2-1]-A[J-1-1]) <=DBL_EPSILON) A0D1=0.0001;
//      A0D1=A(J-2)      -A(J-1)      A0D1=0.0001
//IF (A0D1.EQ.0.) A0D1=0.0001
if (fabs (A0D2=A[J-2-1]-A[J-1]) <=DBL_EPSILON) A0D2=0.0001;
//      A0D2=A(J-2)      -A(J)      A0D2=0.0001
//IF (A0D2.EQ.0.) A0D2=0.0001
if (fabs (A0D3=A[J-2-1]-A[J+1-1]) <=DBL_EPSILON) A0D3=0.0001;
//      A0D3=A(J-2)      -A(J+1)      A0D3=0.0001
//IF (A0D3.EQ.0.) A0D3=0.0001
//c
if (fabs (A1D1=A[J-1-1]-A[J-2-1]) <=DBL_EPSILON) A1D1=0.0001;
//      A1D1=A(J-1)      -A(J-2)      A1D1=0.0001
//IF (A1D1.EQ.0.) A1D1=0.0001
if (fabs (A1D2=A[J-1-1]-A[J-1]) <=DBL_EPSILON) A1D2=0.0001;
//      A1D2=A(J-1)      -A(J)      A1D2=0.0001
//IF (A1D2.EQ.0.) A1D2=0.0001
if (fabs (A1D3=A[J-1-1]-A[J+1-1]) <=DBL_EPSILON) A1D3=0.0001;
//      A1D3=A(J-1)      -A(J+1)      A1D3=0.0001
//IF (A1D3.EQ.0.) A1D3=0.0001
//c
if (fabs (A2D1=A[J-1]-A[J-2-1]) <=DBL_EPSILON) A2D1=0.0001;
//      A2D1=A(J)      -A(J-2)      A2D1=0.0001
//IF (A2D1.EQ.0.) A2D1=0.0001
if (fabs (A2D2=A[J-1]-A[J-1-1]) <=DBL_EPSILON) A2D2=0.0001;
//      A2D2=A(J)      -A(J-1)      A2D2=0.0001
//IF (A2D2.EQ.0.) A2D2=0.0001
if (fabs (A2D3=A[J-1]-A[J+1-1]) <=DBL_EPSILON) A2D3=0.0001;
//      A2D3=A(J)      -A(J+1)      A2D3=0.0001
//IF (A2D3.EQ.0.) A2D3=0.0001
//c
if (fabs (A3D1=A[J+1-1]-A[J-2-1]) <=DBL_EPSILON) A3D1=0.0001;
//      A3D1=A(J+1)      -A(J-2)      A3D1=0.0001
//IF (A3D1.EQ.0.) A3D1=0.0001
if (fabs (A3D2=A[J+1-1]-A[J-1-1]) <=DBL_EPSILON) A3D2=0.0001;
//      A3D2=A(J+1)      -A(J-1)      A3D2=0.0001
//IF (A3D2.EQ.0.) A3D2=0.0001
if (fabs (A3D3=A[J+1-1]-A[J-1]) <=DBL_EPSILON) A3D3=0.0001;
//      A3D3=A(J+1)      -A(J)      A3D3=0.0001
//IF (A3D3.EQ.0.) A3D3=0.0001
//c
A0=(aa-A[J-1-1])* (aa-A[J-1])* (aa-A[J+1-1]);
//A0=(aa-A(J-1) )*(aa-A(J) )*(aa-A(J+1) )
A0=A0/(A0D1*A0D2*A0D3);
//A0=A0/(A0D1*A0D2*A0D3)

```



```

A1=(aa-A[J-2-1])*(aa-A[J-1])*(aa-A[J+1-1]);
//A1=(aa-A(J-2 ))*(aa-A(J ))*(aa-A(J+1 ))
A1=A1/(A1D1*A1D2*A1D3);
//A1=A1/(A1D1*A1D2*A1D3)

A2=(aa-A[J-2-1])*(aa-A[J-1-1])*(aa-A[J+1-1]);
//A2=(aa-A(J-2 ))*(aa-A(J-1 ))*(aa-A(J+1 ))
A2=A2/(A2D1*A2D2*A2D3);
//A2=A2/(A2D1*A2D2*A2D3)

A3=(aa-A[J-2-1])*(aa-A[J-1-1])*(aa-A[J-1]);
//A3=(aa-A(J-2 ))*(aa-A(J-1 ))*(aa-A(J ))
A3=A3/(A3D1*A3D2*A3D3);
//A3=A3/(A3D1*A3D2*A3D3)
//c
bb = A0*B[J-2-1] + A1*B[J-1-1] + A2*B[J-1] + A3*B[J+1-1];
//bb = A0*B(J-2 ) + A1*B(J-1 ) + A2*B(J ) + A3*B(J+1 )
}
return bb;
}

double CSpectralCalcs::sp_InterpLagrange(size_t Isot, double X){
// Lagrange interpolation as given in
// Korn G.A., Korn T.M. Mathematical handbook, Saint-Petersburg, Publisher "Lan",
2003, p.676

if(Isot<1 || Isot>NumOfIsotopes) Isot=1;

const size_t
npt=constparsum_NoFTemps,
HalfNumOfNodes=4,
nptmin=0+(HalfNumOfNodes-1),
nptmax=(npt-1)-(HalfNumOfNodes-1);

size_t ii, jj, nptright;

// the input parameter X is T, x point for interpolation
double
xx[HalfNumOfNodes*2], // this is Tdat, x nodes of interpolation
num_by_den=1,
yy[HalfNumOfNodes*2], // this is QofT, y nodes of interpolation
Y=0; // this is Qtot(T), y value at the x point for interpolation

TRange[0]=constparsum_Temps[nptmin];
TRange[1]=constparsum_Temps[nptmax];
if(X<TRange[0] || X>TRange[1]){
TOutOfRange=TRUE;
return Y=-1;
}

for (nptright=nptmin+1; nptright<=nptmax; nptright++){
if(X<=constparsum_Temps[nptright])break;
}

for (ii=(nptright-1)-(HalfNumOfNodes-1), jj=0; ii<=nptright+(HalfNumOfNodes-1);
ii++, jj++){
// the following constants are taken from
// ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Sums/B
D_TIPS_2011_v1p0.for
xx[jj]=constparsum_Temps[ii]; // this is Tdat, x nodes of interpolation
switch(MolNum){
case conMolNum_CDiox:
yy[jj]=constparsum_CDiox[Isot-1][ii]; // this is QofT, y nodes of
interpolation
break;
case conMolNum_CMonox:
yy[jj]=constparsum_CMonox[Isot-1][ii]; // this is QofT, y nodes of
interpolation
break;
default:
TOutOfRange=TRUE;
return Y=-1;
}
}
}

```

```

    }
}

Y=0;
for (ii=0; ii<HalfNumOfNodes*2; ii++){
    num_by_den=1;
    for (jj=0; jj<HalfNumOfNodes*2; jj++){
        if (jj!=ii){
            num_by_den*=(X-xx[jj])/(xx[ii]-xx[jj]);
        }
    }
    Y+=num_by_den*yy[ii];
}

return Y; // this is Qtot(T), y value at the x point for interpolation
}

double CSpectralCalcs::sp_S_T(double QTrefByQT, double TrefK, double TK, double v0,
double S, double E_lower){
    // returns the temperature correction of the line intensity (in [cm-1/(molecule*cm-
    2)]) (i.e. a corrected line intensity S) according to:
    // formula (A11) in
    // the HITRAN 1996 Article

    //if(Isot<1 || Isot>conNumOfIsotopes_CDiox) Isot=1;

    return S*QTrefByQT*
        exp(-con_c2_HITRunits*E_lower/TK )/
        exp(-con_c2_HITRunits*E_lower/TrefK)*
        (1-exp(-con_c2_HITRunits*v0/TK ))/
        (1-exp(-con_c2_HITRunits*v0/TrefK));

    // VERIFICATION:
    //
    // M.S_T=HT.S.*Qtot_T(Qtot.abcd,M.Tref)/Qtot_T(Qtot.abcd,Spec.T.K).*...
    // exp(-Const.c2_HITRANunits*HT.E_lower/Spec.T.K)/exp(-
    Const.c2_HITRANunits*HT.E_lower/M.Tref).*...
    // (1-exp(-Const.c2_HITRANunits*HT.nu/Spec.T.K))/(1-exp(-
    Const.c2_HITRANunits*HT.nu/M.Tref));
}

double CSpectralCalcs::sp_S_T(size_t Isot, double v0, double S, double E_lower){
    // returns the temperature correction of the line intensity (in [cm-1/(molecule*cm-
    2)]) (i.e. a corrected line intensity S) according to:
    // formula (A11) in
    // the HITRAN 1996 Article

    if(Isot<1 || Isot>NumOfIsotopes) Isot=1;
    //if(Isot<1 || Isot>conNumOfIsotopes_CDiox) Isot=1;

    return S_T=sp_S_T(QTref_by_QT[Isot-1], Tref, T, v0, S, E_lower);
    //return S_T=S*QTref_by_QT[Isot-1]*
    // exp(-con_c2_HITRunits*E_lower/T )/
    // exp(-con_c2_HITRunits*E_lower/Tref)*
    // (1-exp(-con_c2_HITRunits*v0/T ))/
    // (1-exp(-con_c2_HITRunits*v0/Tref));

    // VERIFICATION:
    //
    // M.S_T=HT.S.*Qtot_T(Qtot.abcd,M.Tref)/Qtot_T(Qtot.abcd,Spec.T.K).*...
    // exp(-Const.c2_HITRANunits*HT.E_lower/Spec.T.K)/exp(-
    Const.c2_HITRANunits*HT.E_lower/M.Tref).*...
    // (1-exp(-Const.c2_HITRANunits*HT.nu/Spec.T.K))/(1-exp(-
    Const.c2_HITRANunits*HT.nu/M.Tref));
}

double CSpectralCalcs::sp_n_molec_per_cm3(){
    // returns the concentration n (number of absorbing particles per unit volume in
    [molecule/cm^3]) according to:
    // formula (2) in
    // Thomas Fleckl, Helmut Jaeger and Ingwald Obernberger. Experimental verification
    of gas spectra

```

```

// calculated for high temperatures using the HITRAN/HITEMP database.
// J. Phys. D: Appl. Phys. 35 (2002) 3138-3144:
return n_molec_per_cm3=c/100*p*con_atm/(con_kB*T)*1e-6; // [molecule/cm^3]
// VERIF.:
// n_molecule_per_cm3=c_percent/100*p_atm*Const.atm/(Const.k_Boltzmann*T_K)*1e-6; %
[molecule/cm^3]
// M.n=M.c*Spec.p.atm*Const.atm/(Const.k_Boltzmann*Spec.T.K)*1e-6; % [mole-
cule/cm^3]
}

double CTauAbs::ta_TauTrue(double kSum, double n_molec_per_cm3, double path_cm){
return exp(-kSum*n_molec_per_cm3*path_cm);
// kSum*n_molec_per_cm3*path_cm
}

double CTauAbs::ta_CrossSecFromTau(double tau, double n_molec_per_cm3, double path_cm){
return -log(tau)/(n_molec_per_cm3*path_cm); // [cm2/molecule]
}

double CTauAbs::ta_CrossSecFromAbs(double Abs, double n_molec_per_cm3, double path_cm){
return Abs*log(10.)/(n_molec_per_cm3*path_cm); // [cm2/molecule]
}

double CTauAbs::ta_TauTrue(double Absorbance){ //double kSum_X_n_molec_per_cm3_X_path_cm
return exp (-Absorbance *log(10.));
//return kSum*n_molec_per_cm3*path_cm/log(10.)=ta_Absorbance(double kSum, dou-
ble n_molec_per_cm3, double path_cm);
//return exp(-kSum*n_molec_per_cm3*path_cm)=ta_TauTrue(double kSum, double
n_molec_per_cm3, double path_cm);
}

double CTauAbs::ta_Absorbance(double tau){
// returns the absorbance according to:
// Formula (6) in
// Thomas Fleckl, Helmut Jaeger and Ingwald Obernberger. Experimental verification
of gas spectra
// calculated for high temperatures using the HITRAN/HITEMP database.
// J. Phys. D: Appl. Phys. 35 (2002) 3138-3144
return -log10(tau);
}

double CTauAbs::ta_Absorbance(double kSum, double n_molec_per_cm3, double path_cm){
// returns the absorbance according to:
// Formula (6) in
// Thomas Fleckl, Helmut Jaeger and Ingwald Obernberger. Experimental verification
of gas spectra
// calculated for high temperatures using the HITRAN/HITEMP database.
// J. Phys. D: Appl. Phys. 35 (2002) 3138-3144
return kSum*n_molec_per_cm3*path_cm/log(10.);
}

BOOL CSpectralCalcs::sp_TauTrue_AbsTrue_Clc(){

//double* AxDyn = new double [XModel.ActualLen];
//for(size_t ii=0; ii<UINT_MAX && ii<XModel.ActualLen; ii++){
// AxDyn[ii]=ii+1;
//}

//size_t NumOfElems=0,NumOfElems2=0,NumOfElems3=0;
////for (;(AxDyn+(sizeof(double))*NumOfElems++)!=NULL && NumOfElems<UINT_MAX;);
//for(NumOfElems=0;(AxDyn+ NumOfElems )!=NULL && NumOfElems <UINT_MAX-
10;NumOfElems++);
//for(NumOfElems2=0;(AxDyn+ NumOfElems2++ )!=NULL && NumOfElems2<UINT_MAX-10;
);
//for(NumOfElems3=0;(AxDyn(++NumOfElems3) )!=NULL && NumOfElems3<UINT_MAX-10;
);

//double AxSta[10]; AxSta[10-1]=9;

//if(XUser.IsMemoryAllocated) {free( XUser.Ax
);XUser.IsMemoryAllocated=FALSE;}
//if(XModel.IsMemoryAllocated) {free( XModel.Ax
);XModel.IsMemoryAllocated=FALSE;}

```

```

        //if (kSum.IsMemoryAllocated)                {free( kSum.Ax
);kSum.IsMemoryAllocated=FALSE;}
        //if (Tau.True.IsMemoryAllocated)           {free( Tau.True.Ax
);Tau.True.IsMemoryAllocated=FALSE;}
        //if (Absorbance.True.IsMemoryAllocated)   {free( Absorbance.True.Ax
);Absorbance.True.IsMemoryAllocated=FALSE;}
        //if (Tau.Conv.IsMemoryAllocated)           {free( Tau.Conv.Ax
);Tau.Conv.IsMemoryAllocated=FALSE;}
        //if (Absorbance.Conv.IsMemoryAllocated)   {free( Absorbance.Conv.Ax
);Absorbance.Conv.IsMemoryAllocated=FALSE;}

    _tprintf_s(_T("//_ftprintf_s
L"\n\nCalculating the %-s (%-s) true transmissivity and true absorb-
ance...\n\n"
L"System message:\n"
L"XUser.ActualLen                = %-u\n"
L"defNumOfElems_Dynamic(XUser.Ax) = %-u\n"
//L"defNumOfElems_Dynamic(Tau.Conv.Ax) = %-u\n"
//L"defNumOfElems_Dynamic(Absorbance.Conv.Ax) = %-u\n"
L"XModel.ActualLen              = %-u\n"
L"defNumOfElems_Dynamic(XModel.Ax) = %-u\n"
L"defNumOfElems_Dynamic(kSum.Ax) = %-u\n"
L"defNumOfElems_Dynamic(Tau.True.Ax) = %-u\n"
L"defNumOfElems_Dynamic(Absorbance.True.Ax) = %-u\n"
L"sizeof (double)                = %-u\n"
//L"defNumOfElems_Static(AxSta) = %-u\n"
)
, MolNameShort.tBuf
, MolNameLong.tBuf
, XUser.ActualLen
, defNumOfElems_Dynamic(XUser.Ax)
//, defNumOfElems_Dynamic(Tau.Conv.Ax)
//, defNumOfElems_Dynamic(Absorbance.Conv.Ax)
, XModel.ActualLen
, defNumOfElems_Dynamic(XModel.Ax)
, defNumOfElems_Dynamic(kSum.Ax)
, defNumOfElems_Dynamic(Tau.True.Ax)
, defNumOfElems_Dynamic(Absorbance.True.Ax)
, sizeof (double)
//, defNumOfElems_Static(AxSta)
);

    //_tprintf_s(_T("//_ftprintf_s
// L"\n\nCalculating the %-s (%-s) true transmissivity and true absorb-
ance...\n\n"
// L"Output for Debugging:\n"
// L"XUser.ActualLen                = %-u\n"
// L"XUser.Ax                        = %-u\n"
// L"XModel.ActualLen              = %-u\n"
// L"AxDyn                          = %-u\n"
// L"XModel.Ax                      = %-u\n"
// L"kSum.Ax                        = %-u\n"
// )
// , MolNameShort.tBuf
// , MolNameLong.tBuf
// , XUser.ActualLen
// , sizeof XUser.Ax[0]//_msize(XUser.Ax)/sizeof
XUser.Ax[0]//defNumOfElems_Dynamic(XUser.Ax)
// , XModel.ActualLen
// , defNumOfElems_Dynamic(AxDyn)//_msize(AxDyn)/sizeof(AxDyn[0])//(sizeof
AxDyn/sizeof AxDyn[0])
// , defNumOfElems_Dynamic(XModel.Ax)//_msize(XModel.Ax)/sizeof XModel.Ax[0]//
// , defNumOfElems_Dynamic(kSum.Ax)
// );

//delete AxDyn;

//XUser.Ax[0]=0;
//kSum.Ax[0]=0;

for(size_t ii=0; ii<UINT_MAX && ii<XModel.ActualLen; ii++){
    Tau.True.Ax[ii]= Tau.ta_TauTrue
(kSum.Ax[ii], n_molec_per_cm3, path);

```

```

        Absorb-
ance.True.Ax[ii]=Absorbance.ta_Absorbance(kSum.Ax[ii],n_molec_per_cm3,path);
    }
    _tprintf_s(_T("
        L"         Done.\n\n"));

    return TRUE;
}

BOOL CFileProcessing::Print_Tau_etc(CSpectralCalcs* SpecClc, CSpectralCalcs
CMonoxSpecClc){

    _tprintf_s(_T("
        L"\n\nPrinting the results into the output file...\n"));

    register size_t ii;

    _ftprintf_s(OutputFile.Stream, _T("
        L"\n\nJust checking the number of digits of precision (the numerical output
format was set to \"%%-.50g\"):\"
        L"\npi = %-.50g (wikipedia)\"
        L"\npi = %-.50g (acos(-1))\"
        L\"\"
        )
        ,pi
        ,acos(-1.)
        );

    //  VERIFICATION OF PARTITION FUNCTIONS (Total Internal Partition Sums)
    //  For CO2 (Carbon Dioxide)
    _ftprintf_s(OutputFile.Stream, _T("
        L"\n\nTotal Internal Partition Sums for CO2 (Carbon Dioxide)\n\nObtained by a
routine based (namely based but not copied) on the FORTRAN routine TIPS_2011 downloaded
from:\n\"
        L"ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%%20programs%%20for%%20Partition%%20Su
ms/TIPS_2011.zip\n\n\"
        L"NB: The reference data points for total internal partition sums are NOT taken
from the above routine but from\n\"
        L"ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat\n\n\"
        L"         T(K)                 626                 638                 636
628                 627                 626                 638
637                 828                 728\n\n\"
        ));

