



LIISim: a modular signal processing toolbox for laser-induced incandescence measurements

Raphael Mansmann¹ · Tobias Terheiden¹ · Philip Schmidt¹ · Jan Menser¹ · Thomas Dreier¹ · Torsten Endres¹ · Christof Schulz¹

Received: 25 January 2018 / Accepted: 26 March 2018 / Published online: 4 April 2018
© Springer-Verlag GmbH Germany, part of Springer Nature 2018

Abstract

Evaluation of measurement data for laser-induced incandescence (LII) is a complex process, which involves many processing steps starting with import of data in various formats from the oscilloscope, signal processing for converting the raw signals to calibrated signals, application of models for spectroscopy/heat transfer and finally visualization, comparison, and extraction of data. We developed a software tool for the LII community that helps to evaluate, exchange, and compare measurement data among research groups and facilitate the application of this technique by providing powerful tools for signal processing, data analysis, and visualization of experimental results. A common file format for experimental data and settings simplifies inter-laboratory comparisons. It can be further used to establish a public measurement database for standardized flames or other soot/synthetic nanoparticle sources. The open-source concept and public access to the software development should encourage other scientists to validate and further improve the implemented algorithms and thus contribute to the project. In this paper, we present the structure of the LIISim software including the materials database concept, signal-processing algorithms, and the implemented models for spectroscopy and heat transfer. With two application cases, we show the operation of the software how data can be analyzed and evaluated.

1 Introduction

1.1 Laser-induced incandescence

Laser-induced incandescence (LII) is well-established for in situ measurements of gas-borne soot particles [1–3], which is also used in an increasing number of applications for the investigation of gas-borne inorganic nanoparticles [4–13]. In LII, particles are heated by a pulsed laser to incandescent temperatures (i.e., higher than 2000 K). The magnitude of the incandescence at a given heat-up temperature scales with the particle volume fraction in the detection volume and the temporal variation of the particle temperature

during cooling can be exploited to derive information about the particle size (as small particles cool more quickly than larger ones). The latter approach is commonly called time-resolved LII (TiRe-LII). Both the integral signal intensity and the temporal decay strongly depend on the heat-up temperature of the particles as well as the temperature of the bath gas. The temperature of the laser-heated particles is commonly determined by pyrometry. Often, two-color pyrometry is used based on the time-resolved signal measured with two fast photomultipliers [14–16] assuming black- or grey-body characteristics of the particles. If more than two detection wavelengths are available (either through the combination of more than two detectors with respective bandpass filters [17, 18] or through detection with spectrometers and streak-camera systems [11, 12, 17, 19]), the measured intensities can be related to spectroscopic models that can incorporate a more complex spectral response.

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s00340-018-6934-9>) contains supplementary material, which is available to authorized users.

✉ Raphael Mansmann
raphael.mansmann@uni-due.de

1.2 Models

In LII history, model development is an ongoing process and many research groups have been using their own models. It is a common practice to relate the measured (electrical)

¹ Institute for Combustion and Gas Dynamics - Reactive Fluids (IVG), Center for Nanointegration Duisburg-Essen (CENIDE), University of Duisburg-Essen, Duisburg, Germany

signal from a fast detector to another physical quantity using a model that describes the signal generation and detection based on physical principles. This “spectroscopic model” usually consists of a temperature-dependent equation derived from Planck’s law corrected for deviation from blackbody behavior, which relates the measured signal to a temperature. In the early stage of LII, only the normalized signal of a single channel was related to that equation and the absorption of laser energy was used to estimate the LII peak temperature [20]. Later, two detectors were used to increase the robustness of the measurements using two-color pyrometry for the determination of the LII peak temperature [15, 21] or the complete temperature trace [7]. Most recent studies use three or more detectors [17, 18] or streak cameras [11, 12, 16] and apply spectral fitting. The result from the spectroscopic model is usually a temperature trace during and after the laser pulse, which can then be interpreted with a “heat-transfer model”. The first effort of model standardization was made in 2007 by ten research groups that published a comparison of their models [22]. All LII models share the same energy- and mass-balance equations, but the submodels describing the heat transfer phenomena and the selected physical properties varied widely.

1.3 Software

For the processing of LII signals, usually unpublished source code is used by each research group. They target various purposes. Some software allows familiarizing students and beginning researchers with the complex dependences of LII signals—often with the opportunity to select competing models and submodels. Additionally, there are several in-house codes that are often not published in detail. In a few cases, models (with limited documentation) are incorporated in commercial devices. Within the LII-workshop series (e.g., [23–25]) there has been a continuous effort to provide model data sets to compare the models in a series of test cases with increasing hierarchy [1, 22]. This led to the desire to make consolidated models available to the entire community. We are aware of two such approaches: (i) The Community Laser-Induced Incandescence Modeling Environment (CLiME) project, started in 2013, aimed at providing a collaborative model infrastructure for researchers. The concept was based on an agile software development process between scientists and scientific software developers [26, 27]. After the release of the open-source software, different research groups should have been able to extend the CLiME’s functionalities by adding physical models based on new available data. The software is based on FORTRAN and Java; at the time of writing this manuscript, it has not yet been published. (ii) **LIISim** has been introduced in 2001 as a console application (LIISim 1.5) and has been provided via a web interface (<http://web.liisim.com>) [28]. The console

application was written in C and the web interface was based on Perl. Modeling settings and file names for experimental data could be defined in DAT files and after execution of the console application, DAT output files containing the modeling results are created. Later, the console application was extended by a C++-based graphical user interface (GUI) by Tobias Terheiden and Martin Leschowski that allows also the visualization of the modeling results from the DAT files (LIISim Desktop 1.02 and Console 2.14). While the console application was distributed among the community, the desktop version was not published and was only internally used. The software was previously used for soot diagnostics on laminar premixed high-pressure flames [29–32], on atmospheric pressure laminar non-premixed flames [33] and at Diesel engine conditions [34]. The LIISim web interface has been referenced in recent publications [35, 36].

The new version of LIISim 3.0 presented in this paper, builds on the previous experience but follows a fundamentally different approach: data analysis in the past has been a static and iterative process starting with modification and execution of processing settings following processing, visualization, interpretation, and extraction of results. The new approach is based on a dynamic evaluation concept using a graphical user interface and modular tools for processing and visualization to get a deeper insight into the data. While in previous versions the processing chain and physical properties were hard-coded, the new concept allows flexible combination and configuration of processing steps within the graphical user interface. Physical properties are now not limited to a single materials system anymore (i.e., soot), but can be provided as any constant or temperature-dependent equation through standardized text files. This allows easy switching between materials systems while keeping the same processing chain. This opens the way for various synthetic nanoparticle applications, such as already available for silicon [10, 12]. Researchers could provide their experimental data from their publications in the LIISim-compatible format including various meta-data (i.e., laser fluence, process pressure, measurement position, etc.). This would be the foundation for a public reference measurement database for standardized flames or synthetic nanoparticle applications, which will simplify the comparison of data and evaluation strategies among research groups. Particularly with regard to the biennial LII workshop, the LIISim format could be used for exchange of data for inter-laboratory comparisons. The new LIISim source code is published under the GNU General Public License (GPL) to provide transparency in all implemented algorithms and allows other researchers to review and modify the source code or add new heat-transfer models.