    //CO2 (2)
    //      626  .984204E+00   2.8694E+02   1   43.989830
    //      636  1.10574E-02   5.7841E+02   2   44.993185
    //      628  3.94707E-03   6.0948E+02   1   45.994076
    //      627  7.33989E-04   3.5527E+03   6   44.994045
    //      638  4.43446E-05   1.2291E+03   2   46.997431
    //      637  8.24623E-06   7.1629E+03  12   45.997400
    //      828  3.95734E-06   3.2421E+02   1   47.998322
    //      728  1.47180E-06   3.7764E+03   6   46.998291

    const size_t NumOfT=2;
    double T_CDSD4k[NumOfT]={SpecClc->Tref, SpecClc->T};

    for(ii=0; ii<NumOfT; ii++){

        _ftprintf_s(OutputFile.Stream, _T("
            L"%10.8g %28.18g %28.18g %28.18g %28.18g %28.18g %28.18g %28.18g\n\"
            ,T_CDSD4k[ii]
            ,SpecClc->sp_InterpLagrange(1,T_CDSD4k[ii])
            ,SpecClc->sp_InterpLagrange(2,T_CDSD4k[ii])
            ,SpecClc->sp_InterpLagrange(3,T_CDSD4k[ii])
            ,SpecClc->sp_InterpLagrange(4,T_CDSD4k[ii])
            ,SpecClc->sp_InterpLagrange(5,T_CDSD4k[ii])
            ,SpecClc->sp_InterpLagrange(6,T_CDSD4k[ii])
            ,SpecClc->sp_InterpLagrange(7,T_CDSD4k[ii])
            ,SpecClc->sp_InterpLagrange(8,T_CDSD4k[ii])
            );
    }
}

```

```

_ftprintf_s(OutputFile.Stream, _T("
L"\n"));

for(T_CDSD4k[0]=300; T_CDSD4k[0]<=2900; T_CDSD4k[0]+=100){

    _ftprintf_s(OutputFile.Stream, _T("
L"%10.8g %28.18g %28.18g %28.18g %28.18g %28.18g %28.18g %28.18g %28.18g\n")
    ,T_CDSD4k[0]
    ,SpecClc->sp_InterpLagrange(1,T_CDSD4k[0])
    ,SpecClc->sp_InterpLagrange(2,T_CDSD4k[0])
    ,SpecClc->sp_InterpLagrange(3,T_CDSD4k[0])
    ,SpecClc->sp_InterpLagrange(4,T_CDSD4k[0])
    ,SpecClc->sp_InterpLagrange(5,T_CDSD4k[0])
    ,SpecClc->sp_InterpLagrange(6,T_CDSD4k[0])
    ,SpecClc->sp_InterpLagrange(7,T_CDSD4k[0])
    ,SpecClc->sp_InterpLagrange(8,T_CDSD4k[0])
    );
}

// VERIFICATION OF PARTITION FUNCTIONS (Total Internal Partition Sums)
// For CO (Carbon Monoxide)
_ftprintf_s(OutputFile.Stream, _T("
L"\n\nTotal Internal Partition Sums for CO (Carbon Monoxide)\n\nObtained by a
routine based (namely based but not copied) on the FORTRAN routine TIPS_2011 downloaded
from:\n"
L"ftp://cfa-
ftp.harvard.edu/pub/HITRAN2008/Global_Data/Fortran%20programs%20for%20Partition%20Su
ms/TIPS_2011.zip\n\n"
L"NB: The reference data points for total internal partition sums are NOT taken
from the above routine but from\n"
L"ftp://cfa-ftp.harvard.edu/pub/HITRAN2008/Global_Data/parsum.dat\n\n"
L"
T(K)
28
37\n\n"
));

//CO (5)
// 26 .986544E+00 1.0712E+02 1 27.994915
// 36 1.10836E-02 2.2408E+02 2 28.998270
// 28 1.97822E-03 1.1247E+02 1 29.999161
// 27 3.67867E-04 6.5934E+02 6 28.999130
// 38 2.22250E-05 2.3582E+02 2 31.002516
// 37 4.13292E-06 1.3809E+03 12 30.002485

//const size_t NumOfT=2;
//double T_CDSD4k[NumOfT]={Tref, T};
T_CDSD4k[0]=CMonoxSpecClc.Tref;
T_CDSD4k[1]=CMonoxSpecClc.T;

for(ii=0; ii<NumOfT; ii++){

    _ftprintf_s(OutputFile.Stream, _T("
L"%10.8g %28.18g %28.18g %28.18g %28.18g %28.18g %28.18g\n")
    ,T_CDSD4k[ii]
    ,CMonoxSpecClc.sp_InterpLagrange(1,T_CDSD4k[ii])
    ,CMonoxSpecClc.sp_InterpLagrange(2,T_CDSD4k[ii])
    ,CMonoxSpecClc.sp_InterpLagrange(3,T_CDSD4k[ii])
    ,CMonoxSpecClc.sp_InterpLagrange(4,T_CDSD4k[ii])
    ,CMonoxSpecClc.sp_InterpLagrange(5,T_CDSD4k[ii])
    ,CMonoxSpecClc.sp_InterpLagrange(6,T_CDSD4k[ii])
    );
}

_ftprintf_s(OutputFile.Stream, _T("
L"\n"));

for(T_CDSD4k[0]=300; T_CDSD4k[0]<=2900; T_CDSD4k[0]+=100){

    _ftprintf_s(OutputFile.Stream, _T("
L"%10.8g %28.18g %28.18g %28.18g %28.18g %28.18g %28.18g\n")
    ,T_CDSD4k[0]
    ,CMonoxSpecClc.sp_InterpLagrange(1,T_CDSD4k[0])
    ,CMonoxSpecClc.sp_InterpLagrange(2,T_CDSD4k[0])
    ,CMonoxSpecClc.sp_InterpLagrange(3,T_CDSD4k[0])
}

```

```

,CMonoxSpecClc.sp_InterpLagrange(4,T_CDSD4k[0])
,CMonoxSpecClc.sp_InterpLagrange(5,T_CDSD4k[0])
,CMonoxSpecClc.sp_InterpLagrange(6,T_CDSD4k[0])
);
}

//// VERIFICATION of sp_Trapz
//_fprintf_s(OutputFile.Stream, _T("
// L"\n\n\nsp_Trapz\n\n"
// L""
// L"
// X
// Y
sp_Trapz\n"
// ));

//double
// X[]={-8.2e1, -5, 2, -10.1e-2, 0, 1e-2, 1, -10, -2.5e1, 5.6e2},
// Y[]={-8e1, -60, 30, 0, 0, -5.7e-1, 1e-6, -30e2, 2.5e1, -5.6e-2};
//const size_t XYLength=defNumOfElems_Static(X);

//for(ii=0; ii<XYLength; ii++){
//_fprintf_s(OutputFile.Stream, _T("
// L"%#28.18g %#28.18g %#28.18g\n"
// ,X[ii]
// ,Y[ii]
// ,sp_Trapz(X,Y,(ii+1))
// );
//}

//_fprintf_s(OutputFile.Stream, _T("
// L"\n\n\nIMPORTANT\n\n"
// L"Only lines with intensities more than %-.10g cm/molecule at %-.8g K were used
in the summations,\n"
// L"i.e. the line intensity cutoff applied was %-.10g cm/molecule at %-.8g K.\n"
// )
// ,conIntensityCutoff
// ,SpecClc.T//conIntensityCutoffTemperK
// ,conIntensityCutoff
// ,SpecClc.T//conIntensityCutoffTemperK
// );

//_fprintf_s(OutputFile.Stream, _T("
// L"\n\n\nIMPORTANT\n\n"
// L"All line intensities were scaled by k = %-.6g %%\n"
// L"according to Si_scaled=Si+k/100*Si.\n"
// )
// ,conSExtension_pc
// );

_ffprintf_s(OutputFile.Stream, _T("
L"\n\n\nTRUE TRANSMISSIVITY"
L"\n\n
Wave number [cm-1] CO2 Cross-Sec [cm2/molecule] CO Cross-Sec
[cm2/molecule] True Absorbance (CO2+CO) CO True Absorbance True
Transmissivity (CO2+CO)\n"
));
for(ii=0; ii<UINT_MAX && ii<SpecClc->XModel.ActualLen; ii++){
//Tau.True.Ax[ii]=Tau.ta_TauTrue(kSum.Ax[ii],n_molec_per_cm3,path);
//Absorbance.True.Ax[ii]=Absorbance.ta_Absorbance(Tau.True.Ax[ii]);
if(SpecClc->XUser.Range[0]<=SpecClc->XModel.Ax[ii] && SpecClc->XModel.Ax[ii]<=SpecClc->XUser.Range[1]){
_fffprintf_s(OutputFile.Stream, _T(" %#28.18g %#28.18g %#28.18g %#28.18g
%#28.18g %#28.18g\n")
,SpecClc->XModel.Ax[ii]
,SpecClc->kSum.Ax[ii]
,CMonoxSpecClc.kSum.Ax[ii]
,SpecClc->Absorbance.True.Ax[ii]
,CMonoxSpecClc.Absorbance.True.Ax[ii]
,SpecClc->Tau.True.Ax[ii]
);
}
}

_ffprintf_s(OutputFile.Stream, _T("
L"\n\n\nEFFECTIVE TRANSMISSIVITY (CO2+CO) "

```

```

        L"\n\n          Wave Number [cm-1]          Transmissivity
Absorbance  CO Cross-Sec[cm2/molecule]  CO Cross-Sec[cm2/molecule] (Verif.)\n"
        L"\nNB: The last two columns, CO Cross-Sections [cm2/molecule], are only valid
for the case CO2 = 0%\n\n"
    ));

    // Summing the concentrations:
    SpecClc->n_molec_per_cm3+=CMonoxSpecClc.n_molec_per_cm3; // [molecule/cm3]

    for(ii=0; ii<UINT_MAX && ii<SpecClc->XUser.ActualLen; ii++){
        //Tau.True.Ax[ii]=Tau.ta_TauTrue(kSum.Ax[ii],n_molec_per_cm3,path);

        //SpecClc.Absorbance.Conv.Ax[ii]=SpecClc.Absorbance.ta_Absorbance(SpecClc.Tau.Conv.A
x[ii]);
        _ftprintf_s(OutputFile.Stream, _T("%#28.18g %#28.18g %#28.18g %#28.18g
%#28.18g\n")
            ,SpecClc->XUser.Ax[ii]
            ,SpecClc->Tau.Conv.Ax[ii]
            ,SpecClc->Absorbance.Conv.Ax[ii]
            ,SpecClc->Tau.ta_CrossSecFromTau(SpecClc->Tau.Conv.Ax[ii]          ,SpecClc-
>n_molec_per_cm3,SpecClc->path)
            ,SpecClc->Tau.ta_CrossSecFromAbs(SpecClc->Absorbance.Conv.Ax[ii],SpecClc-
>n_molec_per_cm3,SpecClc->path)
        );
    }

    _tprintf_s(_T("
L"    Done.\n\n"));

    return TRUE;
}

void CFileProcessing::DBFileError_InvalidFormat(){
    _ftprintf_s(OutputFile.Stream, _T("
L"\n\nERROR\nin  \"%-s\":\n"
L"Line number %-d in the above file has an invalid format (not compatible with
the "
        defDBName
L" format).\n\n")
        ,TextDBFile.FullNameCurrent.tBuf
        ,FileLine);
    _tprintf_s(_T("
L"\n\nERROR:\nin  \"%-s\":\n"
L"Line number %-d in the above file has an invalid format (not compatible with
the "
        defDBName
L" format).\n\n")
        ,TextDBFile.FullNameCurrent.tBuf
        ,FileLine);
    TextDBFile.CloseFile();
}
//
BOOL CFileProcessing::WaveNumberFileProcessing(CSpectralCalcs* SpecClc){
    // Reading the wave numbers
    if(WaveNumberFile.OpenFile(_T("rt"))){
        if(!WNFileProcAgain){
            PrintOpeningSeparatorToFileAndStdout(OutputFile.Stream);
            _ftprintf_s( OutputFile.Stream, _T(
                "\nWAVE NUMBER AXIS (This summary is printed two times for verification
purposes)\n\nFile with the wave number axis:\n\"%-s\"\n"
                ), WaveNumberFile.FullName.tBuf );
            _tprintf_s(_T(
                "\nWAVE NUMBER AXIS (This summary appears two times for verification
purposes)\n\nFile with the wave number axis:\n\"%-s\"\n"
                ), WaveNumberFile.FullName.tBuf );
        }else{
            if(!SpecClc->XUser.AllocMemFromActualLenAndMaxAlloc()){
                _ftprintf_s(OutputFile.Stream, _T("\nERROR:\nToo many values for the
wave number axis. Failed to allocate memory.\n"));
                _tprintf_s(
                    _T("\nERROR:\nToo many values for the
wave number axis. Failed to allocate memory.\n"));
                return FALSE;
            }
        }
    }
}

```



```

}

register ULONG m=0;
Cntr.Initialize();

//Cycle over the lines
for (FileLine=0, Cntr.FileLines=0; !feof(WaveNumberFile.Stream) &&
FileLine<ULONG_MAX;){
    FileLine++;
    //Reading the line and writing it into a string:
    for(m=0; (tch=_gettc(WaveNumberFile.Stream))!=WEOF && tch!='\n' &&
m<ULONG_MAX; m++){
        if(m<BufSmall.ArrLen)
            BufSmall.tBuf[m]=tch;
    }

    BufSmall.tBuf[__min(BufSmall.ArrLen-1,m)]=0;
    //_ftprintf_s( OutputFile.Stream, _T("%-s\n"), Num2TCHAR.tBuf );

    if(m==0 && (tch=='\n' || tch==WEOF)){
        continue;
    }
    if(m==ULONG_MAX && tch=='\n' && tch!=WEOF){
        _ftprintf_s( OutputFile.Stream, _T("\nERROR\nin \"%-s\":\n"),
WaveNumberFile.FullName.tBuf );
        _ftprintf_s( OutputFile.Stream, _T("Line number %-d in the above file
has an invalid number format.\n"), FileLine);
        _tprintf_s( _T("\n\nERROR:\nin \"%-s\":\n"),
WaveNumberFile.FullName.tBuf );
        _tprintf_s( _T("Line number %-d in the above file has an invalid number
format.\n\n"), FileLine);
        return FALSE;
    }

    //Extracting the first wavenumber from the string:
    nptr=endptr=BufSmall.tBuf;
    double CurrentValue;
    CurrentValue=_tcstod(nptr,&endptr);
    if(nptr==endptr){
        _ftprintf_s( OutputFile.Stream, _T("\nERROR\nin \"%-s\":\n"),
WaveNumberFile.FullName.tBuf );
        _ftprintf_s( OutputFile.Stream, _T("Line number %-d in the above file
has an invalid number format.\n"), FileLine);
        _tprintf_s( _T("\n\nERROR:\nin \"%-s\":\n"),
WaveNumberFile.FullName.tBuf );
        _tprintf_s( _T("Line number %-d in the above file has an invalid number
format.\n\n"), FileLine);
        return FALSE;
    }
    if(
        CurrentValue>=SpecClc->XUser.MainInputFileSpecifiedRange[0]
        &&
        CurrentValue<=SpecClc->XUser.MainInputFileSpecifiedRange[1]
    ){
        Cntr.FileLines++;
        if(WNFileProcAgain){
            SpecClc->XUser.Ax[Cntr.FileLines-1]=CurrentValue;
            //_ftprintf_s( OutputFile.Stream, _T("%25.17g\n"),
SpecClc.UserX[Cntr.FileLines-1] );
            if(Cntr.FileLines>1){
                if(CurrentValue<SpecClc->XUser.Range[0])
                    SpecClc->XUser.Range[0]=CurrentValue;
                if(CurrentValue>SpecClc->XUser.Range[1])
                    SpecClc->XUser.Range[1]=CurrentValue;
            }else{
                SpecClc->XUser.Range[1]=SpecClc->XUser.Range[0]=CurrentValue;
            }
        }
    }
}

if(Cntr.FileLines+1>SpecClc->XUser.MaxAlloc){
    if(WNFileProcAgain){

```

```

        _ftprintf_s(OutputFile.Stream, _T("\n\nWARNING:\n\nas regards \"%-
s\":"\n\nThe maximum number of points (which is %-d) allowed for the wave number axis has
been reached.\n"
        L"The calculations were continued but using the wave number
axis with the first %-d elements.\n\n"), WaveNumberFile.FullName.tBuf, SpecClc-
>XUser.MaxAlloc, Cntr.FileLines);
        _tprintf_s( _T("\n\nWARNING:\n\nas regards \"%-s\":"\n\nThe maximum
number of points (which is %-d) allowed for the wave number axis has been reached.\n"
        L"The calculations were continued but using the wave number
axis with the first %-d elements.\n\n"), WaveNumberFile.FullName.tBuf, SpecClc-
>XUser.MaxAlloc, Cntr.FileLines);
    }
    break;
}
}
if (!WaveNumberFile.CloseFile())
    return FALSE;
SpecClc->XUser.ActualLen=Cntr.FileLines;
if(WNFileProcAgain){
    _ftprintf_s( OutputFile.Stream, _T("
L"Finished processing file with the wave number axis.\n"
L"   Specified range of values: [ %.18g   %.18g ] cm-1\n"
L"   Actual range of values:   [ %.18g   %.18g ] cm-1\n"
L"   The number of values:  %-d\n"
    )
    ,SpecClc->XUser.MainInputFileSpecifiedRange[0], SpecClc-
>XUser.MainInputFileSpecifiedRange[1]
    ,SpecClc->XUser.Range[0], SpecClc->XUser.Range[1]
    ,SpecClc->XUser.ActualLen
    );
    _tprintf_s( _T("
L"Finished processing file with the wave number axis.\n"
L"   Specified range of values: [%.18g   %.18g] cm-1\n"
L"   Actual range of values:   [%.18g   %.18g] cm-1\n"
L"   VERIFICATION:           [%.18g   %.18g] cm-1\n"
L"   The number of values:  %-d\n"
    )
    ,SpecClc->XUser.MainInputFileSpecifiedRange[0], SpecClc-
>XUser.MainInputFileSpecifiedRange[1]
    ,SpecClc->XUser.Range[0], SpecClc->XUser.Range[1]
    ,SpecClc->XUser.Ax[0], SpecClc->XUser.Ax[SpecClc->XUser.ActualLen-1]
    ,SpecClc->XUser.ActualLen
    );
    PrintClosingSeparatorToFileAndStdout(OutputFile.Stream);
    WNFileProcAgain=0;
}
else{
    WNFileProcAgain=1;
}
}
else{
    _ftprintf_s( OutputFile.Stream, _T("\n\nERROR\nUnable to open file with the wave
number axis:\n\"%-s\""), WaveNumberFile.FullName.tBuf);
    //_tprintf_s( _T("\n\nERROR\nUnable to open file with the wave number axis:\n\"%-
s\""), WaveNumberFile.FullName.tBuf);
    return FALSE;
}
return TRUE;
}

BOOL CFileProcessing::MainInputFileProcessing(CSpectralCalcs* SpecClc){

    FILE* Stream;
    if( (_tfopen_s( &Stream,
defMainInputFileName,
_T("rt") )) != 0 ){
        //cout<<endl<<"ERROR:"<<endl<<"Unable to open the main input
file:"<<(PCHAR)defMainInputFileName<<endl;
        _tprintf_s( _T("\n\nERROR:\nUnable to open the main input file:
\n\"%-s\""\n\n"),defMainInputFileName);
        return FALSE;
    }
    else{
        register ULONG m=0, n=0, k=0;
        BOOL RightQuote=FALSE;

```

```

Cntr.Initialize();
// Reading the entries in the main input file

//Cycle over the lines
for ( FileLine=0, Cntr.FileLines=0; Cntr.FileLines<=defMainInputFileNumOfEntries
&& !feof(Stream) && FileLine<ULONG_MAX; ){
    FileLine++;

    //Reading the line and writing it into a string:
    for(m=0; (tch=_gettc(Stream))!=WEOF && tch!='\n' && m<ULONG_MAX; m++){
        if(m<BufLarge.ArrLen)
            BufLarge.tBuf[m]=tch;
    }

    BufLarge.tBuf[ __min(BufLarge.ArrLen-1,m) ]=0;
    //_ftprintf_s( OutputFile.Stream, _T("%-s\n"), Num2TCHAR.tBuf );

    if(m==0 && (tch=='\n' || tch==WEOF)){
        //Cntr.FileLines--;
        continue;
    }
    if(m==ULONG_MAX && tch!='\n' && tch!=WEOF){
        _tprintf_s(_T(
            "\nERROR:\nWrong string format in file: \n\"%-s\"\nLine number %-d.
The newline character has not been found.\n\n"
        ),defMainInputFileName,FileLine);
        return FALSE;
    }

    for(n=0;BufLarge.tBuf[n]!='\"' && BufLarge.tBuf[n]!=0 && n<=m &&
n<BufLarge.ArrLen && n<ULONG_MAX;n++);
    if(BufLarge.tBuf[n]!='\"'){
        //_tprintf_s(_T(
            //"\nERROR:\nWrong string format in file:\n\"%-s\"\nLine number %-d. The
newline character has not been found.\n\n"
        //),defMainInputFileName,FileLine);
        continue;
    }
    RightQuote=FALSE;
    Cntr.FileLines++;
    n++;

    if(

        Cntr.FileLines>=1

        &&

        Cntr.FileLines<=4

    ){
        for(k=0;BufLarge.tBuf[n]!=0 && n<=m && n<BufLarge.ArrLen-1 &&
k<Folder.ArrLen-1 && n<ULONG_MAX;n++,k++){
            if (BufLarge.tBuf[n]=='\"'){
                RightQuote=TRUE;
                BufLarge.tBuf[n]=0;
            }
            switch(Cntr.FileLines){

                case 1:

                    WaveNumberFile.Path.tBuf[k]=
                    OutputFile.Path.tBuf[k]=
                    //Folder.tBuf[k]=
                    BufLarge.tBuf[n];
                    break;

                case 2:

                    OutputFile.FileName.tBuf[k]=BufLarge.tBuf[n];
                    break;

```

```

    case 3:
        WaveNumberFile.FileName.tBuf[k]=BufLarge.tBuf[n];
        break;

    case 4:
        TextDBFile.Path.tBuf[k]=
            BinaryDBFile.Path.tBuf[k]=
                BufLarge.tBuf[n];
        break;
    }
    if(!RightQuote){
        Cntr.FileLines--;
        continue;
    }
}
if(
    Cntr.FileLines>=5
    &&
    Cntr.FileLines<=defMainInputFileNumOfEntries
){
    //Extracting the first wavenumber from the string:
    nptr=endptr=BufLarge.tBuf+n;/**sizeof(TCHAR);
    double CurrentValue;
    CurrentValue=_tcstod(nptr,&endptr);
    if(nptr==endptr){
        Cntr.FileLines--;
        continue;
    }
    switch(Cntr.FileLines){

    case 5:
        TextDBFile.WaveNumStart=(ULONG)CurrentValue;
        nptr=endptr;
        CurrentValue=_tcstod(nptr,&endptr);
        if(nptr==endptr){
            Cntr.FileLines--;
            continue;
        }
        TextDBFile.WaveNumStartStop=(ULONG)CurrentValue;
        break;

    case 6:
        SpecClc->XUser.MainInputFileSpecifiedRange[0]=CurrentValue;
        nptr=endptr;
        CurrentValue=_tcstod(nptr,&endptr);
        if(nptr==endptr){
            Cntr.FileLines--;
            continue;
        }
        SpecClc->XUser.MainInputFileSpecifiedRange[1]=CurrentValue;
        break;

    case 7:
        SpecClc->Tref=CurrentValue;
        break;

    case 8:
        SpecClc->T=CurrentValue;
        break;

    case 9:

```

```

        SpecClc->p=CurrentValue;
        break;

    case 10:

        SpecClc->path=CurrentValue;
        break;

    case 11:

        SpecClc->c=CurrentValue;
        break;

    case 12:

        SpecClc->ILSParm=CurrentValue;
        break;

        //case 12:

        // SpecClc->ILSType.Num=(USHORT)CurrentValue;
        // break;
        //default;;
    }
}

if (Cntr.FileLines<defMainInputFileNumOfEntries){
    _tprintf_s(_T("\n\nERROR:\nWrong format in the main input file: \n\"%-
s\"\nCould not find a closing quote.\n"),defMainInputFileName);
    return FALSE;
}
if (fclose(Stream)){
    _tprintf_s(_T("\n\nERROR:\nAn internal error has occurred. Unable to close
the main input file: \n\"%-s\"\n\n"),defMainInputFileName);
    //MainExitPrompt();
    return FALSE;
}

_tprintf_s(_T(""
L"\nThe main input file \"%-s\" has been processed (This summary is dis-
played two times for the purpose of verification).\n\nThe following values will be
used:\n\n"
L"Output file: \n\"%-s\"\"
L\"%-s\"\"
L"File with the wave number axis: \n\"%-s\"\"
L\"%-s\"\"
L"Specified spectral range of interest: [%-.18g    %-.18g] cm-1\n\n"

//L"Database files: \n\"%-s\"\"
//defDBFileNamePattern
//L\"\"\"\"\"\"

L"T_ref = %-.8g K\n"
L"T_gas = %-.8g K\n"
L"p_tot = %-.6g atm\n"
L"path = %-.6g cm\n"
L"CO_2 = %-.6g %%\n"
L"CO = %-.6g %%\n"
L"ILS = %-.10g cm-1\n\n"
L\"\"\"\"
)
,defMainInputFileName
,OutputFile.Path.tBuf
,OutputFile.FileName.tBuf
,WaveNumberFile.Path.tBuf
,WaveNumberFile.FileName.tBuf
,SpecClc->XUser.MainInputFileSpecifiedRange[0],SpecClc-
>XUser.MainInputFileSpecifiedRange[1]

//,TextDBFile.Path.tBuf
//,TextDBFile.WaveNumStart
//,TextDBFile.WaveNumStartStop

```

```

        ,SpecClc->Tref
        ,SpecClc->T
        ,SpecClc->p
        ,SpecClc->path
        ,SpecClc->c
        ,defMoleFract_CMonox
        ,SpecClc->ILSParam
    );
}
return TRUE;
}

BOOL CFileProcessing::InitializationOfSpecClc(CSpectralCalcs* SpecClc){
//SpecClc->MolNameLong.tBuf=_T("Carbon Dioxide");
//SpecClc->MolNameShort.tBuf=_T("CO2");

    _stprintf_s(SpecClc->MolNameLong.tBuf, SpecClc->MolNameLong.ArrLen, _T(" "
        L"%-s")
        ,defNameLong_CDiox);
    _stprintf_s(SpecClc->MolNameShort.tBuf, SpecClc->MolNameShort.ArrLen, _T(" "
        L"%-s")
        ,defNameShort_CDiox);

    SpecClc->MolNum=conMolNum_CDiox;
    SpecClc->NumOfIsotopes=conNumOfIsotopes_CDiox;

    SpecClc->Con.CalcCon();
    SpecClc->sp_ps();
    SpecClc->sp_n_molec_per_cm3();

    switch (SpecClc->DataBaseType.Num){
        case defDBType_HITEMP2010:
            SpecClc->Tref_g_self=296; // [K]
            break;
            // Verific.:          296; % [K]
            //          if strcmp(DataB, 'HITEMP2010', 10) ||...
            //          strcmp(DataB, 'HITEMP1995', 10)
            //          Tref_K_gamma_self=296; % [K]
            //          end
            //          if strcmp(DataB, 'CDSD-1000', 9)
            //          Tref_K_gamma_self=1000; % [K]
            //          end
            //case defDBType_CDSDOnek:
            // break;
        default:
            SpecClc->Tref_g_self=SpecClc->Tref;
    }

    if(
        !SpecClc->sp_Q(OutputFile)
    )
    {
        return FALSE;
    }

    return TRUE;
}

BOOL CFileProcessing::PreliminaryOutput(CSpectralCalcs SpecClc, CSpectralCalcs
CMonoxSpecClc){
    if(
        _ftprintf_s(OutputFile.Stream, _T(" "
            defDBName
            L" Application\NDTU Kemiteknik 2012\n\n"
            L"Calculation Results\n\n"
            L"A summary is given first. Calculated transmissivities (and absorbances) are
            given after that.\nBrowse the summary for possible WARNINGS and ERRORS."
            L"\n\n"
        ))<=0
        ||
    )

```

```

        _tstrdate_s(BufSmall.tBuf)!=0
        ||
        _tstrtime_s(Num2TCHAR.tBuf)!=0
        ||
        _ftprintf_s(OutputFile.Stream, _T("Date (mm/dd/yy) %-s    Time (hh:mm:ss) %-
s\n\n"), BufSmall.tBuf, Num2TCHAR.tBuf)<=0
        ||
        _ftprintf_s(OutputFile.Stream,
        _T("
L"The CO2 (Carbon Dioxide) database: \"
defDBName
L\"\n
L>Note: see the list of the CO2 (Carbon Dioxide) database files below, under the
heading \"CO2 (Carbon Dioxide) DATABASE FILES\"\n
L\nThe CO (Carbon Monoxide) database file name: \"%-s\"\n\n

L"The temperature of ALL database files: T_ref = %-.8g K\n

L\nGas temperature: T_gas = %-.8g K\n
L"Total pressure: p_tot = %-.8g atm\n
L"Path length: path = %-.8g cm\n
L"Mole fraction of CO2 (Carbon Dioxide): c(CO2) = %-.6g %%\n
L"Concentration of CO2 (Carbon Dioxide): n(CO2) = %-.18g molecule/cm3\n
L"Mole fraction of CO (Carbon Monoxide): c(CO) = %-.6g %%\n
L"Concentration of CO (Carbon Monoxide): n(CO) = %-.18g molecule/cm3\n

L\nLine profile type: \"Voigt\" (according to eq. (6) - (7) in Morrison et al.
J Electrochem. Soc. 145 (9) 1998)\n
L"ILS function type: \"%-X\" (\\"%-X\" - sinc; \\"%-X\" - Triangular)\n
L"ILS function parameter: ILS = %-.10g cm-1\n
L(Verification:      ILS = %-.10g cm-1)\n
)
, defDBFileName_CMonox
, SpecClc.Tref
, SpecClc.T
, SpecClc.p
, SpecClc.path
, SpecClc.c
, SpecClc.n_molec_per_cm3
, CMonoxSpecClc.c //defMoleFract_CMonox
, CMonoxSpecClc.n_molec_per_cm3
, defILSType
, defILSType_sinc
, defILSType_Triangular
, SpecClc.ILSParam
, CMonoxSpecClc.ILSParam
)<=0
)
{
    return FALSE;
}

return TRUE;
}

CSpectralCalcs::CSpectralCalcs(){
    ProfileType.Num=defProfileType_VoiWhi; // Voigt-Whiting
    //ILSType.Num=defILSType_sinc; // sinc, the ILSParam is the full width of the main
maximum at zero
    //ILSType.Num=defILSType_Triangular; // Triangular, the ILSParam is the length of
the base of the isosceles triangle
    ILSType.Num=defILSType;
    DataBaseType.Num=defDBType;
    TOutOfRange=FALSE;
    MolNum=0;
    NumOfIsotopes=0;

    //_tprintf_s(_T("
// L\n\nSystem message:\nConstructor\nConstructor CSpectralCalcs() has been exe-
cuted.\nConstructor\n\n")
    // )
    // );
}

```

```

BOOL CMonoxFileProcessing::InitializationOfCMonoxSpecClc(CSpectralCalcs* CMonoxSpecClc,
CSpectralCalcs SpecClc){
    //CMonoxSpecClc->MolNameLong.tBuf=_T("Carbon Monoxide");
    //CMonoxSpecClc->MolNameShort.tBuf=_T("CO");

    _stprintf_s(CMonoxSpecClc->MolNameLong.tBuf, CMonoxSpecClc->MolNameLong.ArrLen,
_T(" "
    L"%-s"
    ,defNameLong_CMonox);
    _stprintf_s(CMonoxSpecClc->MolNameShort.tBuf, CMonoxSpecClc->MolNameShort.ArrLen,
_T(" "
    L"%-s"
    ,defNameShort_CMonox);

    CMonoxSpecClc->MolNum=conMolNum_CMonox;
    CMonoxSpecClc->NumOfIsotopes=conNumOfIsotopes_CMonox;

    CMonoxSpecClc->c=defMoleFract_CMonox; // [%]

    CMonoxSpecClc->Con.CalcCon();

    CMonoxSpecClc->DataBaseType.Num=defDBType_HITEMP2010;
    CMonoxSpecClc->ILSParam=SpecClc.ILSParam;

    CMonoxSpecClc->p=SpecClc.p;
    CMonoxSpecClc->path=SpecClc.path;

    CMonoxSpecClc->T=SpecClc.T;
    CMonoxSpecClc->Tref=SpecClc.Tref;
    CMonoxSpecClc->sp_n_molec_per_cm3();
    CMonoxSpecClc->sp_ps();
    //CMonoxSpecClc->n_molec_per_cm3;
    //CMonoxSpecClc->ps;

    //CMonoxSpecClc->Tref_g_self=CMonoxSpecClc->Tref;
    CMonoxSpecClc->Tref_g_self=296; // [K]
    // Verific.: 296; % [K]
    // if strcmp(DataB, 'HITEMP2010', 10) ||...
    //     strcmp(DataB, 'HITEMP1995', 10)
    //     Tref_K_gamma_self=296; % [K]
    // end
    // if strcmp(DataB, 'CDS-1000', 9)
    //     Tref_K_gamma_self=1000; % [K]
    // end

    if(
        !CMonoxSpecClc->sp_Q(OutputFile)
    )
    {
        return FALSE;
    }

    return TRUE;
}

BOOL CMonoxFileProcessing::AxesCopyForCMonoxSpecClc(CSpectralCalcs* CMonoxSpecClc,
CSpectralCalcs SpecClc, CFileProcessing FileProc){

    //CMonoxSpecClc->g_p_min=SpecClc.g_p_min;
    //CMonoxSpecClc->g_p_max=SpecClc.g_p_max;

    CMonoxSpecClc->g_Voi_min=SpecClc.g_Voi_min;
    //CMonoxSpecClc->g_Voi_max=SpecClc.g_Voi_max;

    //CMonoxSpecClc->XModel.ILS_HalfX=SpecClc.XModel.ILS_HalfX;
    //     XModel.ILS_HalfX     XModel.ILS_HalfX

    if(
        !CMonoxSpecClc->XModelCreate(FileProc.OutputFile) //
        XModel.Step=fabs(g_Voi_min/XModel.FactorForMinGamma);
        ||

```



```

        !CMonoxSpecClc->LineProfileWingCutoff(FileProc.OutputFile) // double
RefProfVal=conLineProfileWingCutoffCoeff*sp_VoiWhi(0,g_p_max,g_Voi_max); // [unitless]
    )
    return FALSE;

    return TRUE;
}

BOOL CCMonoxFileProcessing::AbsorbanceTrueSummation(CSpectralCalcs* SpecClc,
CSpectralCalcs CMonoxSpecClc){
    _tprintf_s(_T("
L"\n\nObtaining the true absorbance and true transmissivity for the mixture of
CO2 and CO...\n"));

    for(size_t ii=0; ii<UINT_MAX && ii<SpecClc->XModel.ActualLen; ii++){
        SpecClc->Absorbance.True.Ax[ii]+=
            CMonoxSpecClc. Absorb-
ance.True.Ax[ii];
        SpecClc->
Tau.True.Ax[ii]=SpecClc->Tau.ta_TauTrue(SpecClc-
>Absorbance.True.Ax[ii]);
    }
    _tprintf_s(_T("
L" Done.\n\n"));

    return TRUE;
}

BOOL CSpectralCalcs::DeallocMem(){
    if(IsTimeToDestruct){
        if(XUser.IsMemoryAllocated) {free( XUser.Ax
);XUser.IsMemoryAllocated=FALSE;}
        if(Tau.Conv.IsMemoryAllocated) {free( Tau.Conv.Ax
);Tau.Conv.IsMemoryAllocated=FALSE;}
        if(Absorbance.Conv.IsMemoryAllocated) {free( Absorbance.Conv.Ax
);Absorbance.Conv.IsMemoryAllocated=FALSE;}
        //
        if(XModel.IsMemoryAllocated) {free( XModel.Ax
);XModel.IsMemoryAllocated=FALSE;}
        if(kSum.IsMemoryAllocated) {free( kSum.Ax
);kSum.IsMemoryAllocated=FALSE;}
        if(Tau.True.IsMemoryAllocated) {free( Tau.True.Ax
);Tau.True.IsMemoryAllocated=FALSE;}
        if(Absorbance.True.IsMemoryAllocated) {free( Absorbance.True.Ax
);Absorbance.True.IsMemoryAllocated=FALSE;}
    }
    return TRUE;//
}

// File "User_Funcs.cpp"

// Contains
// the definitions of functions which are used by the user (programmer)

#include <tchar.h>
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
//#include <string.h>
#include <errno.h>
#include <windows.h>
//#include <complex>
#include <math.h>
#include <time.h>

```

```

#include "Define.h"
#include "Header.h"
#include "Spectr.h"

using namespace std;

void ErrorMessage(TCHAR* lpText){
    _tprintf_s(_T("\n\nERROR:\n%-s\n\n"),lpText);
}

void WelcomeMessage(TCHAR* lpText){
    _tprintf_s(_T("\n%-s\n\n"),lpText);
}

void MainExitPrompt(){
    int NumOfStreamsClosed=_fcloseall();
    _tprintf_s(_T("\n\nSystem message: the number of streams closed is %-
d.\n"),NumOfStreamsClosed);
    _tprintf_s(_T("
L"\n\nThe application has finished working."
L"\nThank you for using the "
defDBName
L" application."
L"\n\nEXIT: to exit press any symbol key and then ENTER.\n"
));
    LONG TypeAnyDigitToExit;
    cin>>TypeAnyDigitToExit;
    exit(0);
}
//
void EmergencyTerminationPrompt(){
    int NumOfStreamsClosed=_fcloseall();
    _tprintf_s(_T("\n\nSystem message: the number of streams closed is %-
d.\n"),NumOfStreamsClosed);
    _tprintf_s(_T("\n\nTERMINATION: to exit press any symbol key and then ENTER.\n"));
    LONG TypeAnyDigitToExit;
    cin>>TypeAnyDigitToExit;
    exit(1);
}

BOOL PrintOpeningSeparatorToFileAndStdout(FILE* Stream){
    if(
        _ftprintf_s(Stream,
        _T("\n\n\n::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
|
|
        _tprintf_s(_T("\n\n\n::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
|
|
        return FALSE;
    else
        return TRUE;
}