In this paper, the new LIISim software is presented and its application is demonstrated by two test cases. First, we discuss the implemented spectroscopic and heat-transfer

models, then the software architecture, data structure, and materials database concept are explained. After describing the main modules for signal processing, data analysis and fitting, we show some of the functionalities by using a published validation data set [17] and recently measured data using a standardized laminar diffusion flame.

2 Models implemented in LIISim

Signal processing of experimental LII data can be separated in three consecutive steps: (i) basic signal processing (i.e., baseline correction, calibration, simple arithmetics, etc.); (ii) spectroscopic modeling for calculation of temperature trace from signal data, and (iii) heat-transfer modeling for fitting modeled temperature traces to the experimentally determined temperature traces.

2.1 Spectroscopic models

For the calculation of temperature traces, two methods are implemented in LIISim: two-color pyrometry and spectral fitting of Planck’s law. The two-color pyrometry is commonly used in recent studies [7, 15, 21, 37] and requires calibrated LII signals at two independent wavelengths:

$$T_p = \frac{h c_0}{k_B} \left(\frac{1}{\lambda_2} - \frac{1}{\lambda_1} \right) \left[\ln \left(\frac{I_\lambda(\lambda_1, T_p) E(m_{\lambda_2})}{I_\lambda(\lambda_2, T_p) E(m_{\lambda_1})} \left(\frac{\lambda_1}{\lambda_2} \right)^6 \right) \right]^{-1}, \tag{1}$$

where T_p is the particle temperature, h the Planck’s constant, c_0 the speed of light in vacuum, k_B the Boltzmann constant, λ_i the bandpass center-wavelength of the detector, I_λ the calibrated intensity, and $E(m_\lambda)$ the absorption function. It needs to be noted that this formula is derived using the Wien approximation and differentiates in this aspect from a spectral fitting.

When two or more spectral detection channels are available, a spectral fit including all channels can be performed. For this analysis, Planck’s law corrected for deviation from blackbody behavior is fitted to the data points of all selected channels. A detailed description of the equation can be found in [38]:

$$I(\lambda, T_p) = f_V G_\lambda C_{\lambda, \text{abs}} \frac{2hc_0^2}{\lambda^5} \left[\exp \left(\frac{hc_0}{k_B \lambda T_p} \right) - 1 \right]^{-1}, \tag{2}$$

where f_V is the volume fraction during the given laser shot, G_λ scales the emitted intensity based on the geometry and optical efficiencies of the detector, and $C_{\lambda, \text{abs}}$ is the wavelength-dependent absorption cross-section given by the Rayleigh limit of Mie theory [39]:

$$C_{\lambda, \text{abs}}(d_p) = \frac{\pi^2 d_p^3 E(m_\lambda)}{4 \lambda}. \tag{3}$$

In LIISim, wavelength-independent properties of Eqs. (2) and (3) are summarized in a general scaling factor C , which simplifies Eq. (2) to:

$$I(\lambda, T_p) = C \frac{E(m_\lambda)}{\lambda} \frac{2hc_0^2}{\lambda^5} \left[\exp \left(\frac{h c_0}{k_B \lambda T_p} \right) - 1 \right]^{-1}. \tag{4}$$

2.2 Heat-transfer models

The LII technique is based on particle heating by a pulsed laser and the investigation of the subsequent energy transfer to the surroundings. The energy transfer rates depend on the particle size and can be incorporated in heat-transfer models. The particle diameter can then be used as a fitting parameter and the residual between the simulated and the measured temperature decay is minimized in a least-square fitting algorithm.

For determination of particle sizes, various heat-transfer models from literature are implemented that share the same equations for the energy- and mass balance. The individual heat transfer rates from published models are described in “Appendix B”. For the description of particle cooling, the following energy balance is used:

$$\frac{d}{dt} (m_p c_p T_p) = -\dot{Q}_{\text{evap}} - \dot{Q}_{\text{cond}} - \dot{Q}_{\text{rad}}, \tag{5}$$

where m_p is the particle mass, c_p the specific heat capacity, and T_p the particle temperature. \dot{Q}_{evap} , \dot{Q}_{cond} , and \dot{Q}_{rad} represent cooling rates for heat conduction, evaporation, and radiation, respectively. These cooling rates are re-implemented in each individual heat-transfer model. It needs to be noted that the LIISim energy balance is limited to particle cooling and does not include heat transfer by laser absorption. The mass balance is simplified to

$$\frac{d m_p}{dt} = -\dot{m}_{\text{evap}}, \tag{6}$$

which includes only \dot{m}_{evap} , the mass loss due to evaporation, which is also re-implemented in the individual heat-transfer models. Michelsen et al. [22] summarized different heat-transfer models and the physical properties used by various research groups. The following models from this publication are implemented: “Kock”, “Liu” and “Melton (workshop)”. A detailed description of the implemented models and the corresponding physical properties can be found in the “Appendix B”.

3 Implementation

3.1 Software architecture

The LIISim software is written in object-oriented C++ using the Qt-framework (5.5.0) [40] for the graphical user interface (GUI). For mathematical operations like ordinary differential equation (ODE) stepper and complex data container, the Boost library (1.55) [41] and for visualization of data the Qwt library (6.1.3) [42] was used. The source code can be compiled for 64-bit computer architecture to overcome the 4-GB memory limitation or in case older computers are used for 32-bit architecture. Multi-threading using more than one CPU kernel at a time is implemented for parallel signal processing and fitting, which reduces processing time.

In addition to the C++ software, a model development kit (MDK) was written in MATLAB for comparison and validation of the integrity of the implemented algorithms. For the MDK the same class structure is used and the functionalities are limited to simulation of temperature traces using the previously discussed heat-transfer models. The MDK can be used for testing new models before they are integrated in the LIISim framework. The source code of LIISim-MDK can be found on the LIISim project page and is not discussed further.

3.2 Data structure

The import of LII signals supports comma-separated values (CSV) or plain text files. File structure and file names

are documented within the import tools. For data export, additionally the MATLAB file format can be chosen. The data visualized in every plot can be copied to the clipboard and can so be directly used in any spreadsheet—compatible software. Experimental data are stored in a hierarchical data model, which is shown in Fig. 1. Three data container classes (i.e., MRun, MPoint, and Signal) connect the single LII signal traces. MRun represents a series of LII signals collected at identical experimental conditions (i.e., laser fluence, PMT gain, flame position, etc.). MPoint stores all signals related to the same laser shot. This means for the same laser shot there could be a “Raw Signal”, a calibrated “Absolute Signal”, and the “Temperature Trace” calculated from the respective signals.