BOOL PrintClosingSeparatorToFileAndStdout(FILE* Stream){
    if(
        _ftprintf_s(Stream,
        _T("\n\n\n::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
|
|
        _tprintf_s(_T("\n\n\n::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
|
|
        return FALSE;
    else
        return TRUE;
}

BOOL CFile::IsFullNameLong(){
    FullName.StrLenTot=

```

```

        _tccscnlen(Path.tBuf, Path.ArrLen) +
        _tccscnlen(FileName.tBuf, FileName.ArrLen);
    if ((FullName.StrLenTot+3+20) >= FullName.ArrLen) {
        ErrorMessage(_T("The specified working directory name and/or one of the file
names are too long!"));
        return TRUE;
    }else
        return FALSE;
}

BOOL CFile::CreateFullName() {
    //_tcscopy_s(FullName.tBuf, Path.tBuf);
    //_tcscat_s(FullName.tBuf, _T("\\"));
    //_tcscat_s(FullName.tBuf, FileName.tBuf);
    if(
        _stprintf_s(FullName.tBuf, FullName.ArrLen, _T(
            ""
            L"%-s\\%-s"
            )
            , Path.tBuf
            , FileName.tBuf) > 0
        )
        return TRUE;
    else
        return FALSE;
}

BOOL CFile::OpenFile(TCHAR* mode) {
    CreateFullName();
    if(
        !CreateFullName()
        ||
        _tfopen_s(&Stream, FullName.tBuf, mode) != 0
    ){
        _tprintf_s(_T("\n\nERROR:\nUnable to open: \n\"%-s\"\n\nNo such directory or
file.\n\n"), FullName.tBuf);
        return FALSE;
    }else
        return TRUE;
}

BOOL CFile::CloseFile() {
    if (fclose(Stream)) {
        _tprintf_s(_T("\n\nERROR:\nInternal error: failed to close file: \n\"%-s\"\n\nThe
application will terminate.\n\n"), FullName.tBuf);
        //EmergencyTerminationPrompt();
        return FALSE;
    }else
        return TRUE;
}

BOOL CFileMultiple::OpenCurrentFile(ULONG WNStart, ULONG WNStop, CSpectralCalcs*
SpecClc) {
    switch (SpecClc->MolNum) {
        case conMolNum_CDiox:
            if(
                _stprintf_s(FileNameCurrent.tBuf, FileNameCurrent.ArrLen, _T("
defDBFileNamePattern)
, WNStart, WNStop) <= 0
            ){
                return FALSE;
            }
            break;
        case conMolNum_CMonox:
            if(
                _stprintf_s(FileNameCurrent.tBuf, FileNameCurrent.ArrLen, _T("
L"%-s"
)
, defDBFileName_CMonox
) <= 0
            ){
                return FALSE;
            }
    }
}

```

```

        break;
    default:
        _tprintf_s(_T("
L"\n\nERROR:\nInternal error: failed to create the full name for a text
database file (due to MolNum not recognized). The application will terminate.\n\n"));
        return FALSE;
    }
    if(
        _stprintf_s(FullNameCurrent.tBuf, FullNameCurrent.ArrLen, _T("
L"%-s\\%-s"
)
        ,Path.tBuf
        ,FileNameCurrent.tBuf)<=0
        ||
        _tfopen_s(&Stream, FullNameCurrent.tBuf, _T("rt"))!=0
        ){
        return FALSE;
    }
    return TRUE;
}
//
BOOL CFileMultipleBinary::CreateFullName(ULONG WNStart, ULONG WNStop, CSpectralCalcs*
SpecClc){
    switch(SpecClc->MolNum){
        case conMolNum_CDiox:
            if(
                _stprintf_s(FileNameCurrent.tBuf, FileNameCurrent.ArrLen, _T("
defDBFileNamePattern
defBinaryDBFileNameSuffix)
                ,WNStart,WNStop)<=0
                ){
                _tprintf_s(_T("
L"\n\nERROR:\nInternal error: failed to create the file name
for a CO2 (Carbon Dioxide) binary database file. The application will terminate.\n\n"));
                return FALSE;
            }
            break;
        case conMolNum_CMonox:
            if(
                _stprintf_s(FileNameCurrent.tBuf, FileNameCurrent.ArrLen, _T("
L"%-s"
defBinaryDBFileNameSuffix)
                ,defDBFileName_CMonox
                )<=0
                ){
                _tprintf_s(_T("
L"\n\nERROR:\nInternal error: failed to create the file name
for a CO (Carbon Monoxide) binary database file. The application will terminate.\n\n"));
                return FALSE;
            }
            break;
        default:
            _tprintf_s(_T("
L"\n\nERROR:\nInternal error: failed to create the file name for one of
the binary database files (due to MolNum not recognized). The application will termi-
nate.\n\n"));
            return FALSE;
    }
    if(
        _stprintf_s(FullNameCurrent.tBuf, FullNameCurrent.ArrLen, _T("
L"%-s\\%-s"
)
        ,Path.tBuf
        ,FileNameCurrent.tBuf)<=0
        ){
        _tprintf_s(_T("
L"\n\nERROR:\nInternal error: failed to create the full name for a bina-
ry database file. The application will terminate.\n\n"));
        return FALSE;
    }
    return TRUE;
}

```

```

//
BOOL CFileMultipleBinary::CreateFileForWriting(CFileMultiple TextDataBaseFile){
    if(
        _stprintf_s(FileNameCurrent.tBuf, FileNameCurrent.ArrLen, _T("L"%-s"
        defBinaryDBFileNameSuffix)
        ,TextDataBaseFile.FileNameCurrent.tBuf)<=0
        ||
        _stprintf_s(FullNameCurrent.tBuf, FullNameCurrent.ArrLen, _T("L"%-s\\"%-s"
        )
        ,Path.tBuf
        ,FileNameCurrent.tBuf)<=0
        ||
        _tfopen_s(&Stream, FullNameCurrent.tBuf, _T("wb"))!=0
        )
        return FALSE;
    else
        return TRUE;
}

BOOL CFileMultipleBinary::OpenFileForReading(ULONG WNStart, ULONG WNStop,
CSpectralCalcs* SpecClc){
    if(
        !CreateFullName(WNStart, WNStop, SpecClc)
        ||
        _tfopen_s(&Stream, FullNameCurrent.tBuf, _T("rb"))!=0
        )
        return FALSE;
    else
        return TRUE;
}

void CTextBufferLarge::Initialise(){
    tBuf[0]=0;
}
//
CTextBufferLarge::CTextBufferLarge(){
    Initialise();
}
//
void CTextBufferSmall::Initialise(){
    tBuf[0]=0;
}
//
CTextBufferSmall::CTextBufferSmall(){
    Initialise();
}
//
void CCounter::Initialize(){
    Files=FileLines=LineSymbols=FileTrans=FileTransInUse=TransTotal=TransTotalInUse=0;
    LinesTotal=0;
    //ULONG Files, FileLines, LineSymbols, FileTrans, FileTransInUse, TransTotal,
    TransTotalInUse;
    //double LinesTotal;
}

CCounter::CCounter(){
    Initialize();
}

CCLcTime::CCLcTime(){
    TimeIntervalInClocks=ct_Sec2Clocks(defRemTimeDisplayIntervalInSec);
    IsPossible=TRUE;
    GlobalTimeElapsed=0; // [seconds]
}

double CCLcTime::ct_Clocks2Sec(clock_t clocks){
    return (double)(clocks/CLOCKS_PER_SEC);
    //return (clock_t)(sec*CLOCKS_PER_SEC);
}

clock_t CCLcTime::ct_Sec2Clocks(double sec){

```

```

    return (clock_t)(sec*CLOCKS_PER_SEC);
}

BOOL CClcTime::ct_SetGlobalTimer(){
    if( (GlobalStart=clock()) == (clock_t)(-1) ){
        IsPossible=FALSE;
        return FALSE;
    }else
        return TRUE;
}

BOOL CClcTime::ct_SetTimeIntervalTimer(){
    if( (TimeIntervalStart=clock()) == (clock_t)(-1) ){
        IsPossible=FALSE;
        return FALSE;
    }else
        return TRUE;
}

BOOL CClcTime::ct_HasTimeIntervalElapsed(){
    if( (clock()-TimeIntervalStart)>=TimeIntervalInClocks )
        return TRUE;
    else
        return FALSE;
}

BOOL CClcTime::ct_GlobalTimeElapsed(){
    clock_t cl=clock();
    if( cl==(clock_t)(-1) ){
        IsPossible=FALSE;
        return FALSE;
    }
    else{
        GlobalTimeElapsed=ct_Clocks2Sec(cl-GlobalStart);
        return TRUE;
    }
}

BOOL CClcTime::ct_RemainingTimeForOutput(){
    RemainingTimeInSec+=defRemTimeDisplayAdvanceInSec;
    RemainingTime.ct_ConvFromSec(RemainingTimeInSec);

    if(
        time(&lTime)==(time_t)(-1)
        ||
        !(lTime=lTime+(time_t)(RemainingTimeInSec))
        ||
        !_tctime_s(BufFor_ctime.tBuf,BufFor_ctime.ArrLen,&lTime)
    ){
        IsPossible=FALSE;
        return FALSE;
    }
    else
        return TRUE;
}

void CTimeIntervalDD::ct_ConvFromSec(double sec){
    double
        minutes=sec/60,
        mmm=floor(minutes);
    SSS=(minutes-mmm)*60;

    double
        hours=mmm/60,
        hhh=floor(hours);
    MM=(mmm-hhh*60);

    double
        days=hhh/24;
    DD=floor(days);
    HH=(hhh-DD*24);
}

```

```
}
```

```
// File "Main.cpp"

#include <tchar.h>
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
#include <windows.h>
#include <time.h>

#include "Define.h"
#include "Header.h"
#include "Spectr.h"

using namespace std;

int _tmain(void) {
    WelcomeMessage(_T("
        L>Welcome to the "
        defDBName
        L" Application\nDTU Kemiteknik 2012\n"));

    CFileProcessing FileProc;
    CMonoxFileProcessing CMonoxFileProc;

    CSpectralCalcs
        SpecClc, //; //,
        CMonoxSpecClc;

    IsTimeToDestruct=FALSE;

    if(
        IsTimeToDestruct=FALSE
        ||
        !FileProc.MainInputFileProcessing(&SpecClc) // CO2 (Carbon Dioxide)
        ||
        !CMonoxFileProc.MainInputFileProcessing(&CMonoxSpecClc) // CO (Carbon Monoxide)
        ||
        !FileProc.WaveNumberFile.OpenFile(_T("rt")) // For a more clear error message if
the specified working directory can not be found
        ||
        !FileProc.WaveNumberFile.CloseFile() // and then it has to be closed for futher
proper use
        ||
        !FileProc.OutputFile.OpenFile(_T("wt"))
        ||
        !FileProc.InitializationOfSpecClc(&SpecClc) // CO2 (Carbon Dioxide)
        ||
        !CMonoxFileProc.InitializationOfCMonoxSpecClc(&CMonoxSpecClc, SpecClc) // CO
(Carbon Monoxide)
        ||
        !FileProc.PreliminaryOutput(SpecClc, CMonoxSpecClc)

        // CO2 (Carbon Dioxide)
        ||
        !FileProc.WaveNumberFileProcessing(&SpecClc) // calculating the number of values
for the wave number axis
        ||
        !FileProc.WaveNumberFileProcessing(&SpecClc) // allocating memory for and creat-
ing the wave number axis

        // CO (Carbon Monoxide)
        ||
```

```

        !FileProc.WaveNumberFileProcessing(&CMonoxSpecClc) // calculating the number of
values for the wave number axis
        ||
        !FileProc.WaveNumberFileProcessing(&CMonoxSpecClc) // allocating memory for and
creating the wave number axis

        // CO2 (Carbon Dioxide)
        ||
        !FileProc.TextDataBaseFileProcessing(&SpecClc)
        ||
        !FileProc.BinaryDataBaseFileProcessing(&SpecClc) // preliminary processing
        ||
        !SpecClc.XModelCreate(FileProc.OutputFile)
        ||
        !SpecClc.LineProfileWingCutoff(FileProc.OutputFile)
        ||
        !FileProc.BinaryDataBaseFileProcessing(&SpecClc) // primary processing

        //// CO (Carbon Monoxide)
        ||
        !FileProc.TextDataBaseFileProcessing(&CMonoxSpecClc)
        ||
        !FileProc.BinaryDataBaseFileProcessing(&CMonoxSpecClc) // preliminary processing
        ||
        !CMonoxFileProc.AxesCopyForCMonoxSpecClc(&CMonoxSpecClc, SpecClc, FileProc)
        ||
        !FileProc.BinaryDataBaseFileProcessing(&CMonoxSpecClc) // primary processing

        ||
        !SpecClc.sp_TauTrue_AbsTrue_Clc() // CO2 (Carbon Dioxide)
        ||
        !CMonoxSpecClc.sp_TauTrue_AbsTrue_Clc() // CO (Carbon Monoxide)

        // CO2+CO
        ||
        !CMonoxFileProc.AbsorbanceTrueSummation(&SpecClc,CMonoxSpecClc)
        ||
        !SpecClc.sp_ConvolutionWithILS(FileProc.OutputFile)
        ||
        !FileProc.Print_Tau_etc(&SpecClc, CMonoxSpecClc)
        ||
        !FileProc.OutputFile.CloseFile()
        ||
        !(IsTimeToDestruct=TRUE)
        ||
        !SpecClc.DeallocMem() // CO2 (Carbon Dioxide)
        ||
        !CMonoxSpecClc.DeallocMem() // CO (Carbon Monoxide)
    ){
        IsTimeToDestruct=TRUE;
        SpecClc.DeallocMem(); // CO2 (Carbon Dioxide)
        CMonoxSpecClc.DeallocMem(); // CO (Carbon Monoxide)
        EmergencyTerminationPrompt();
    }
    IsTimeToDestruct=TRUE;
    SpecClc.DeallocMem(); // CO2 (Carbon Dioxide)
    CMonoxSpecClc.DeallocMem(); // CO (Carbon Monoxide)