3.3 Materials database

Physical properties for materials or gases are usually defined in the source code prior to execution. In LIISim, a modular database approach was chosen that allows to change or to switch properties during runtime. An unlimited number of user-defined database objects (material, gas, gas mixtures, LII settings) can be saved within plain text files that are automatically imported from a database folder. The properties can be defined as constant or as functions of temperature or wavelength according to Table 1. The database files can then be selected and combined for heat-transfer modeling. The most recent list of available properties and guidelines

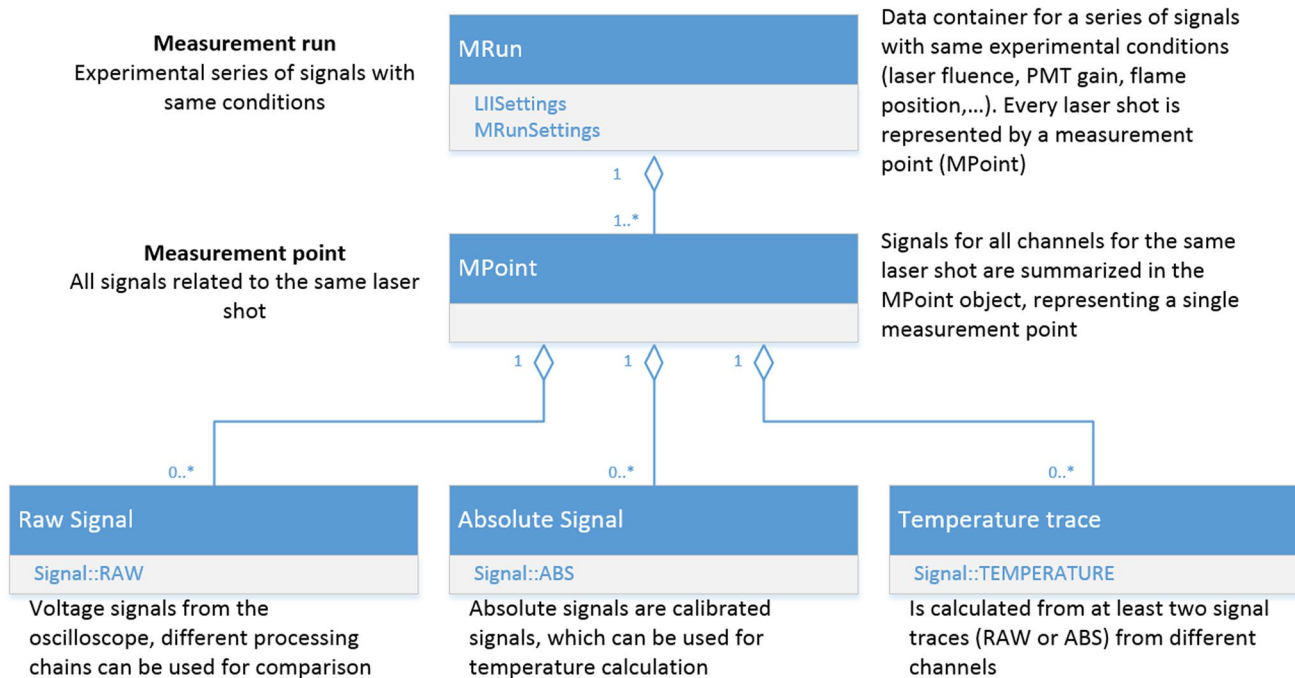


Fig. 1 Data structure used within the LIISim software

Table 1 List of available property types and the corresponding temperature- or wavelength-dependent formulae

LIISim type	Function	Equation (temperature T , parameter $a_0 \dots a_8$)
const	Constant value	$f(T) = a_0$
case	Piecewise-defined	$f(T) = \begin{cases} a_1, & T \leq a_0 \\ a_2, & T > a_0 \end{cases}$
poly	Polynomial	$f(T) = a_0 + a_1T + a_2T^2 + a_3T^3 + a_4T^4 + a_5T^5 + a_6T^6 + a_7T^7 + a_8T^8$
poly2	Polynomial	$f(T) = a_0 + a_1T + a_2T^2 + a_3T^3 + a_4T^{-1} + a_5T^{-2}$
polycase	Piecewise-defined polynomial	$f(T) = \begin{cases} a_1 + a_2T + a_3T^2 + a_4T^3, & T \leq a_0 \\ a_5 + a_6T + a_7T^2 + a_8T^3, & T > a_0 \end{cases}$
exp	Exponential with Euler number as base	$f(T) = a_0 + a_1 \exp(a_2 + a_3T^{-1} + a_4T)$
exppoly	Exponential with polynomial exponent	$f(T) = a_0 + a_1 \exp(a_2 + a_3T + a_4T^2 + a_5T^3 + a_6T^4 + a_7T^5)$
powx	Exponential with parameter as base	$f(T) = a_0 + a_1 a_2^{(a_3 + a_4T^{-1} + a_5T)}$
optics_temp	Wavelength-dependent polynomial provided for each wavelength	$f(\lambda, T) = f(a_0, T) = a_1 + a_2T + a_3T^2 + a_4T^3$
optics_case	Wavelength-dependent piecewise-defined polynomial	$f(\lambda, T) = f(a_0, T) = \begin{cases} a_2, & T \leq a_1 \\ a_3, & T > a_1 \end{cases}$
optics_lambda	Wavelength-dependent polynomial	$f(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3 + a_4\lambda^4 + a_5\lambda^5 + a_6\lambda^6 + a_7\lambda^7 + a_8\lambda^8$
optics_exp	Wavelength-dependent exponential	$f(\lambda) = a_0 \lambda^{(1-a_1)}$

to create customized property files can be found in the user guide of the respective LIISim version.

3.4 Numerical procedures

For data evaluation, the residual between simulated and experimental data is minimized by varying the model parameters using the Levenberg–Marquardt nonlinear least-squares method [43]. The implemented algorithm is loosely based on Numerical Recipes 3rd Edition [44] and is described in detail in the Appendix. All numerical procedures for matrix transformation, integration, solving ordinary differential equations (ODEs) and fitting are summarized in the Numeric Class of the LIISim source code.

3.5 Modules

The software is structured in four main modules: Database, Signal Processing, Analysis Tools and Fit Tools, which can be accessed through a tab menu. Database visualizes the previously described materials databases and also provides some tools for modifications. Signal Processing is the main module, which allows the processing of raw signals, absolute signals and temperature traces. Analysis Tools provides only visualization functionalities for deeper insight into the processed data and for comparison of multiple measurement runs. Finally, the Fit Tools use the calculated temperature traces for fitting the data to pre-implemented heat-transfer models. The graphical user interface for the Signal Processing module is exemplarily shown in Fig. 2. A detailed

description of all functionalities can be found in the user guide.

3.5.1 Signal processing

Within the “Signal Processing” module, processing steps can be individually ordered and parameters can be set individually for every measurement run, for a group or globally. This allows the systematic processing of all loaded measurement data. Each signal type (i.e., raw, absolute, temperature) has its own processing chain, which is consecutively executed in the order: raw, absolute, temperature. The following processing plugins are available and can be applied multiple times in a signal processing chain:

- “Baseline” provides two options for signal-offset correction. (i) For gated PMTs: the signal before the gate opening can be used to calculate the offset. This is done by averaging a user-defined signal range and subtract this from the signal. (ii) For ungated PMTs, the baseline offset for each channel can be defined in the LIISettings file. During processing, these values are automatically retrieved and subtracted from the signal trace.
- “Arithmetic” operations (addition, subtraction, multiplication, and division) can be applied on each data point for all or a single channel. Applicable, for example for signal inversion, scaling, offset correction.
- “Normalization” scales the signal according to the signal peak or a user-defined value. The operation can be applied on all or a single channel.