    MainExitPrompt();
    return 0;
}

```



# Appendix B.

## High-resolution transmission measurements of CO<sub>2</sub> at high temperatures for industrial applications

As mentioned in the objectives of the PhD project (Sec. 1.3.3), this appendix contains the journal article [17]

*Vadim Evseev, Alexander Fateev, Sønnik Clausen, High-resolution transmission measurements of CO<sub>2</sub> at high temperatures for industrial applications, Journal of Quantitative Spectroscopy and Radiative Transfer, Volume 113, Issue 17, November 2012, Pages 2222-2233, ISSN 0022-4073, <http://dx.doi.org/10.1016/j.jqsrt.2012.07.015>.*

included in this PhD thesis.

The article is written jointly by the PhD student (the first author) and the co-supervisor (*Alexander Fateev*) and the supervisor (*Sønnik Clausen*).

The transmission spectra of CO<sub>2</sub> were measured before the PhD project by *Alexander Fateev* and *Sønnik Clausen* at DTU Chemical Engineering (before 1 January 2012 Risø DTU, Optical Diagnostics Group). The high-temperature flow gas cell described in this article and in Refs. [2, 18, 19, 20] was also developed in the department before the PhD project. The first draft of the article presenting the high-temperature flow gas cell, the first calculations of the transmission spectra in comparison with the results of the measurements in the gas cell was developed by *Alexander Fateev* and *Sønnik Clausen* also before the PhD project.

The PhD student developed the software for the line-by-line calculations of the CO<sub>2</sub> transmission spectra (Sec. 3.1, Appendix A) which is able to use the most recent but huge CDSD-4000 database [16] (containing, e.g., for the 4.3 μm band of CO<sub>2</sub> almost 20 GBytes of the most recent high-temperature line-by-line data) such that it was possible to update the line-by-line calculations presented in the first draft of the article using the most recent high-temperature line-by-line data on CO<sub>2</sub>. The software is also able to calculate the transmission spectra of the mixture of CO<sub>2</sub> and CO which is important since CO<sub>2</sub> dissociates into CO (and O) at high temperatures. Thus the PhD student updated the line-by-line calculations of CO<sub>2</sub> at high temperatures and associated figures and discussion in the article.

It should be noted that Section “3.5. Energy exchange and thermal dissociation of CO<sub>2</sub> at  $T \geq 1473$  K” of the article as well as associated figures was developed and updated by *Alexander Fateev*.

The article starts from the next page and is attached in a form as it is published by Elsevier B.V. The article is followed by the joint author statements.



Contents lists available at SciVerse ScienceDirect

# Journal of Quantitative Spectroscopy & Radiative Transfer

journal homepage: [www.elsevier.com/locate/jqsrt](http://www.elsevier.com/locate/jqsrt)

## High-resolution transmission measurements of CO<sub>2</sub> at high temperatures for industrial applications

Vadim Evseev, Alexander Fateev\*, Sønnik Clausen

Department of Chemical and Biochemical Engineering, Technical University of Denmark (DTU Chemical Engineering), 2800 Kgs. Lyngby, Denmark

### ARTICLE INFO

#### Article history:

Received 20 March 2012

Received in revised form

11 July 2012

Accepted 12 July 2012

Available online 20 July 2012

#### Keywords:

High temperature

Carbon dioxide

Transmittance

FTIR spectroscopy

High-temperature flow gas cell

HITEMP

CDS

### ABSTRACT

High-resolution transmission spectra of CO<sub>2</sub> in the 2.7, 4.3 and 15 μm regions at temperatures up to 1773 K and at approximately atmospheric pressure (1.00 ± 0.01 atm) are measured and compared with line-by-line calculations based on the HITEMP-1995, HITEMP-2010, CDS-D-HITEMP and CDS-D-4000 databases. The spectra have been recorded in a high-temperature flow gas cell and using a Fourier transform infrared (FTIR) spectrometer at a nominal resolution of 0.125 cm<sup>-1</sup>. The volume fractions of CO<sub>2</sub> in the measurements were 1, 10 and 100%. The measurements have been validated by comparison with medium-resolution data obtained by Bharadwaj and Modest [6]. The deviations between the experimental spectra and the calculations at 1773 K and the vibrational energy exchange and thermal dissociation of CO<sub>2</sub> at high temperatures are discussed.

© 2012 Elsevier Ltd. All rights reserved.

### 1. Introduction

Measurements of gas temperature and CO<sub>2</sub> concentrations in various industrial environments (e.g., hot flue gases or flames) provide essential data for process optimization and the design of future environmentally friendly systems with reduced CO<sub>2</sub> emissions. CO<sub>2</sub> is one of the major products in various industrial processes, e.g., combustion. One way to measure the gas temperature and CO<sub>2</sub> concentrations in combustion systems is to compare optical spectral measurements with line-by-line calculations, which requires a knowledge of the spectroscopic parameters that are normally available from high-temperature databases, e.g., HITEMP [1,2] and CDS [3,4,5].

To develop and validate the high-temperature databases, high-resolution transmission measurements of CO<sub>2</sub>

at high temperatures are needed. A comprehensive review on this issue can be found in Ref. [6,7]. Since this review (2007), few studies have been reported. Cai et al. [8] performed spectral measurements of two line pairs of CO<sub>2</sub> and CO at low pressure in the temperature range 300–1000 K at 1.573 μm using a fiber-coupled distributed feedback diode laser. For the purpose of that study, a heated static cell with a total path length of 38 cm and an inner diameter of 3.5 cm was developed. The maximum temperature deviation along the path length was reported to be less than 1%. The motivation of the work was that the two line pairs could be used in a tunable diode laser absorption sensor for the simultaneous detection of CO<sub>2</sub> and CO in a single scan of the diode laser.

Depraz et al. [9,10] made emission measurements of CO<sub>2</sub> in the 2.7 and 4.3 μm regions at atmospheric pressure and temperatures up to 5000 K. A microwave discharge was used to produce the hot gas mixture, and the emission was recorded with a Fourier transform spectrometer at a spectral resolution between 0.01 and 0.1 cm<sup>-1</sup>. The plasma was confined inside quartz or sapphire tubes. The measurements were compared with calculations

\* Corresponding author. Tel.: +45 4677 4564; fax: +45 4677 4565.

E-mail addresses: vaev@kt.dtu.dk (V. Evseev), alfa@kt.dtu.dk (A. Fateev), sqcl@kt.dtu.dk (S. Clausen).

based on the spectroscopic CDSD-4000 and HITELOR databases. The former database has been reported to yield better agreement with the experimental data.

In this work, we present high-resolution transmission measurements of CO<sub>2</sub> in the 2.7, 4.3 and 15 μm regions at approximately atmospheric pressure ( $1.00 \pm 0.01$  atm) and temperatures up to 1773 K. The transmission spectra were recorded with a Fourier transform infrared (FTIR) spectrometer at a nominal spectral resolution of  $0.125 \text{ cm}^{-1}$ . The volume fractions of CO<sub>2</sub> in the measurements were 1, 10 and 100%. The measurements have been performed previously in a high-temperature flow gas cell [11–14] (the details are also presented here in Section 2). Line-by-line calculations were performed based on the spectroscopic HITEMP-1995 [1], HITEMP-2010 [2], CDSD-HITEMP [3] (which is the most recent version of CDSD-1000 [4]) and CDSD-4000 databases [5].

The transmission measurements have been validated by comparison with the medium-resolution ( $4 \text{ cm}^{-1}$ ) transmission spectra of CO<sub>2</sub> at 1550 K obtained by Bharadwaj and Modest [6]. To the best of our knowledge, these are the most recent experimental data available regarding CO<sub>2</sub> transmission at temperatures up to 1550 K and with a spectral resolution of  $4 \text{ cm}^{-1}$ .

Some experimental high-resolution CO<sub>2</sub> transmission spectra presented in this work were downgraded to a lower spectral resolution and used by Becher et al. [14] to validate various spectral gas radiation models, which would be used for thermal radiation heat transfer in oxyfuel combustion systems. In the work by Becher et al. [14], the transmission spectra were downgraded to a nominal resolution of  $32 \text{ cm}^{-1}$  and compared with line-by-line calculations based on the HITRAN-2004, HITRAN-2008, HITEMP-1995 and HITEMP-2010 databases, the two statistical-narrow-band models RADCAL and EM2C and the exponential-wide-band model [14].

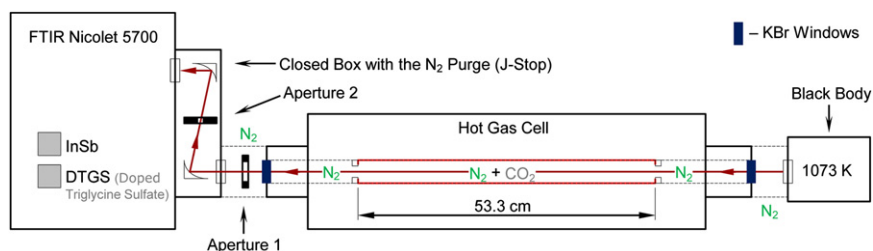
Deviations between the experimental spectra and the calculations at 1773 K as well as the vibrational energy exchange and the thermal dissociation of CO<sub>2</sub> at high temperatures are discussed.

## 2. Experimental details

The measurements were performed in our latest-generation atmospheric-pressure high-temperature flow gas cell (HGC) [11,14]. The gas cell was designed as a flow

gas cell with a so-called laminar window, nozzle seal cell principle, in which special care was taken to obtain a uniform gas temperature profile and a well-defined path length, similar to the other two gas cells with hot windows in our laboratory. The HGC consists of three different parts: a high-temperature sample cell with an effective length of 53.3 cm and two “buffer” hot-to-cold gas zones on both sides of the hot sample cell. The buffer zones are purged with a UV/IR-transparent (purge) gas (e.g., N<sub>2</sub>), whereas the central sample cell is filled with a sample gas (e.g., N<sub>2</sub> + CO<sub>2</sub>). The aperture of the sample cell is small (the diameter of the aperture is 1.5 cm) to reduce the loss of heat by radiation in the sample cell and to obtain a well-defined and stable gas flow in the laminar flow windows. The double apertures of the laminar flow windows also function as radiation shields. Similarly, extra apertures placed between the laminar flow windows and the cold windows of the buffer zones reduce the loss of heat by radiation and convection due to breaking down the vortices created by the thermal gradient in the buffer zones. High-quality pure ceramic (Al<sub>2</sub>O<sub>3</sub> (99.5%)) was used for the inner walls of the sample cell to minimize hetero-phase reactions and to avoid the contact of the sample gas with the hot metal parts. A uniform temperature profile was obtained by heating the sample cell with a three-zone furnace that compensates for the heat losses at the ends of the cell. The sample gas is preheated before it enters the hot sample cell. The gas flows in the sample cell and in the buffer zones are maintained at the same flow rates. The outer cold windows placed at the ends of the buffer zones are replaceable. In all of the experiments, KBr windows have been used. The windows allow spectral measurements in the region 0.25–25 μm. The gas flow through the sample cell is maintained with a high uniformity at a stable temperature in the range of 296–1873 K and at atmospheric pressure. The temperature uniformity along the axis of the sample cell has been verified using thermocouple measurements at three different points inside the sample cell. The temperature uniformity along the 45 cm path inside the sample cell was determined to be better than  $\pm 1$  K, with an average of  $\pm 0.5$  K. Further details on the HGC, its performance and a comparison with the other high-temperature flow gas cells in our laboratory will be presented in reference [12].

The optical set-up used for the CO<sub>2</sub> transmission measurements in this work is shown in Fig. 1. High-resolution



**Fig. 1.** Experimental set-up for the high-resolution transmission measurements of CO<sub>2</sub> at high temperatures. The central part of the hot gas cell (HGC) is marked with red. A representative IR ray emitted by the blackbody radiation source and passing through the HGC and J-stop is shown by the red line. All of the components of the optical set-up have been aligned in the horizontal and vertical planes.

(nominal resolution  $\Delta\nu=0.125\text{ cm}^{-1}$ ) IR-absorption measurements were performed using an FTIR spectrometer (Nicolet model 5700) with an external emission port and equipped with DTGS (Doped Triglycine Sulfate) and InSb detectors. A highly stable calibrated blackbody (BB) radiation source operating at 1073 K was used as an IR light source for the absorption and calibration measurements. After passing through the HGC, the IR light beam is restricted by Aperture 1, which varies from 10 to 20 mm in diameter (Fig. 1), to minimize the thermal radiation emitted from the surfaces in the HGC. Furthermore, the beam passes through a Jacquinot-stop (J-stop) system mounted on the outer part of the Nicolet spectrometer, which was operated in the external light source mode. Aperture 2 was set to a diameter of 1.4 mm to minimize the self-apodization effects. The FTIR spectrometer, J-stop, BB and all the interface parts between the BB-HGC and HGC-J-stop were purged with dried CO<sub>2</sub>-free air generated by a Balstron Purge Gas Generator.

In the experiments, the pre-mixed mixtures of N<sub>2</sub>+CO<sub>2</sub> (1%), N<sub>2</sub>+CO<sub>2</sub> (10%) and pure CO<sub>2</sub> (100%) were obtained from AirLiquid and were used at a flow rate of 2 l/min. The purity of N<sub>2</sub> in the gas cylinder was 99.1 vol%. The purity of CO<sub>2</sub> in the gas cylinder was also 99.1 vol%. N<sub>2</sub> (purity 99.998 vol%) at the same flow rate was used for the reference transmission measurements. The pressure in the HGC was measured to be approximately atmospheric, i.e.  $(1.00 \pm 0.01)$  atm.

The experimental data were recorded as interferograms,  $I(T)$ , where  $T$  [K] is the temperature of the gas mixture N<sub>2</sub>+CO<sub>2</sub> in the HGC. The “single beam” spectra,  $SB=SB(I(T))$ , were then calculated using the Nicolet software as an inverse (fast) Fourier transform of  $I(T)$  at a certain apodization function. A boxcar apodization function corresponding to the nominal resolution of  $0.125\text{ cm}^{-1}$  and the Mertz phase correction were used. No zero filling was applied.

The experimental uncertainties on transmittance measurements were estimated to be within 0.5% at a unity transmittance value.

### 3. Results and discussion

The transmission measurements were performed at temperatures of 1000, 1073, 1223, 1373, 1473, 1550, 1573 and 1773 K for the CO<sub>2</sub> volume concentrations of 1, 10 and 100%. Although the measurements were made at eight different temperatures, only the measurements at 1000, 1473, 1550 and 1773 K are discussed in this work.

#### 3.1. Data analysis

The CO<sub>2</sub> transmission spectra measured at temperatures of 1000, 1473, 1550 and 1773 K were compared with available medium-resolution data at 1550 K from Ref. [6] and with the calculated transmission spectra based on the HITEMP-1995 [1], HITEMP-2010 [2], CDSH-HITEMP [3] and CDSH-4000 [5] databases.

An experimental transmission spectrum  $\tau_{exp}(\nu, T)$  of CO<sub>2</sub> at a certain temperature  $T$  [K] in the HGC and at a

frequency  $\nu$  [ $\text{cm}^{-1}$ ] has been calculated as

$$\tau_{exp}(\nu, T) = \frac{SB(I_{\text{CO}_2 + \text{N}_2 + \text{BB}}(T) - I_{\text{CO}_2 + \text{N}_2}(T))}{SB(I_{\text{N}_2 + \text{BB}}(T) - I_{\text{N}_2}(T))} \quad (1)$$

where

$I_{\text{CO}_2 + \text{N}_2 + \text{BB}}(T)$  is the interferogram generated by the light coming from the HGC filled with the sample gas (N<sub>2</sub> and CO<sub>2</sub> mixture) at temperature  $T$  and a BB at 1073 K;  $I_{\text{CO}_2 + \text{N}_2}(T)$  is the signal only from the HGC filled with the sample gas at temperature  $T$  when the BB is blocked;  $I_{\text{N}_2 + \text{BB}}(T)$  is the signal from both the HGC filled with N<sub>2</sub> at temperature  $T$  and the BB at 1073 K; and  $I_{\text{N}_2}(T)$  is the light intensity only from the HGC filled with N<sub>2</sub> at temperature  $T$  with the BB blocked.