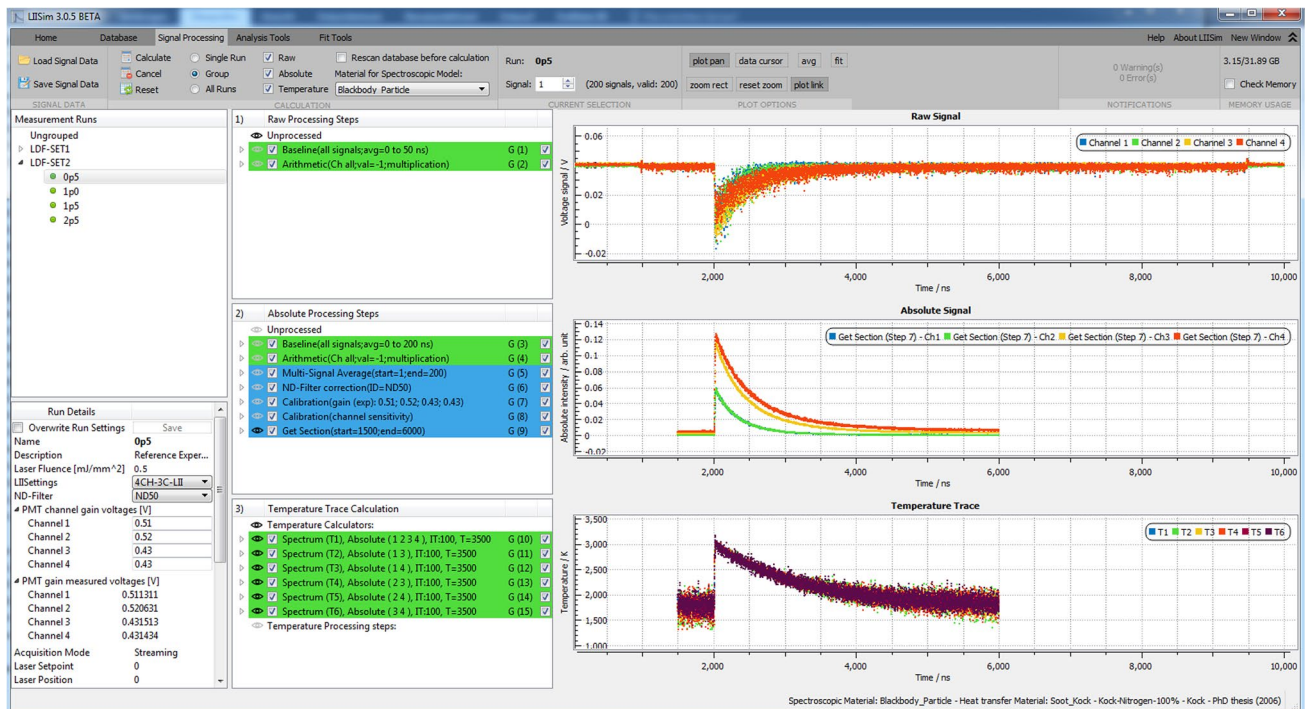


Fig. 2 Graphical user interface (GUI) for “Signal Processing”

- “Neutral-density (ND) filter correction” uses the transmission values from the current LIISettings according to the selected filter identifier. The identifier can be automatically selected from the measurement run settings file or manually chosen. Each signal trace is divided by the respective filter transmission values.
- “Calibration (gain, channel sensitivity)” takes the channel calibration values for detector gain and sensitivity from the current LIISettings. The gain correction factor y_i for channel i is calculated for “gain (exp)” according to:

$$y_i = \exp(-A_i \ln(x_i/x_{i, \text{ref}})), \quad (7)$$

and for “gain (log10)” as:

$$y_i = 10^{-A_i \log_{10}\left(\frac{x_i}{x_{i, \text{ref}}}\right)}, \quad (8)$$

where x_i is the channel gain voltage, $x_{i, \text{ref}}$ the gain reference voltage and A_i the gain calibration value for the respective channel. The gain correction factor and channel-sensitivity calibration values are multiplied with the signal traces giving calibrated signal traces.

- “Multi-signal average” calculates an averaged signal from multiple single signals within a measurement run. Subsequent processing steps are then applied only to the averaged signal.

The absolute signal, which is corrected for spectral sensitivity of the detectors, can then be used in the temperature processing chain to calculate temperature traces with the “Two-Color” and “Spectrum” method as described in Sect. 2.1. When the “Spectrum” method is used, the fitted spectrum for each iteration and time step can be visualized using the Analysis Tool “Temperature Fit” (see next section).

3.5.2 Analysis tools

The analysis tools allow the systematic analysis and comparison of multiple measurement runs. Three tools are available: “Plotter”, “Temperature Fit”, and “Parameter Analysis”. The simplest tool “Plotter” visualizes selected measurement runs and allows comparison of the signal traces.

“Temperature Fit” (Fig. 3) requires previously calculated temperature traces using the “Spectrum” method in the “TemperatureCalculator”-processing step. This tool allows the comparison of different temperature calculation settings (i.e., spectral fit with various numbers of channels, different initial conditions, or optical properties) and helps making assumptions of the quality of the fitting result or identify any errors within the processing iterations. Figure 3 illustrates the GUI with the temperature trace plot (top left) for data point selection, the fitting result at the selected position (top right) and visualization of the fitting iterations with plots (bottom left) and tables (bottom right).

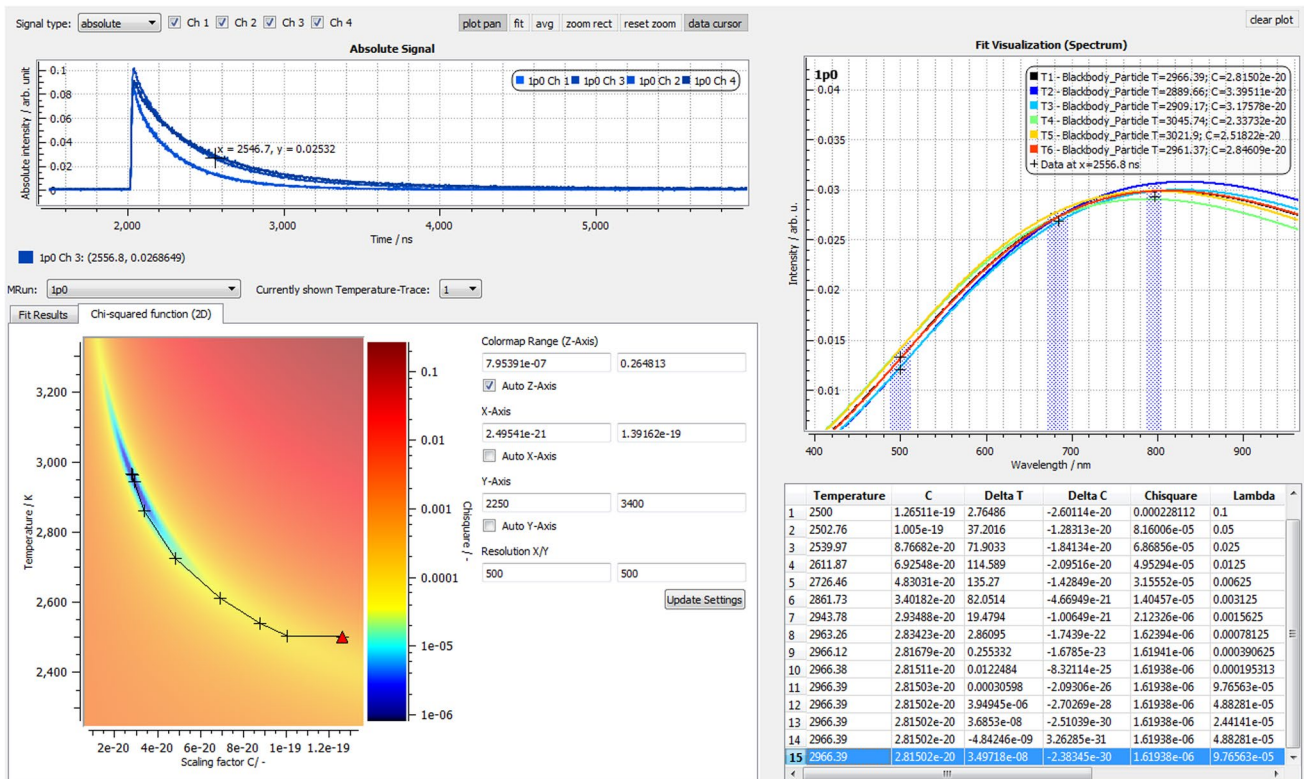


Fig. 3 GUI for the “Temperature Fit” analysis tool showing the temperature trace and for the selected position, the iterations and the final result of the spectral temperature fit

For each measurement run, experimental parameters (i.e., laser fluence, ND-filter, PMT gain voltages, etc.) can be stored in the measurement run settings that are accessible in every module. These parameters can be used to compare selected measurement runs in the “Parameter Analysis” tool (Fig. 4). Available built-in parameters are for example laser fluence, PMT gain, ND-filter, average of signal range, but also user-defined parameters can be added. The GUI shows the signal traces of the selected measurement runs (top left), the experimental parameters curves to be visualized (bottom left) and the result in graphical (top right) and tabular format (bottom right).

3.5.3 Fit tools

After signal processing and analysis of the temperature traces, pre-implemented heat-transfer models and materials databases can be used to perform a fit to the data. For this, temperature traces or sections of temperature traces can be selected as data range for the fit. Three fit parameters (i.e., initial particle diameter, gas temperature, and peak temperature) can be set as free or fixed parameters. For solving the ordinary differential equation (ODE) of the selected heat-transfer model, the type of ODE solver and the accuracy can be selected. The result of the fit and each iteration can be analyzed using various plots (i.e., heat transfer rates, particle

diameter history, fit error, etc.). The GUI of the “Fit Creator” (Fig. 5) shows the data to be fitted (top left), settings for the fitting (top right), and an overview of the meta-data for the performed fitting runs (bottom right). The experimental temperature trace is overlaid with the modeled temperature trace (bottom left). Additional plots provide information about the temporally resolved heat transfer rates, particle size, and fitting parameter changes during the fitting iterations.

4 Application cases

Two example cases are used to demonstrate the main functionalities of LIISim. First, the recently published validation dataset by Goulay et al. [17] and second, recently collected own unpublished data using the same apparatus as in [18] on a standardized laminar diffusion flame [45]. Both data sets are available online as electronic supplementary material and can be imported within LIISim as benchmark data (for details refer to the user guide on <http://www.liisim.com>). A MATLAB script for data conversion of the Goulay et al. dataset into the LIISim format can be found in the electronic supplementary material. For both application cases, the materials database combinations as summarized in Table 2 were used.

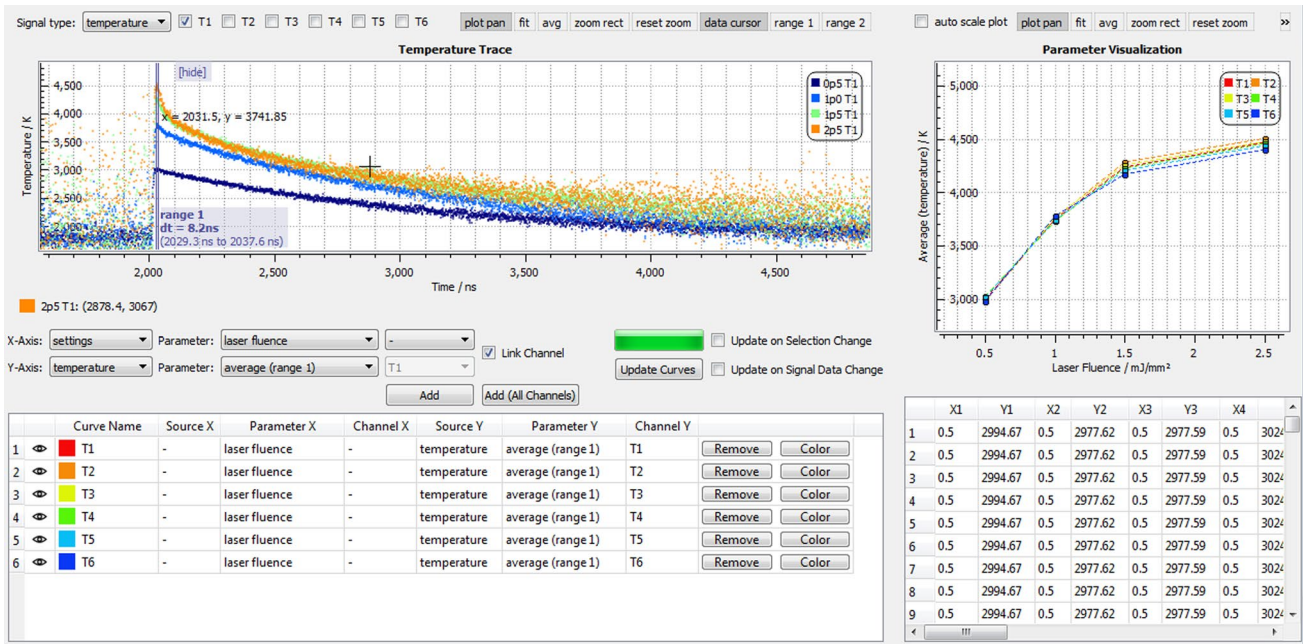


Fig. 4 GUI for the “Parameter Analysis” tool showing multiple measurement runs with varying laser fluence