The calculations of the CO<sub>2</sub> transmission spectra,  $\tau_{cic}(\nu, T)$ , were performed assuming the Lambert-Beer law:

$$\tau_{cic}(\nu, T) = \exp(-\sigma_{\Delta P}^{eff}(\nu, T) l N), \quad (2)$$

where  $\sigma_{\Delta P}^{eff}(\nu, T)$  is the effective cross section for the series  $\Delta P=1, 3, 5$  in the notation of Ref. [4], [ $\text{cm}^2$ ];  $l$  is the absorption path length [cm]; and  $N$  is the molecular number density [ $\text{cm}^{-3}$ ] at a pressure of 1 atm and temperature  $T$ .

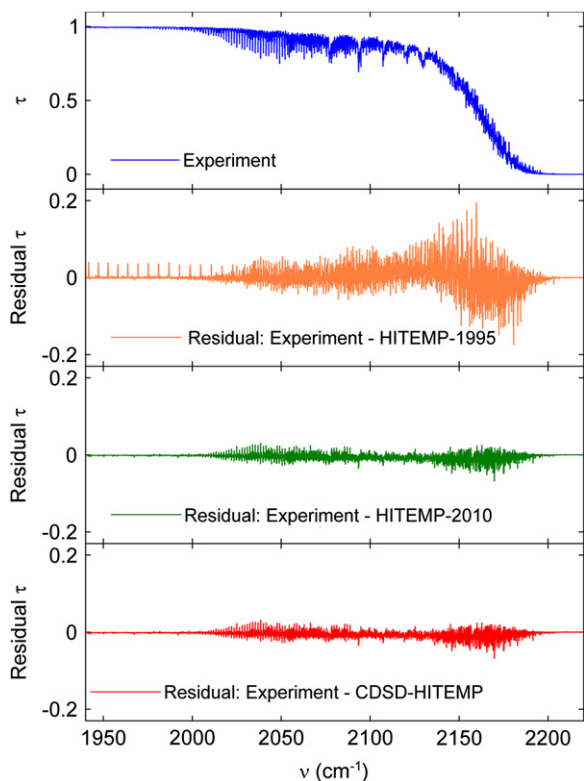
The effective cross section is calculated as the convolution of the true absorption cross section  $\sigma_{\Delta P}^{true}(\nu, T)$  with the instrument line-shape function (ILS)  $f(\nu - \nu_0)$  of the spectrometer

$$\sigma_{\Delta P}^{eff}(\nu_0, T) = -\frac{1}{l N} \ln \int_{\text{line}} f(\nu - \nu_0) \exp(-\sigma_{\Delta P}^{true}(\nu, T) l N) d\nu \quad (3)$$

The calculation of  $\sigma_{\Delta P}^{true}(\nu, T)$  requires knowledge regarding a) the integrated line intensities for the transitions  $\nu'', J'' \rightarrow \nu', J'$  between the vibrational levels of the two states involved in the absorption,  $S_{J'' J'}^{\nu'' \nu'}(\Delta P)$ , which are available from the databases (e.g., HITEMP-1995, HITEMP-2010, CDSH-HITEMP, CDSH-4000) for the series  $\Delta P=1$  (the 15  $\mu\text{m}$  band), 3 (the 4.3  $\mu\text{m}$  band) and 5 (the 2.7  $\mu\text{m}$  band); and b) the spectral distribution of  $S_{J'' J'}^{\nu'' \nu'}(\Delta P)$ , which is often assumed to be given by a Voigt profile as described in Ref. [15].

#### 3.2. High-resolution transmission measurements at 1000 K

Because the HITEMP-1995, HITEMP-2010 and CDSH-HITEMP databases were designed for a temperature of 1000 K, the measurements in the HGC were made at this temperature to demonstrate the performance of the HGC and to compare the high-resolution transmission measurements with the results of the calculations. The 4.3  $\mu\text{m}$  band of CO<sub>2</sub> (100%) is shown in Fig. 2 (lower frequency) and Fig. 3 (higher frequency). The HITEMP-2010 and CDSH-HITEMP databases both yield a better fit to the experiment than the HITEMP-1995 database does. This result is in agreement with the conclusion in Ref. [6] regarding HITEMP-1995 and CDSH-1000 (which is an older version of CDSH-HITEMP [3,4] used in Ref. [6]). The differences between the calculations based on HITEMP-2010 and CDSH-HITEMP are negligible at temperatures around 1000 K. In fact, the spectroscopic data for CO<sub>2</sub> in HITEMP-2010 were obtained from CDSH-HITEMP [2,3]. However, CDSH-HITEMP contains one spectroscopic parameter more, temperature-dependence

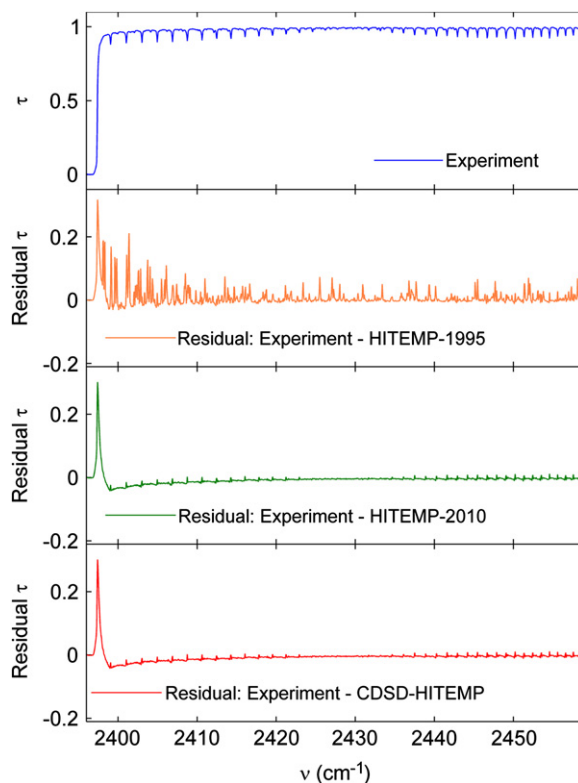


**Fig. 2.** The 4.3  $\mu\text{m}$  band of  $\text{CO}_2$  (100%) at 1000 K (lower frequencies, see also Fig. 3): measured at high resolution (1st from top, blue), residuals between the measured transmittance and HITEMP-1995 (2nd, orange), HITEMP-2010 (3rd, olive) and CDSH-HITEMP (4th, red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

exponent for self-broadened half-width, than HITEMP-2010 does. In the calculations using HITEMP-2010, this parameter was assumed to be equal to temperature-dependence exponent for air-broadened half-width. Nonetheless, differences between the calculations based on these two databases become more pronounced only at much higher temperatures and only in certain spectral regions as will be shown further in Section 3.4 (see Fig. 9).

Our experimental results are also in agreement with the data obtained previously [6] (see Fig. 8 in Ref. [6]) for  $\text{CO}_2$  (100%) at 1000 K,  $l=50$  cm and a resolution of  $4\text{ cm}^{-1}$ . In the range  $2000\text{--}2200\text{ cm}^{-1}$ , the CDSH-1000 (for Ref. [6])/CDSH-HITEMP (for this work) database reports slightly more absorption than the measurements do [6]. In the range  $2390\text{--}2420\text{ cm}^{-1}$ , the CDSH-1000 (for Ref. [6])/CDSH-HITEMP (for this work) database reports less absorption, but the rotational structure is the same as that of the experimental spectrum, as shown in Fig. 3. The same behavior can be observed in Fig. 8 of Ref. [6]. HITEMP-1995 is known to overestimate the absorption and to predict a more complex structure in this region when compared with the experimental data (see Fig. 3).

The  $2.7\text{ }\mu\text{m}$  band of  $\text{CO}_2$  (100%) is shown in Fig. 4. Again, the CDSH-HITEMP and HITEMP-2010 calculations both yield better agreement with the experimental spectrum than HITEMP-1995 does. The same phenomenon



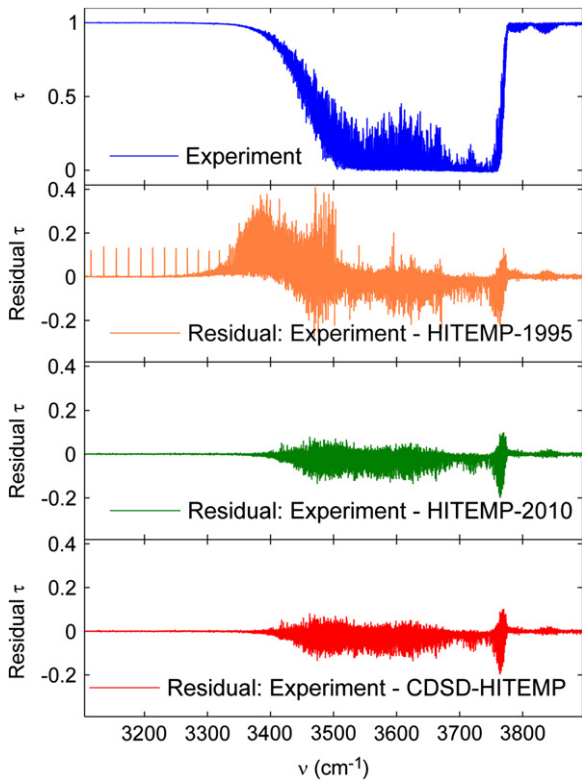
**Fig. 3.** The 4.3  $\mu\text{m}$  band of  $\text{CO}_2$  (100%) at 1000 K (higher frequencies): measured at high resolution (1st from top, blue), residuals between the measured transmittance and HITEMP-1995 (2nd, orange), HITEMP-2010 (3rd, olive) and CDSH-HITEMP (4th, red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with regard to comparison between HITEMP-1995 and CDSH-1000 was observed previously [6] (see Fig. 5 in Ref. [6]) for  $\text{CO}_2$  (100%),  $l=50$  cm and a resolution of  $4\text{ cm}^{-1}$ .

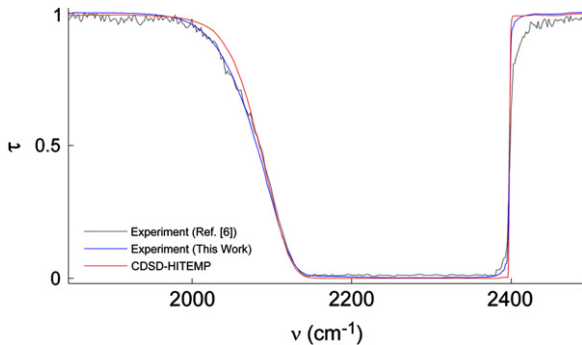
### 3.3. Direct comparison with the medium-resolution data from Ref. [6] at 1550 K

To our knowledge, high-temperature  $\text{CO}_2$  transmission measurements are only available at temperatures up to 1550 K and with a medium resolution [6]. Therefore, a direct comparison of our data scaled down to  $l=50$  cm and a resolution of  $4\text{ cm}^{-1}$  with the results obtained in Ref. [6] is of high interest. The 4.3 and  $2.7\text{ }\mu\text{m}$  bands of  $\text{CO}_2$  (100%) are shown in Figs. 5 and 6, respectively. It can be observed from these figures that there is a good agreement between the data from Ref. [6] and the current measurements except the region over  $2400\text{ cm}^{-1}$  in Fig. 5 where the transmittance is expected to be equal to unity which can be seen from both the CDSH-HITEMP calculation and the experiment of this work. However the experimental transmittance from Ref. [6] appears to be lower than unity which can probably be due to a wavelength-dependent base line offset in the experimental data of Ref. [6].

High-resolution measurements on the HGC have been made with the InSb detector and, therefore, are limited by

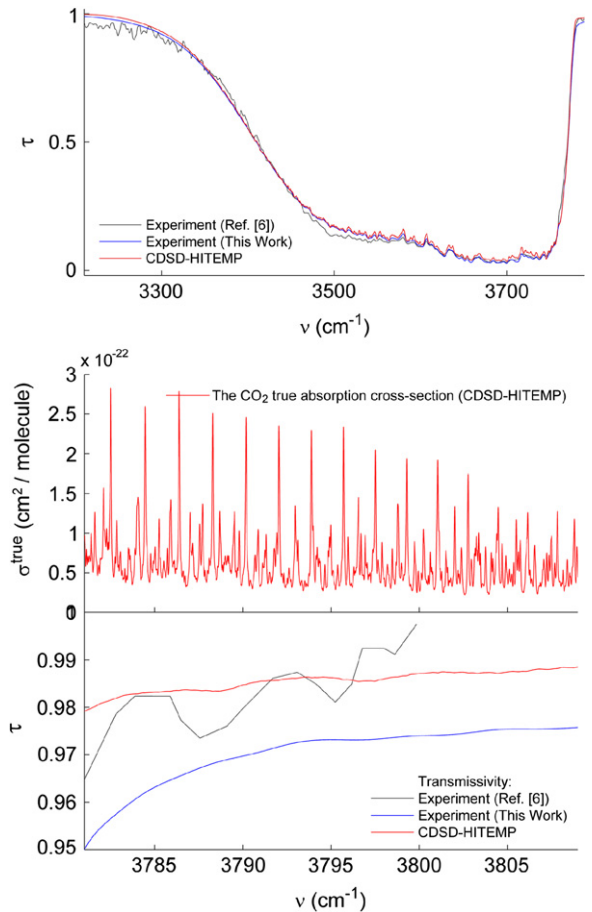


**Fig. 4.** The 2.7  $\mu\text{m}$  band of  $\text{CO}_2$  (100%) at 1000 K: measured at high resolution (1st from top, blue), residuals between the measured transmittance and HITEMP-1995 (2nd, orange), HITEMP-2010 (3rd, olive) and CDSD-HITEMP (4th, red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

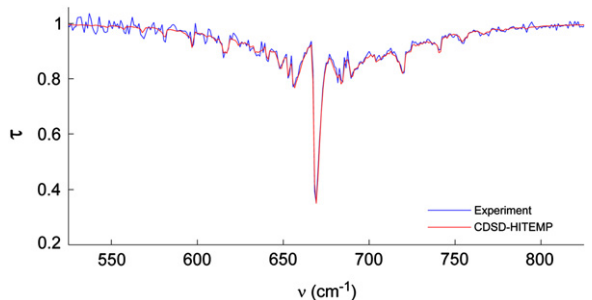


**Fig. 5.** The 4.3  $\mu\text{m}$  band of  $\text{CO}_2$  (100%) at 1550 K: measured in work [6] at medium resolution ( $\Delta\nu=4\text{ cm}^{-1}$ ) (black), measured in this work at high resolution and convoluted to  $\Delta\nu=4\text{ cm}^{-1}$  and corrected for  $l=50\text{ cm}$  (blue), CDSD-HITEMP calculation (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

the  $\Delta P=3$  and 5 bands of  $\text{CO}_2$ . To cover the  $\Delta P=1$  band, a DTGC detector has been used. In Fig. 7, a part of the experimental spectrum in the vicinity of the 15.0  $\mu\text{m}$  band of  $\text{CO}_2$  (1%) at 1473 K (measured with a resolution of  $2\text{ cm}^{-1}$ ) is shown together with the results of calculations based on the CDSD-HITEMP database. There is a very good agreement between the measurements and the

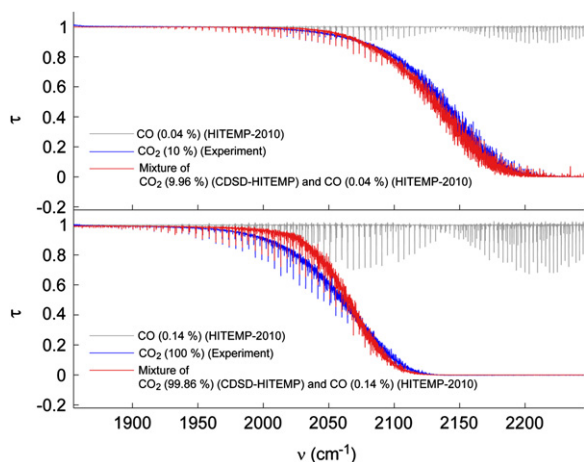


**Fig. 6.** The 2.7  $\mu\text{m}$  band of  $\text{CO}_2$  (100%) at 1550 K: measured in work [6] at medium resolution ( $\Delta\nu=4\text{ cm}^{-1}$ ) (black), measured in this work at high resolution and convoluted to  $\Delta\nu=4\text{ cm}^{-1}$  and corrected for  $l=50\text{ cm}$  (blue), CDSD-HITEMP calculation (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).



**Fig. 7.** The 15.0  $\mu\text{m}$  band of  $\text{CO}_2$  (1%) at 1473 K: measured at a resolution of  $2\text{ cm}^{-1}$  (blue), CDSD-HITEMP calculation (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

calculations. It should be noted that the calculations based on the HITEMP-1995 database yield results that are also in good agreement with the measurements. Fig. 13 in Ref. [6] compares the experimental data for the same  $\text{CO}_2$  (1%) band at 1550 K and a resolution of



**Fig. 8.** The  $4.3 \mu\text{m}$  band of  $\text{CO}_2$  (lower frequencies, see also Fig. 9) at 1773 K (in comparison with the CDS-D-HITEMP database) and the  $2143 \text{ cm}^{-1}$  band of CO. Upper panel: the HITEMP-2010 calculation of CO (0.04%) (gray, shown as a reference),  $\text{CO}_2$  (10%) measured at high resolution (blue), the calculation of the mixture of  $\text{CO}_2$  (9.96%) (CDS-D-HITEMP) and CO (0.04%) (HITEMP-2010) (red). Lower panel: the HITEMP-2010 calculation of CO (0.14%) (gray, shown as a reference),  $\text{CO}_2$  (100%) measured at high resolution (blue), the calculation of the mixture of  $\text{CO}_2$  (99.86%) (CDS-D-HITEMP) and CO (0.14%) (HITEMP-2010) (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$4 \text{ cm}^{-1}$  with the HITEMP-1995 and CDS-D-1000 databases. Some deviation between the experimental and calculated results can be observed from Fig. 13 in Ref. [6]. Our measurements were made at a temperature that is only 77 K lower than that used in Ref. [6]. In our opinion, the deviation in Ref. [6] cannot be explained only by uncertainties in the databases such as missing hot lines because both databases yield the same fit at 1550 K [6].

### 3.4. High-resolution transmission measurements at 1773 K

The  $4.3 \mu\text{m}$  band of  $\text{CO}_2$  at 1773 K and concentrations of 10 and 100% is shown in Fig. 8 (lower frequency) and Fig. 9 (higher frequency). The measurements are compared with the CDS-D-HITEMP as well as HITEMP-2010 (in Fig. 9 only) calculations. Another feature that can clearly be observed in Fig. 8 is a sharp structure in the  $1900\text{--}2100 \text{ cm}^{-1}$  region corresponding to absorption by CO due to the thermal dissociation of  $\text{CO}_2$  at 1773 K (See also Section 3.5). The  $2143 \text{ cm}^{-1}$  band of CO calculated based on HITEMP-2010 is shown by the gray line. The best fit calculations yield CO concentrations of 0.04 and 0.14% at 10%  $\text{CO}_2$  and 100%  $\text{CO}_2$ . The best fit calculations for CO were performed using the sharp CO absorption lines in the experimental absorbance in the region  $1925\text{--}1975 \text{ cm}^{-1}$  taking into account the  $\text{CO}_2$  absorption from the CDS-D-HITEMP calculations.

The calculation of  $\text{CO}_2$  transmittance at 1773 K (taking into account the thermal dissociation) is in reasonable agreement with the experimental results for the  $\text{CO}_2$  concentration of 10%. It should be noted that a change in the  $\text{CO}_2$  concentration due to its thermal dissociation

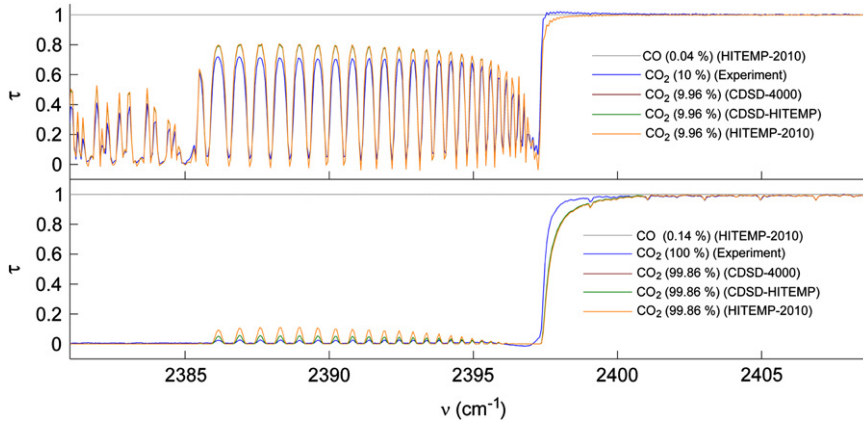
has a minor effect on the  $\text{CO}_2$  transmission spectra calculations.

For 100%  $\text{CO}_2$ , however, the deviations become significant in the  $1900\text{--}2100 \text{ cm}^{-1}$  and  $2385\text{--}2400 \text{ cm}^{-1}$  regions. The mismatch in the “oscillations” in the  $2386\text{--}2396 \text{ cm}^{-1}$  region (see Fig. 9) can be partly explained by the lack of some hot lines in all of the databases under consideration but, in particular, in HITEMP-2010 as can be seen in the lower panel of Fig. 9 (see also the discussion below regarding the calculations in this region). Nevertheless, it should be noted that this mismatch in the  $2386\text{--}2396 \text{ cm}^{-1}$  region has no effect on either the CO or the gas temperature measurements because 1) there are no CO absorption lines in this region, and 2) the transmission reaches zero at least in the  $2240\text{--}2320 \text{ cm}^{-1}$  region for 10%  $\text{CO}_2$ , which is sufficient for highly accurate gas temperature measurements on an industrial scale [16].

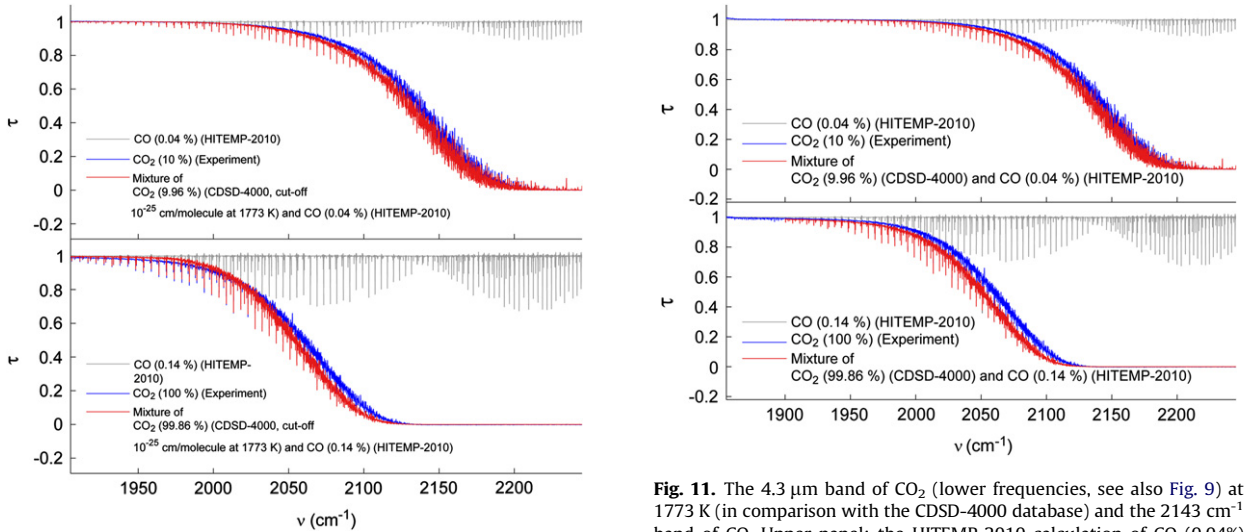
The deviations between the experiment and calculations at 1773 K can also be understood by the fact that this temperature is approximately twice as high as the reference temperature of the CDS-D-HITEMP (as well as HITEMP-2010) database, i.e., 1773 K is, perhaps, out of the valid temperature range for CDS-D-HITEMP (HITEMP-2010).

The measurements at 1773 K have been compared with the CDS-D-4000 [5] calculations (Figs. 9–11) of the  $\text{CO}_2$  transmittance (accounting for the thermal dissociation). It can be observed from Figs. 8, 10 and 11 that the long-wavelength wing of the  $4.3 \mu\text{m}$  100%  $\text{CO}_2$  band can be better described by the CDS-D-4000-based calculations than by the calculations based on CDS-D-HITEMP, particularly in the  $1970\text{--}2050 \text{ cm}^{-1}$  region (see Figs. 8 and 11). In this region, CDS-D-HITEMP underestimates the absorption of 100%  $\text{CO}_2$ , whereas there is good agreement for 10%  $\text{CO}_2$ . In contrast, at both  $\text{CO}_2$  concentrations, CDS-D-4000 overestimates the absorption along the long-wavelength wing of the  $4.3 \mu\text{m}$  band. This observation can be understood by the fact that the CDS-D-4000 database is designed for the significantly higher temperature range ( $2500\text{--}5000 \text{ K}$ ) than that used in the present work. CDS-D-4000 may contain a set of weak lines whose intensities ( $< 10^{-25} \text{ cm/molecule}$ ) can be incorrectly re-calculated from the higher temperatures used during the CDS-D-4000 generation (750, 2000 and 4000 K) relative to the reference temperature of the database (296 K). For this reason, as has been suggested by Dr. V.I. Perevalov, the calculations have also been performed with a line intensity cut-off of  $10^{-25} \text{ cm/molecule}$  at 1773 K (Fig. 10). In this case, the calculations are closer to the experiment, but there is still some underestimation and overestimation in the  $\text{CO}_2$  transmission compared with the measurements. This discrepancy can indicate that either the intensity extrapolation on the transitions from the highly excited states can be improved in CDS-D-4000 or another method should be used for (weak) line intensity calculations, as discussed in Section 3.5.

The underestimation by CDS-D-HITEMP may be explained by the inappropriate re-calculation of the intensities of the weak lines at 1773 K and by the lack of a weak line set, which, in fact, appears to contribute to the  $\text{CO}_2$  quasi continuum (see below).



**Fig. 9.** The 4.3  $\mu\text{m}$  band of  $\text{CO}_2$  (higher frequencies) at 1773 K (in comparison with the CDS-4000, CDS-HITEMP and HITEMP-2010 databases) and the 2143  $\text{cm}^{-1}$  band of CO. Upper panel: the HITEMP-2010 calculation of CO (0.04%) (gray, shown as a reference);  $\text{CO}_2$  (10%) measured at high resolution (blue); the CDS-4000 (bordeaux, in the figure the curve is completely overlapped by the CDS-HITEMP curve), CDS-HITEMP (olive, in the figure the curve is for the most part overlapped by the HITEMP-2010 curve) and HITEMP-2010 calculation of  $\text{CO}_2$  (9.96%) (orange). Lower panel: the HITEMP-2010 calculation of CO (0.14%) (gray, shown as a reference);  $\text{CO}_2$  (100%) measured at high resolution (blue); the CDS-4000 (bordeaux, in the figure the curve is completely overlapped by the CDS-HITEMP curve), CDS-HITEMP (olive) and HITEMP-2010 (orange) calculation of  $\text{CO}_2$  (99.86%). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).



**Fig. 10.** The 4.3  $\mu\text{m}$  band of  $\text{CO}_2$  at 1773 K (in comparison with the CDS-4000 calculations using a line intensity cut-off of  $10^{-25}$   $\text{cm}^2/\text{mole}$  at 1773 K) and the 2143  $\text{cm}^{-1}$  band of CO. Upper panel: the HITEMP-2010 calculation of CO (0.04%) (gray, shown as a reference),  $\text{CO}_2$  (10%) measured at high resolution (blue), the calculation of the mixture of  $\text{CO}_2$  (9.96%) (CDS-4000 with a line intensity cut-off of  $10^{-25}$   $\text{cm}^2/\text{mole}$  at 1773 K) and CO (0.04%) (HITEMP-2010) (red). Lower panel: the HITEMP-2010 calculation of CO (0.14%) (gray, shown as a reference),  $\text{CO}_2$  (100%) measured at high resolution (blue), the calculation of the mixture of  $\text{CO}_2$  (99.86%) (CDS-4000 with a line intensity cut-off of  $10^{-25}$   $\text{cm}^2/\text{mole}$  at 1773 K) and CO (0.14%) (HITEMP-2010) (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

The CDS-4000 and CDS-HITEMP databases yield an equally better agreement between the calculations and the measurements in the 2386–2396  $\text{cm}^{-1}$  region than the HITEMP-2010 database does, in particular, for 100%  $\text{CO}_2$ . It has already been mentioned that the differences in the calculations based on HITEMP-2010 and