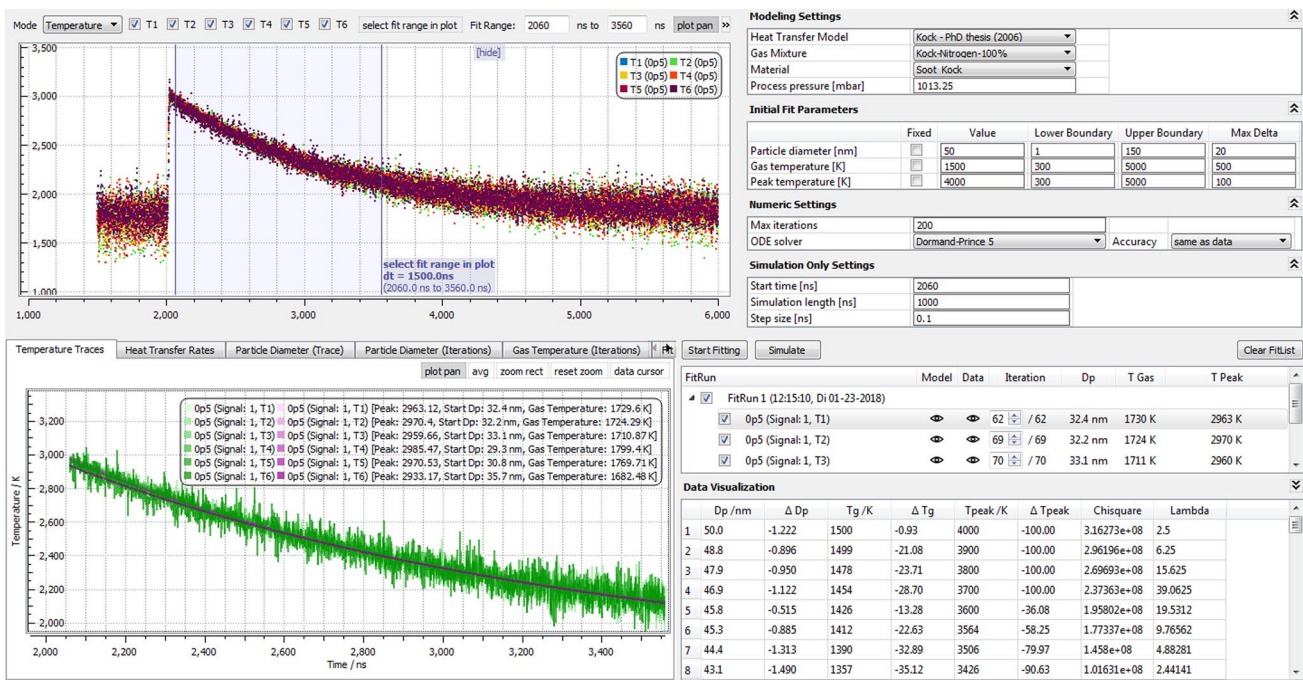


Fig. 5 GUI for the “Fit Creator” tool showing fits for multiple temperature traces

4.1 Application 1

We selected the three-channel LII dataset for the Santoro burner [46] provided by Goulay et al. [17] to visualize the spectral fitting and heat-transfer modeling. The signals are

collected at center wavelengths of 452, 682, and 855 nm. The wavelength information is stored in the LIISettings file and is automatically used during analysis. Basic signal processing was not necessary, because the data have already been corrected for background, PMT sensitivity,

Table 2 Materials database combinations used for the simulations

	Model 1	Model 2	Model 3
Heat-transfer model	Kock	Liu	Melton
Gas mixture	Kock-Nitrogen-100%	Liu-Flame	Melton-Nitrogen-100%
Material	Soot_Kock	Soot_Liu	Soot_Melton(Workshop)

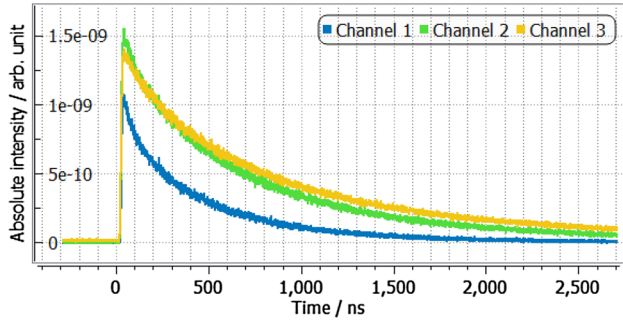


Fig. 6 Original LII signal traces for three spectral detection channels at a laser fluence of 0.986 mJ/mm² (1064 nm)

and neutral-density filter characteristics. To overcome fitting convergence difficulties the data were scaled by a factor of 10⁹. The imported signal traces for a laser fluence of 0.986 mJ/mm² and an excitation wavelength of 1064 nm are shown in Fig. 6.

For the temperature calculation, spectral fits according to Eq. (4) have been performed using the absorption function $E(m_\lambda)$ suggested by the original publication:

$$E(m_\lambda) = \frac{\lambda^{1-\xi} \beta}{6\pi}, \tag{9}$$

which simplifies with the provided values of $\xi = 0.83$ and $\beta = 28.72 \text{ cm}^{-0.17}$ to:

$$E(m_\lambda) = a_0 \lambda^{a_1}, \tag{10}$$

with $a_0 = 3.333 \text{ m}^{-0.17}$ and $a_1 = 0.17$ and λ in meter. These two parameters are then used within the “optics_exp” equation type (see Table 1) through the spectroscopic materials database file.

In a multi-wavelength measurement, temperatures can be derived from various combinations of two (or three) spectral detection channels. If the spectroscopic model is correct and no measurement errors are present, all combinations should lead to the same result. For this dataset, four different temperature traces (T1–T4) have been investigated. The LII signal channels used for the analysis are shown in Table 3.

After processing, the temperature traces are compared in the same plot. Figure 7 shows the calculated temperature traces using the “Spectrum” mode, which reveals some disagreement within the first microsecond after the laser pulse. For further investigation, the analysis tools can be used on

Table 3 LII-signal channels used for the calculation of the temperature traces

	Channel 1 (452 nm)	Channel 2 (682 nm)	Channel 3 (855 nm)
T1	x	x	x
T2	x	x	
T3	x		x
T4		x	x

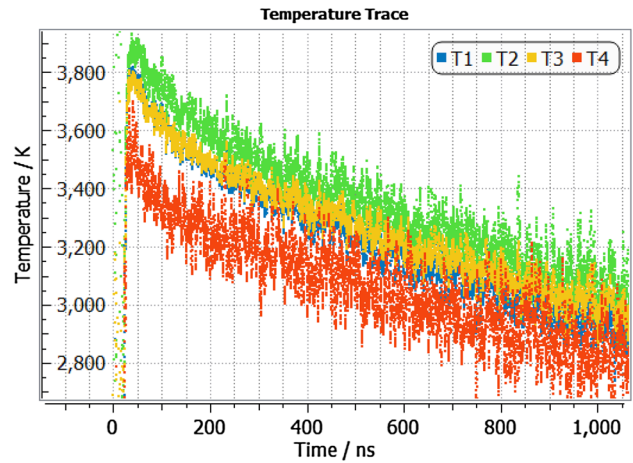


Fig. 7 Temperature traces calculated using the “Spectrum” method

the temperature fit to look into the single iteration steps during temperature fitting.

In the “Temperature Fit” tool, single data points can be picked from the signal to visualize the fitted temperature spectrum for all four temperature traces (Fig. 8). It can be seen that for example at 60 ns the solution of the spectral fits approximately matches the data from the selected signal channels, but the temperatures vary from 3527 to 3797 K for different temperature traces (T1–T4). Possible reasons for this discrepancy could be non-linearity affecting one or more of the three channels as demonstrated in [18] or the change of optical properties during particle cooling. To trace down the cause of this effect, further measurements using the same setup would be required.

For presentation of the particle-sizing feature, we use temperature trace T1 in the FitCreator module, for comparison of selected heat-transfer models (Kock, Liu and

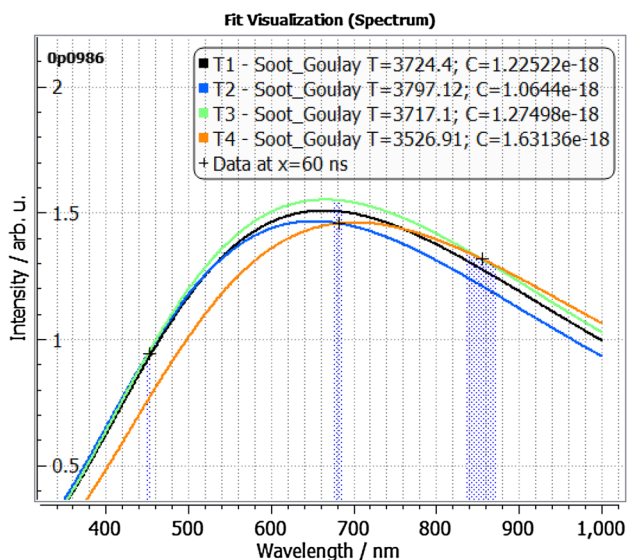


Fig. 8 Fit for different channel combinations T1–T4 at 60 ns

Melton [22]). For the fitting, the process pressure was set to 1.013 bar (1 atm) and the gas temperature to 1676 K according to the original publication. The range for the fit was set from 60 to 1560 ns and the ODE solver “Dormand-Prince 5” was used (accuracy same as data) [47]. The initial particle diameter and the peak temperature are free parameters and the gas temperature was kept constant. For comparison, the same procedure was applied to the 0.495 mJ/mm² data. The simulated temperature traces for each heat-transfer model and the fitting results are displayed in Fig. 9.

The three modeled traces for the low fluence data (0.495 mJ/mm²) fit the measured data, but gave initial particle sizes much higher than previously reported values of 33 nm [1]. For the 0.986 mJ/mm² data, the Melton model predicted higher values for the particle diameter as the

LIISim upper boundary value of 150 nm. This upper boundary should ensure that the range of validity of the physical equations is not exceeded. The fitted values for the particle diameter using the other two pre-implemented models differ also widely from previously reported values. To our knowledge, the only publication using this validation data set for model development was Lemaire and Mobtil [48], who stated they could not find any agreement between their modeled data and the validation data set. With further publications of LII data on this particular flame, data from different groups could be compared to investigate the origin of these discrepancies. This helps to distinguish whether this effect is dependent on the specific experiment or the implemented models.

4.2 Application 2

Data have been collected using a four-channel LII device with bandpass center-wavelengths at 500, 684, and 797 nm. A detailed description of the detection device can be found in [18]. A laminar non-premixed flame (Gülder burner), which has been investigated by TiRe-LII previously [1, 15, 49], was operated at its standard conditions of 0.194 slm (standard liters per minute) ethene and 284 slm air and the soot particles are heated by pulses (8 ns FWHM) from an Nd:YAG laser at 1064 nm with a fluence of 0.5 and 1 mJ/mm². Raw LII signals are collected for 200 laser shots and then processed using the following processing chain to generate calibrated averaged signal traces:

1. Baseline correction (using the signal detected before the PMT gate opening averaged for 200 ns)
2. Arithmetic multiplication with -1 for inversion of the signal trace
3. Multi-signal average
4. ND-filter correction

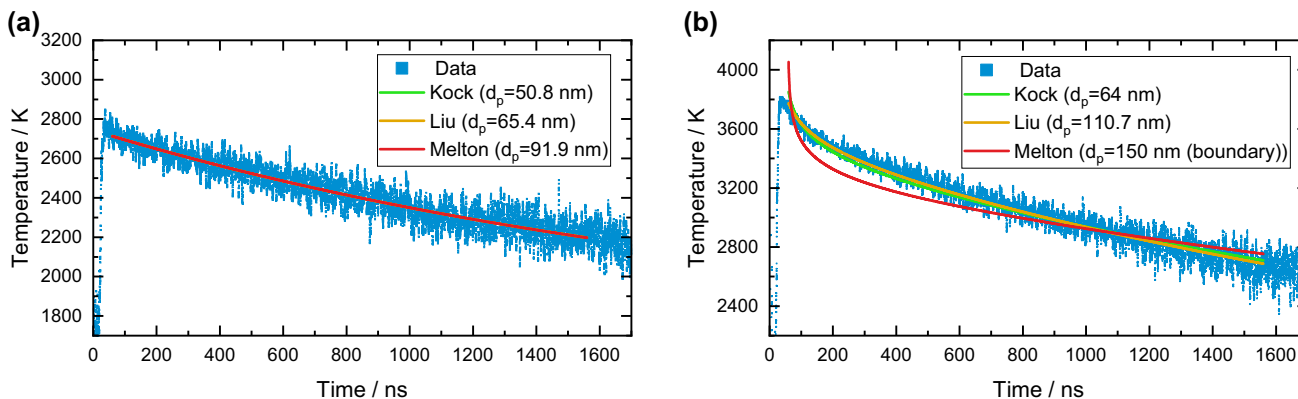


Fig. 9 Comparison of fitting results from three heat-transfer models (range 60 to 1560 ns) for 0.495 mJ/mm² (a) and 0.986 mJ/mm² (b)

5. Calibration (PMT gain)
6. Calibration (channel sensitivity)
7. Get section (1500–6000 ns).

After signal processing, the calibrated and averaged signals (Fig. 10 center row) can be used within the temperature-processing chain to calculate temperature traces (Fig. 10 bottom row).

With four available channels at three different wavelengths one temperature trace (T1) was calculated using a spectral fit of all four channels, and for comparison with two-color pyrometry, five temperature traces (T2–T6) were calculated using all possible pairs at independent wavelengths according to Table 4. For this analysis, a constant $E(m_\lambda)$ value was used for all channels (using the spectroscopic materials data “Blackbody_Particle”).

The same settings as in Application 1 were used except for the gas temperature, which was reported for this laminar diffusion flame as 1730 K [50]. For evaluation, a 1.5- μ s range was selected starting 30 ns after the signal peak to avoid interference with the signal variation during laser heating. The comparison of the fitting results is summarized in Fig. 12.