**Fig. 11.** The 4.3  $\mu\text{m}$  band of  $\text{CO}_2$  (lower frequencies, see also Fig. 9) at 1773 K (in comparison with the CDS-4000 database) and the 2143  $\text{cm}^{-1}$  band of CO. Upper panel: the HITEMP-2010 calculation of CO (0.04%) (gray, shown as a reference),  $\text{CO}_2$  (10%) measured at high resolution (blue), the calculation of the mixture of  $\text{CO}_2$  (9.96%) (CDS-4000) and CO (0.04%) (HITEMP-2010) (red). Lower panel: the HITEMP-2010 calculation of CO (0.14%) (gray, shown as a reference),  $\text{CO}_2$  (100%) measured at high resolution (blue), the calculation of the mixture of  $\text{CO}_2$  (99.86%) (CDS-4000) and CO (0.14%) (HITEMP-2010) (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

CDS-HITEMP are negligible in many cases (see Section 3.2) but in the 2386–2396  $\text{cm}^{-1}$  region these databases yield significantly different results at 1773 K for 100%  $\text{CO}_2$ . HITEMP-2010 yields more underestimated absorption in this region than CDS-HITEMP (as well as CDS-4000) does. It has also already been mentioned Section 3.2 that CDS-HITEMP contains one spectroscopic parameter more, temperature-dependence exponent for self-broadened half-width,  $n_{self}$  [dimensionless], than HITEMP-2010 does [2,3]. In the calculations using HITEMP-2010, this parameter was assumed to be equal



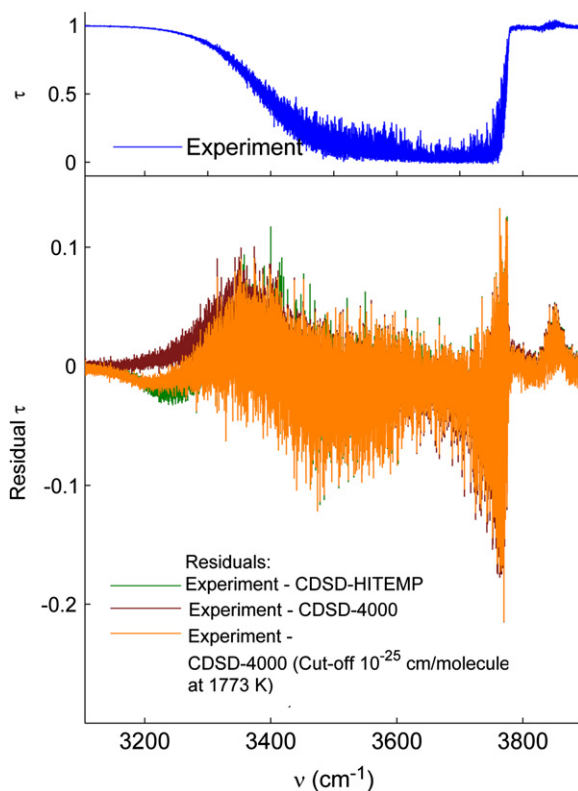
to temperature-dependence exponent for air-broadened half-width,  $n_{air}$  [dimensionless]. Considering files “02\_2250–2500\_HITEMP2010.par” (HITEMP-2010) and “cdsd\_hitemp\_2250\_2500.zip” (CSDS-HITEMP), which both contain the spectroscopic data for the 2250–2500  $\text{cm}^{-1}$  region, the maximum difference between  $n_{self}$  and  $n_{air}$  is 29% which occurs at the line number 4097. Also, self-broadened half-widths  $\gamma_{self}$  differ by 0.8% (this is the maximum difference and it occurs at the line number 268) between the two files. All the other spectroscopic data used in the calculations are identical in the two files which means that the differences between the two files were only observed for the two parameters relating to self-broadening of the spectral lines, which is, in fact, the major broadening mechanism at the concentration of 100%. These differences in the files contribute to the mismatch in the 2386–2396  $\text{cm}^{-1}$  region between the HITEMP-2010 and CSDS-HITEMP calculations. It should be noted that no significant differences were observed in the regions other than 2386–2396  $\text{cm}^{-1}$  between the HITEMP-2010 and CSDS-HITEMP calculations.

As regards the CSDS-4000 and CSDS-HITEMP calculations in the 2386–2396  $\text{cm}^{-1}$  region, the differences between the two databases in this region are negligible. However, even for these databases, some underestimation with respect to the experimental transmittance can be still observed (Fig. 9, lower panel).

It should be noted that varying the line-wing cut-off (from  $10^{-4}$  to  $10^{-6}$ ) for a Voigt line shape profile does not affect the calculations of the  $\text{CO}_2$  transmission spectra.

The measurements and calculations (CSDS-HITEMP and CSDS-4000) for the 2.7  $\mu\text{m}$  band of  $\text{CO}_2$  (100%) at 1773 K are shown in Fig. 12. It should be noted that the experimental transmittance curve reaches the level of almost 1.04 at 3850  $\text{cm}^{-1}$  which is caused by double-modulated light. Any minor imperfections in the alignment of optical system FTIR spectrometer+gas cell will cause double reflections on optical components (e.g. windows, mirrors or beamsplitter) and give double modulated signal on the detector of the FTIR spectrometer. This signal in fact is very low and can only be seen by a sensitive detector. As it was mentioned in the Section 2, the InSb detector was used in the FTIR spectrometer. An InSb detector is very sensitive and has a cut-off at about 1900  $\text{cm}^{-1}$ . Therefore double-modulated signal will give an artificial input to the observed spectra at wavenumbers greater than approximately 3800  $\text{cm}^{-1}$ . Use of a suitable filter will eliminate the problem.

Considering the long-wavelength wing of the 2.7  $\mu\text{m}$  band, deviations between the measurements and calculations result in the same trend as that observed for the case of the 4.3  $\mu\text{m}$  band discussed above. Thus, CSDS-HITEMP slightly underestimates the absorption in the range 3100–3280  $\text{cm}^{-1}$ , whereas CSDS-4000 slightly overestimates the absorption in the range 3200–3320  $\text{cm}^{-1}$ . Applying the line intensity cut-off of  $10^{-25}$   $\text{cm}/\text{molecule}$  to the CSDS-4000 data (Fig. 12, lower panel, orange curve) results in a spectrum that tends to be similar to that obtained from the CSDS-HITEMP calculations (particularly in the 3100–3280  $\text{cm}^{-1}$  range). In the 3320–3450  $\text{cm}^{-1}$  range, the CSDS-4000 and CSDS-HITEMP



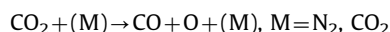
**Fig. 12.** The 2.7  $\mu\text{m}$  band of  $\text{CO}_2$  (100%) at 1773 K. Upper panel: measured at high resolution (blue). Lower panel: residuals between the measured transmittance and CSDS-HITEMP (olive), CSDS-4000 (bordeaux) and CSDS-4000 with a line intensity cut-off of  $10^{-25}$   $\text{cm}/\text{molecule}$  at 1773 K (orange). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

calculations yield similar results. In the 3450–3830  $\text{cm}^{-1}$  range, both databases are in equally good agreement with the experimental measurements.

### 3.5. Energy exchange and thermal dissociation of $\text{CO}_2$ at $T \geq 1473$ K

As described in the previous section, there are deviations between the experimental transmittance and the CSDS-HITEMP calculations at 1773 K, especially in the region 1970–2060  $\text{cm}^{-1}$  (See Fig. 8). The purpose of this section is to suggest and discuss a possible explanation for the deviation of the calculations from the experiment although other explanations may exist as well as to inspire a discussion about  $\text{CO}_2$  spectra calculations at high temperatures.

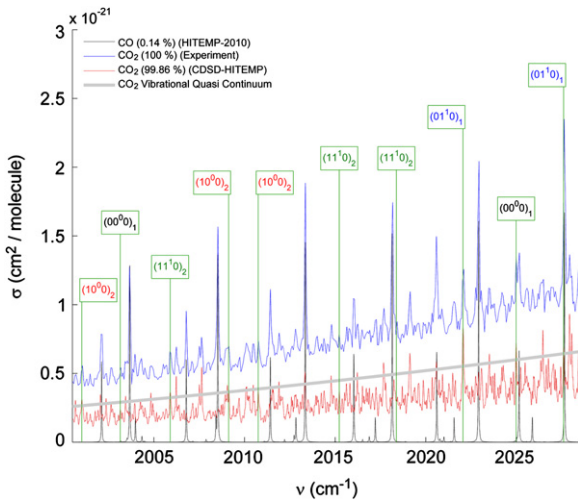
As was previously mentioned (see also Fig. 8), an experimental transmission spectrum at 1773 K in the 1900–2100  $\text{cm}^{-1}$  range consists of  $\text{CO}_2$  and CO bands. Because high-purity  $\text{Al}_2\text{O}_3$  (99.95%) ceramic is used in the inner part of the HGC, the main source of CO is the thermal dissociation of vibrationally excited  $\text{CO}_2$  in the gas phase



Although thermodynamic equilibrium calculations show that the thermal decomposition of CO<sub>2</sub> becomes significant at  $T > 2500$  K, some decomposition appears even at lower temperatures (Fig. 2 in Ref. [17]). High-resolution CO<sub>2</sub> (100%) transmission spectra measured in the HGC demonstrate the appearance of the CO absorption features at temperatures  $T \geq 1473$  K.

The adiabatic dissociation energy of CO<sub>2</sub> ( $56459 \text{ cm}^{-1}$ ) in the step-by-step vibrational excitation is quite high compared with the temperature of the experiment, which is  $1232 \text{ cm}^{-1}$  (1773 K). The products of CO<sub>2</sub> dissociation in this process are CO and oxygen atoms in the electronically excited state. Non-adiabatic transitions in the crossing points of the terms (e.g.,  $^1\Sigma^+ \rightarrow ^3B_2$ ) provide a more efficient dissociation process, which requires less energy ( $44360 \text{ cm}^{-1}$ ), followed by the formation of the products in the ground states [18].

The vibrational excitation of CO<sub>2</sub> in collisions with molecules (M) leads to a population of vibrational states with high  $(v_1 v_2^l v_3)_n$  numbers. Here,  $v$  indicates a vibrational quantum number,  $l$  is the vibrational angular momentum quantum number, and  $n$  is the sequence number of a Fermi resonance group. The magnified region ( $2000\text{--}2030 \text{ cm}^{-1}$ ) of the experimental CO<sub>2</sub> (100%) spectrum in Fig. 8 (lower panel) that was calculated as the effective absorption cross section (according to Eq. (2)) is shown in Fig. 13 (blue). The true CO<sub>2</sub> absorption cross section (CSDS-HITEMP, red) that was calculated according to Eq. (3) and the true CO absorption cross section (HITEMP-2010, scaled by 1/1000, black) are also shown in Fig. 13. The CSDS-HITEMP database was used for simplicity and to magnify the effect of the quasi continuum. Although the effective absorption cross section spectrum



**Fig. 13.** Part of the experimental CO<sub>2</sub> (100%) band from Fig. 8 calculated to the effective absorption cross section (blue,  $\Delta\nu=0.125 \text{ cm}^{-1}$ ), CO<sub>2</sub> (99.86%) true absorption cross section (red, CSDS-HITEMP,  $\Delta\nu=0.039 \text{ cm}^{-1}$ ), CO<sub>2</sub> vibrational quasi continuum (gray) and CO (0.14%) true absorption cross section (black, HITEMP-2010, scaled by 1/1000,  $\Delta\nu=0.025 \text{ cm}^{-1}$ ). The assignment of the transitions is shown by the vertical lines and the notation  $(v_1 v_2^l v_3)_n$  for the lower vibrational states involved in the transition. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

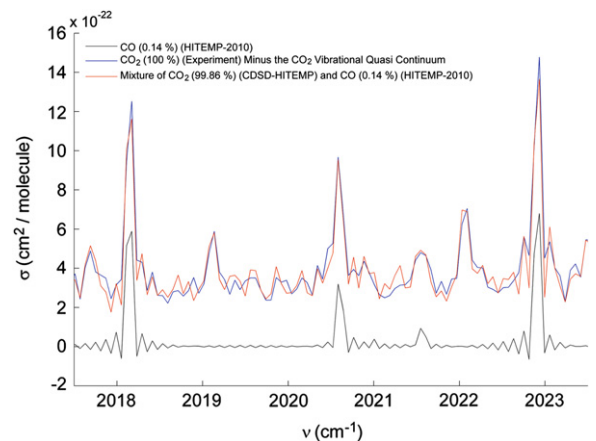
is broadened by the ILS, one can see that the majority of the experimental spectrum structure (e.g., line positions) is well described by the CSDS-HITEMP calculations. The major difference between the effective and true CO<sub>2</sub> absorption cross sections is that the former lies over a continuum-like structure, whereas the latter exhibits mostly flat behavior with a nearly constant offset due to line overlaps (Fig. 13). Subtraction of a continuum (Fig. 13, thick gray line) from the experimental spectrum yields a “corrected” effective cross section, as shown in Fig. 14 (blue). The calculated effective CO<sub>2</sub> cross section (CSDS-HITEMP) is shown by the red line. For convenience, the calculated effective CO cross section (HITEMP-2010) is also shown (black). There is a good agreement between the line intensities in the experimental and calculated absorption cross sections.

A continuum-like CO<sub>2</sub> absorption (Fig. 13, thick gray line) or, more accurately, a *vibrational quasi continuum* is a consequence of the vibrational excitation of CO<sub>2</sub> to the levels in the vicinity of its dissociation [19]. The population of highly excited states occurs due to vibrational (VV) relaxation. At low levels of vibrational excitation, the VV exchange occurs independently along the different modes. However, with an increased level of excitation, the vibrations of different types are collisionlessly mixed due to the intermode anharmonicity and the Coriolis interaction.

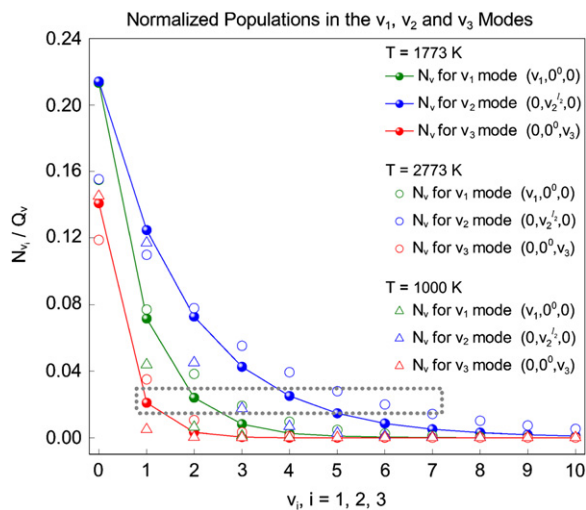
The intramolecular VV' exchange results in the vibrational quasi continuum of the highly excited states of CO<sub>2</sub> molecules. The energy of the asymmetric mode,  $\nu_3$ , changes due to the interaction of this mode with symmetric modes ( $\nu_1 + \nu_2 \approx \nu_3$ ) that leads to changes in the effective frequency of the  $\nu_3$  mode ( $\Delta\omega_3$ ) [20]

$$\Delta\omega_3 \approx \chi_{33}^{1/3} (A\omega_3 n_{sym})^{2/3}, \quad (4)$$

where  $A \approx 0.03\text{--}0.1$  is a dimensionless coefficient characteristic of the interaction between the  $\nu_3$  and  $\nu_1 + \nu_2$



**Fig. 14.** CO<sub>2</sub> (100%) effective cross section (blue) from Fig. 13 after subtraction of the CO<sub>2</sub> vibrational quasi continuum, calculated effective absorption cross section for the mixture of CO<sub>2</sub> (99.86%) (CSDS-HITEMP) and CO (0.14%) (HITEMP-2010) (red), the calculated CO (0.14%) effective absorption cross section (HITEMP-2010, scaled by 1/1000) is shown as a reference (black). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

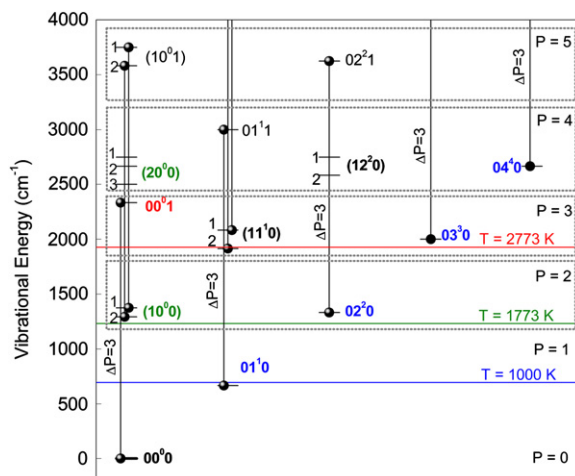


**Fig. 15.** Normalized (by the vibrational partition,  $Q_v$ ) Boltzmann distribution over the  $v_1$ ,  $v_2$  and  $v_3$  modes at  $T=1000$  K (triangles),  $T=1773$  K (beads) and  $T=2773$  K (open circles) and in  $\text{CO}_2$ . Energy levels have been calculated without  $l$ -splitting. A group of vibrational levels significantly involved in intramolecular  $VV$  energy exchange is included in the gray dotted rectangle. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

modes;  $\omega_3$  is an effective frequency of the asymmetric  $v_3$  mode;  $x_{33}$  is an anharmonicity coefficient for  $v_3$  mode; and  $n_{\text{sym}}$  is the total number of quanta in symmetric modes  $v_1 + v_2$ .

As the degree of  $\text{CO}_2$  excitation increases, i.e., the value of  $n_{\text{sym}}$  increases, the value of  $\Delta\omega_3$  increases and, at a certain critical number of quanta,  $n_{\text{sym}}^{\text{cr}}$ , becomes equal to the defect of resonance,  $\Delta\omega$ , of the asymmetric-to-symmetric quantum transition. When  $n_{\text{sym}} > n_{\text{sym}}^{\text{cr}}$ , the motion becomes quasi random, and the modes become mixed in the vibrational quasi continuum.

The normalized (on the vibrational partition sum  $Q_v$ ) Boltzmann vibrational distributions over the  $v_3$  ( $2396 \text{ cm}^{-1}$ ),  $v_1$  ( $1351 \text{ cm}^{-1}$ ) and  $v_2$  ( $672 \text{ cm}^{-1}$ ) modes at various temperatures are shown in Fig. 15. Because of the temperature levels  $v_i \geq 1$ ,  $i=1-3$  in the dotted gray rectangle result in additional absorption lines in the range  $2000-2030 \text{ cm}^{-1}$ . Some of these lines are clearly observed in Fig. 13 and are assigned to the transitions in the  $P=1$ , 2 and 3 manifolds. The absorption lines corresponding to transitions from  $v_2 \geq 2$  are weak. The energy level diagram of  $\text{CO}_2$  adopted from Ref. [21] is shown in Fig. 16. The levels that are significantly involved in the  $VV$  energy exchange are marked by bold green ( $v_1$  mode), blue ( $v_2$ ), red ( $v_3$ ) and black ( $v_1, v_2$  modes) colors, and they have the same color as their corresponding vibrational populations in Fig. 15. At 1000 K, apart from transitions from the (000) level, only the  $v_1=1$  and  $v_2 \leq 3$  levels can be considered to give rise to observed transitions in the range of interest with  $n_{\text{sym}}=5$ . If the temperature increases, the number of levels involved in the transitions is increased, and at 1773 K ( $n_{\text{sym}}=9$ ), transitions from  $v_1 \leq 2$  and  $v_2 \leq 5$  will give rise to the main contributions in the absorption spectra at  $\nu < 2050 \text{ cm}^{-1}$ . The vibrational levels in  $P=4$ , i.e., ( $20^0_0$ ), ( $12^2_0$ ) and ( $04^4_0$ )

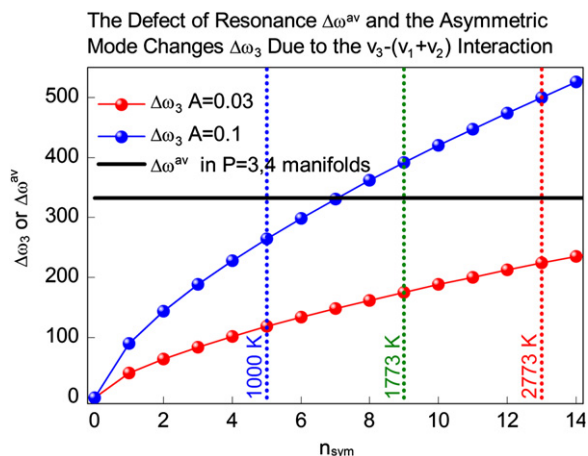


**Fig. 16.** Energy level diagram adopted from Ref. [21]. Each energy level is labeled as  $(v_1 v_2 v_3)_n$ , where  $v$  indicates a vibrational quantum number,  $l$  is the vibrational angular momentum quantum number and  $n$  is the sequence number of a Fermi resonance group. The vertical lines (black) show the first transitions within a  $4 \mu\text{m}$  band ( $\Delta P=3$ ). The polyads,  $P$ , are labeled from the right side, and the corresponding vibrational levels are included in the dotted boxes. The first vibrational levels for the  $v_1$ ,  $v_2$  and  $v_3$  modes are shown in the same colors as in Fig. 15 (dotted gray rectangles). The temperatures 1000, 1773 and 2773 K are indicated by blue, olive and red solid horizontal lines, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and  $P=3$ , i.e., ( $03^3_0$ ), begin to interact with the ( $00^0_1$ ) level and ultimately result in the formation of a weak vibrational quasi continuum. At 2773 K ( $n_{\text{sym}}=13$ ), the transitions from levels  $v_1 \leq 3$  and  $v_2 \leq 7(8)$  will cause the  $\text{CO}_2$  absorption spectrum to be more broad as a result of discrete vibrational transitions and a strong quasi continuum. The latter effect is caused by near-resonance between the ( $10^0_1$ )–( $02^2_1$ ) and close-to-resonance ( $10^0_1$ )–( $11^1_1$ ) levels in the  $P=5$  manifold and the close-to-resonance interaction of the ( $00^0_1$ ) with levels in  $P=3$ , 4 manifolds, as discussed above.

The asymmetric mode frequency changes in the  $v_3$  mode are shown in Fig. 17 at two different values of the  $A$  constant [20]. The average defect of the resonance between the  $v_3$  and  $v_1 + v_2$  modes in the  $P=3$  and 4 manifolds is shown by a solid black line. For the blue beads ( $A=0.1$ ), a difference of  $\Delta\omega^{\text{av}} - \Delta\omega_3 \approx 68 \text{ cm}^{-1}$  is reached at  $n_{\text{sym}}=5$  (1000 K), but  $\Delta\omega^{\text{av}} - \Delta\omega_3 \approx -57 \text{ cm}^{-1}$  at  $n_{\text{sym}}=9$  (1773 K). This result indicates that for the former case,  $n_{\text{sym}} < n_{\text{sym}}^{\text{cr}}$ , and for the latter case,  $n_{\text{sym}} > n_{\text{sym}}^{\text{cr}}$ , i.e., at 1773 K, the motion becomes quasi random and the modes become mixed in the vibrational quasi continuum. For the red beads ( $A=0.03$ ), the difference between  $\Delta\omega^{\text{av}}$  and  $\Delta\omega_3$  has not reached a resonance condition,  $\Delta\omega^{\text{av}} - \Delta\omega_3 \approx 0$ , even at  $n_{\text{sym}}=13$  (2773 K). Because  $\text{CO}_2$  thermal decomposition at 2773 K is approximately 20% [17], the  $A$  constant of 0.03 may be too small. Eq. (4) and Fig. 17 can be used to qualitatively explain the appearance of the  $\text{CO}_2$  vibrational quasi continuum.

Stochastic mixing of modes can be understood as a non-linear effect and can be taken into account by introducing



**Fig. 17.** The asymmetric mode frequency changes,  $\Delta\omega_3$ , due to the interaction with the symmetric modes ( $\nu_1 + \nu_2$ ) for two values of the  $A$  constant (red and blue beads) and the (averaged) defect of resonance,  $\Delta\omega^{av}$ , for the asymmetric-symmetric quantum transition  $\nu_3 \rightarrow \nu_1 + \nu_2$  for the  $(0_1 0_2 1_3)$  and  $(\nu_1 \nu_2 \nu_3)_n$  vibrational levels in the  $P=3, 4$  manifolds (black).  $n_{sym}$  is the total number of quanta in the symmetric modes,  $n_{sym} = 2n_1 + n_2$ . The vertical dotted lines correspond to  $n_{sym}$  at 1000 K (blue), 1773 K (olive) and 2773 K (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

a probability distribution function (in addition to the Boltzmann distribution) in the calculations of the absorption cross sections. Although this effect is small (as  $\text{CO}_2$  dissociation into  $\text{CO}$ ), it has to be taken into account during high-temperature, high- $\text{CO}_2$ -concentration spectra modeling. This effect is beyond the scope of the present work.

#### 4. Summary

High-resolution transmission measurements of  $\text{CO}_2$  in the 2.7, 4.3 and 15  $\mu\text{m}$  regions at atmospheric pressure ( $1.00 \pm 0.01$  atm) and temperatures of 1000, 1473, 1550 and 1773 K have been reported and compared to line-by-line calculations based on the spectroscopic databases HITEMP-1995 [1], HITEMP-2010 [2], CDSH-HITEMP [3] and CDSH-4000 [5]. These experimental measurements have also been compared to the medium-resolution transmission spectra of  $\text{CO}_2$  at 1550 K obtained by Bharadwaj and Modest [6].

A comparison of the measurements in the 2.7 and 4.3  $\mu\text{m}$  regions at 1550 K with the measurements of Bharadwaj and Modest [6] shows a very good agreement.

A comparison of the measurements in the 15  $\mu\text{m}$  region at 1473 K with the CDSH-HITEMP calculations shows a very good agreement, whereas the measurements of Bharadwaj and Modest [6] in the same region but at 1550 K show that both CDSH-1000 and HITEMP-1995 underestimate the absorption, but they generate spectra that are very similar to each other. The temperature difference of 77 K between the present measurements and the measurements in Ref. [6] is minor, and, in our opinion, the deviation between the calculations and the measurements in Ref. [6] cannot be explained only by

the uncertainties in the databases such as missing hot lines.

A comparison of the measurements in the 2.7 and 4.3  $\mu\text{m}$  regions at 1000 K with the HITEMP-1995-, HITEMP-2010- and CDSH-HITEMP-based calculations shows equally good agreement for HITEMP-2010 and CDSH-HITEMP, whereas HITEMP-1995 results in some deviations. The differences between HITEMP-2010 and CDSH-HITEMP are negligible at this temperature. Although spectroscopic data for  $\text{CO}_2$  in HITEMP-2010 was obtained from CDSH-HITEMP (see Refs. [2] and [3]) there are some differences between these databases which lead to some significant mismatch in the calculations at higher temperatures in some regions.

A comparison of the measurements in the 2.7 and 4.3  $\mu\text{m}$  regions at 1773 K with the CDSH-HITEMP and CDSH-4000 calculations shows some deviations for both databases in the long-wavelength wings of these bands. Namely, CDSH-HITEMP underestimates the absorption for both bands, which may be an indication of the lack of a weak line set that can also contribute to the  $\text{CO}_2$  quasi continuum, whereas CDSH-4000 overestimates the absorption. This result indicates that either intensity extrapolation of the transitions from the highly excited states should be improved in CDSH-4000 or another method should be used for the (weak) line intensity calculations (e.g., probability distribution function).

The central region of the 2.7  $\mu\text{m}$  band is equally well modeled with both the CDSH-HITEMP and the CDSH-4000 databases.

Deviations between the measurements and calculations at 1773 K have also been discussed in regard to the vibrational energy exchange and the thermal dissociation of  $\text{CO}_2$  at high temperatures.

#### Acknowledgments

The authors gratefully acknowledge Dr. V.I. Perevalov (Russia) and Dr. S.A. Tashkun (Russia) for their valuable discussions regarding the CDSH databanks and the spectra calculations as well as remarks concerning the draft of the paper. We are also grateful to Prof. M. Modest (USA) for providing us with the experimental data from Ref. [6] and to Prof. V.S. Sizikov (Russia) for valuable advice regarding the computational methods used in the spectra calculations. The hot gas cell was funded by energinet.dk, and the work on the high-resolution  $\text{CO}_2$  comparisons with calculations was partially funded by the DSF project "RADIADÉ".

#### Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at <http://dx.doi.org/10.1016/j.jqsrt.2012.07.015>.

#### References

- [1] Rothman LS, Wattson RB, Gamache RR, Schroeder J, McCann A. HITRAN, HAWKS and HITEMP high temperature database. Proc SPIE 1995;2471:105–11.

- [2] Rothman LS, Gordon IE, Barber RJ, Dothe H, Gamache RR, Goldman A, et al. HITEMP, the high-temperature molecular spectroscopic database. *J Quant Spectrosc Radiat Transfer* 2010;11:2139–50.
- [3] Tashkun SA, Perevalov VI. Readme for CDS-D-HITEMP [Internet]. Toms: Laboratory of Theoretical Spectroscopy, V.E. Zuev Institute of Atmospheric Optics [cited 2012 Jun 6]. Available from: <ftp://ftp.iao.ru/pub/CDS-D-HITEMP/Readme%20for%20cdsd-hitemp.pdf>.
- [4] Tashkun SA, Perevalov VI, Teffo JL, Bykov AD, NN, Lavrentieva. CDS-D-1000 the high-temperature carbon dioxide spectroscopy databank. *J Quant Spectrosc Radiat Transfer* 2003;82:165–96.
- [5] Tashkun SA, Perevalov VI. CDS-D-4000: high-resolution, high-temperature carbon dioxide spectroscopic databank. *J Quant Spectrosc Radiat Transfer* 2011;112:1403–10.
- [6] Bharadwaj SP, Modest MF. Medium resolution transmission measurements of CO<sub>2</sub> at high temperature—an update. *J Quant Spectrosc Radiat Transfer* 2007;103:146–55.
- [7] Modest MF, Bharadwaj SP. Medium resolution transmission measurements of CO<sub>2</sub> at high temperature. *J Quant Spectrosc Radiat Transfer* 2002;73:329–38.
- [8] Cai T, Gao G, Gao X, Chen W, Liu G. Diode laser measurement of line strengths and air-broadening coefficients of CO<sub>2</sub> and CO in the 1.57 μm region for combustion diagnostics. *Mol Phys* 2010;108:539–45.
- [9] Depraz S, Perrin MY, Soufiani A. Infrared emission spectroscopy of CO<sub>2</sub> at high temperature. Part I: experimental setup and source characterization. *J Quant Spectrosc Radiat Transfer* 2012;113:1–13.
- [10] Depraz S, Perrin MY, Rivière Ph, Soufiani A. Infrared emission spectroscopy of CO<sub>2</sub> at high temperature. Part II: experimental results and comparisons with spectroscopic databases. *J Quant Spectrosc Radiat Transfer* 2012;113:14–25.
- [11] Fateev A, Clausen S. Online non-contact gas analysis. Roskilde (Denmark): Risø National Laboratory, Technical University of Denmark; 2008 Mar. Report no.: Risø-R-1636(EN). Contract no.: Energinet.dk nr. 2006 1 6382.
- [12] Clausen S, Nielsen KA, Fateev A. Ceramic high temperature gas cell operating up to 1873 K. *Meas Sci Technol*, in preparation.
- [13] Fateev A, Clausen S. In situ gas temperature measurements by UV-Absorption Spectroscopy. *Int J Thermophys* 2009;30:265–75.
- [14] Becher V, Clausen S, Fateev A, Spliethoff H. Validation of spectral gas radiation models under oxyfuel conditions. Part A: gas cell experiments. *Int J Greenhouse Gas Control* 2011;5:S76–99.
- [15] Morrison Jr PW, Taweechokesupsin O. Calculation of gas spectra for quantitative Fourier transform infrared spectroscopy of chemical vapor deposition. *J Electrochem Soc* 1998;145:3212–9.
- [16] Bak J, Clausen S. FTIR emission spectroscopy methods and procedures for real time quantitative gas analysis in industrial environments. *Meas Sci Technol* 2002;13:150–6.
- [17] Yun SH, Kim GJ, Park DW. Decomposition and conversion of carbon dioxide into synthesis gas using thermal plasma. *J Ind Eng Chem* 1997;3:293–7.
- [18] Fridman A, Kennedy LA. Plasma physics and engineering. New York: Taylor & Francis; 2004. pp. 150, 314..
- [19] Capitelli M, editor. Berlin: Springer-Verlag; 1986. pp. 305–7.
- [20] Shuryak EV. Nonlinear resonance in quantum systems. *Sov Phys JETP* 1976;44:1070–80.
- [21] Bailly D. <sup>12</sup>C<sup>16</sup>O<sub>2</sub>: a set of spectroscopic constants from a simultaneous treatment of rovibrational bands at 4.5 and 15 μm. *J Mol Spectrosc* 1994;166:383–94.

January 2012

## Joint author statement

If a thesis contains articles made in collaboration with other researchers, a joint author statement about the PhD-student's part of the article shall be made by each of the co-authors, cf. article 12, section 4 of the Ministerial Order No. 18 February 2008 about the PhD degree

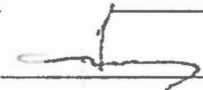
Title of the article: High-resolution transmission measurements of CO<sub>2</sub> at high temperatures for industrial applications

Author(s): Vadim Evseev, Alexander Fateev, Sønnik Clausen

Journal: Journal of Quantitative Spectroscopy and Radiative Transfer, Volume 113, Issue 17, November 2012, Pages 2222-2233, ISSN 0022-4073, <http://dx.doi.org/10.1016/j.jqsrt.2012.07.015>

PhD-student: Vadim Evseev CPR-no.: 240186-3187

Signature of the PhD-student: \_\_\_\_\_ Date: 31 October 2012

Co-author: Alexander Fateev Signature: 

### Description of each author's contribution to the above-mentioned article:

The article is written jointly by the PhD student (the first author) and the co-supervisor (Alexander Fateev) and the supervisor (Sønnik Clausen).

The transmission spectra of CO<sub>2</sub> were measured before the PhD project by Alexander Fateev and Sønnik Clausen at DTU Chemical Engineering (before 1 January 2012 Risø DTU, Optical Diagnostics Group). The high-temperature flow gas cell described in this article was also developed in the department before the PhD project. The first draft of the article presenting the high-temperature flow gas cell, the first calculations of the transmission spectra in comparison with the results of the measurements in the gas cell was developed by Alexander Fateev and Sønnik Clausen.

The PhD student developed the software for the line-by-line calculations of the transmission spectra of the CO<sub>2</sub>/CO mixture which is able to use the most recent but huge CDSD-4000 database such that it was possible to update the line-by-line calculations presented in the first draft of the article using the most recent high-temperature line-by-line data on CO<sub>2</sub>. Thus the PhD student updated the line-by-line calculations of CO<sub>2</sub> at high temperatures together with the associated figures and discussion. Section "3.5. Energy exchange and thermal dissociation of CO<sub>2</sub> at T ≥ 1473 K" of the article as well as the associated figures was developed and updated by Alexander Fateev.

January 2012

## Joint author statement

If a thesis contains articles made in collaboration with other researchers, a joint author statement about the PhD-student's part of the article shall be made by each of the co-authors, cf. article 12, section 4 of the Ministerial Order No. 18 February 2008 about the PhD degree

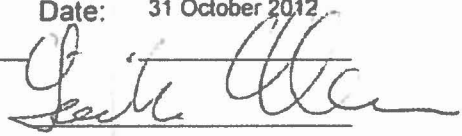
Title of the article: High-resolution transmission measurements of CO<sub>2</sub> at high temperatures for industrial applications

Author(s): Vadim Evseev, Alexander Fateev, Sønnik Clausen

Journal: Journal of Quantitative Spectroscopy and Radiative Transfer, Volume 113, Issue 17, November 2012, Pages 2222-2233, ISSN 0022-4073, <http://dx.doi.org/10.1016/j.jqsrt.2012.07.015>

PhD-student: Vadim Evseev CPR-no: 240186-3187

Signature of the PhD-student: \_\_\_\_\_ Date: 31 October 2012

Co-author: Sønnik Clausen Signature: 

### Description of each author's contribution to the above-mentioned article:

The article is written jointly by the PhD student (the first author) and the co-supervisor (Alexander Fateev) and the supervisor (Sønnik Clausen).

The transmission spectra of CO<sub>2</sub> were measured before the PhD project by Alexander Fateev and Sønnik Clausen at DTU Chemical Engineering (before 1 January 2012 Risø DTU, Optical Diagnostics Group). The high-temperature flow gas cell described in this article was also developed in the department before the PhD project. The first draft of the article presenting the high-temperature flow gas cell, the first calculations of the transmission spectra in comparison with the results of the measurements in the gas cell was developed by Alexander Fateev and Sønnik Clausen.

The PhD student developed the software for the line-by-line calculations of the transmission spectra of the CO<sub>2</sub>/CO mixture which is able to use the most recent but huge CDSD-4000 database such that it was possible to update the line-by-line calculations presented in the first draft of the article using the most recent high-temperature line-by-line data on CO<sub>2</sub>. Thus the PhD student updated the line-by-line calculations of CO<sub>2</sub> at high temperatures together with the associated figures and discussion. Section "3.5. Energy exchange and thermal dissociation of CO<sub>2</sub> at T ≥ 1473 K" of the article as well as the associated figures was developed and updated by Alexander Fateev.