All fits for the low fluence of 0.5 mJ/mm² show a good agreement between modeled and calculated temperatures traces and initial particle sizes from the Kock model agree with previously reported values of 29 to 32 nm using LII and TEM [1]. The Liu and Melton model yielded higher particle sizes of 52.9 and 48.4 nm, respectively. For the 1 mJ/

mm² fits, the modeled temperature trace shows a much step-per gradient in the beginning of the signal, which indicates over-prediction of the evaporation cooling. Comparison with measurements of the same flame from other groups could help to trace down, whether this is a LII model problem or related to the specific experimental setup.

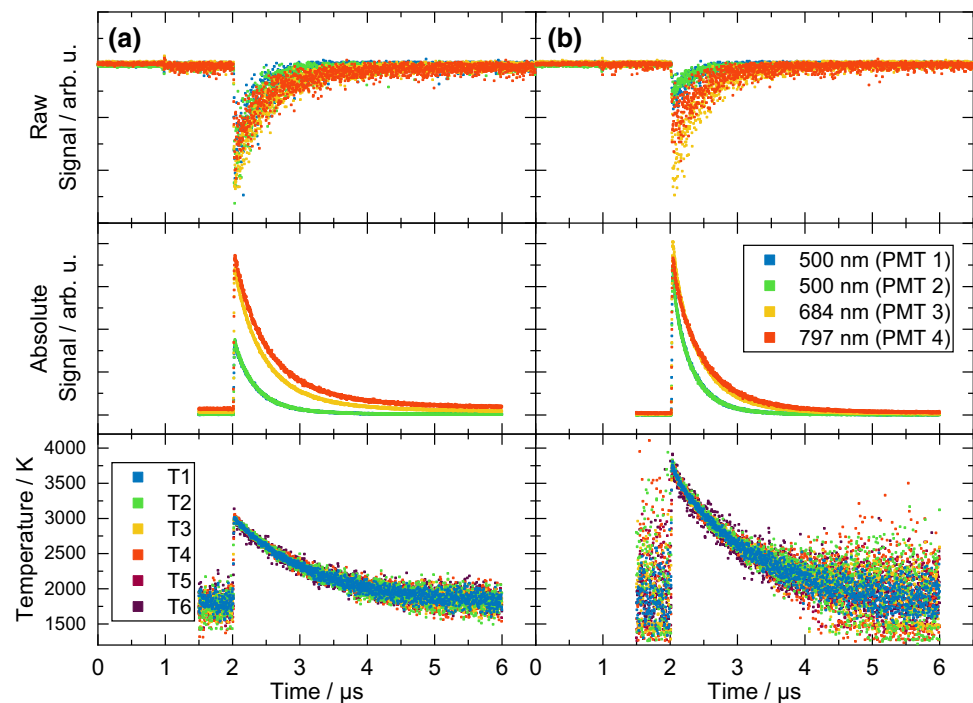
5 Software availability and documentation

As the new LIISim version is a complete rebuild, we want to avoid confusion with earlier LIISim versions and have chosen the version number 3.0 for the first release. For tracking changes of the source code, we are using the public version-control service GitHub (<http://www.github.com/LIISim>). Documentation, software releases and further announcements are available through <http://www.liisim.com>.

6 Summary

The new LIISim is the first open and public software toolbox for signal processing and data analysis of laser-induced incandescence measurement data. It was designed to be user-friendly, freely available for the scientific community, with transparent algorithms and versatile tools for data processing and evaluation. The software targets newcomers in the LII technique, but also experienced researchers who want to compare their signal processing procedures to

Fig. 10 Raw LII signals (single shot) collected from the oscilloscope (top row), calibrated and averaged signals (center row), calculated temperature traces from calibrated signals (bottom row) for 0.5 mJ/mm² (a) and 1 mJ/mm² (b). (bottom row) shows the result of the temperature processing for all six temperature traces, which show good agreement. The spectral fit at 2060 ns was visualized using the temperature fit tool (see Fig. 11) giving temperatures between 2901 and 2945 K for 0.5 mJ/mm² and between 3637 and 3680 K for 1 mJ/mm²



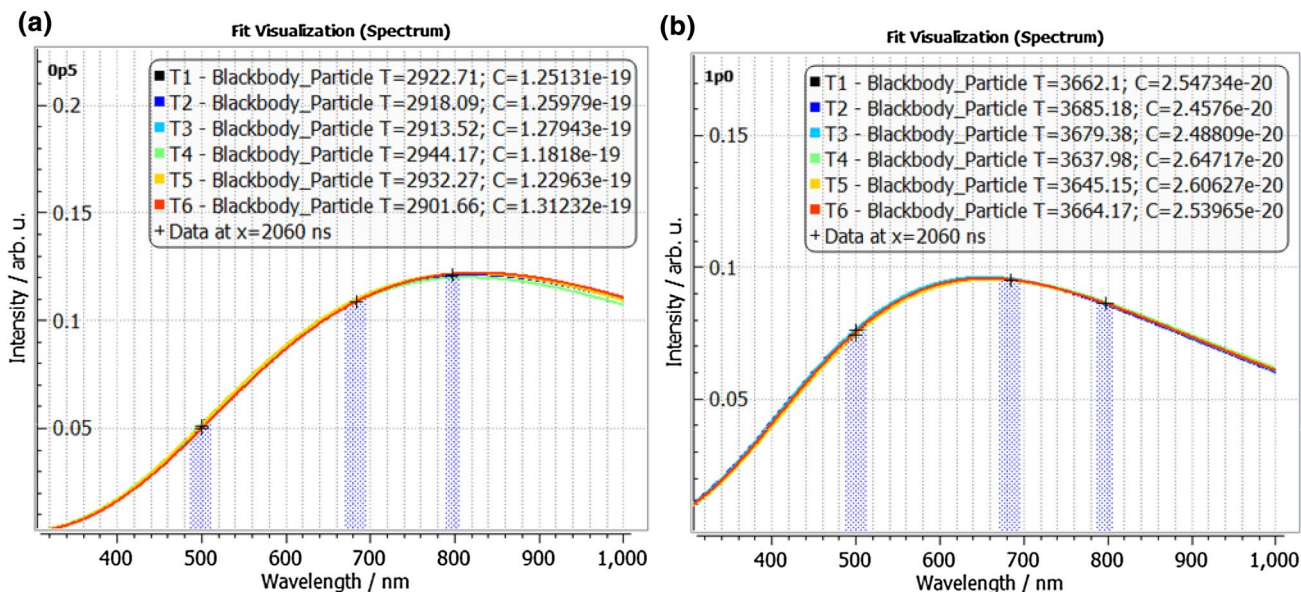


Fig. 11 Fit for channel combinations T1–T6 at 2060 ns for 0.5 mJ/mm² (a) and 1 mJ/mm² (b)

Table 4 LII signal channels used for the calculation of the temperature traces

	Channel 1 (500 nm)	Channel 2 (500 nm)	Channel 3 (684 nm)	Channel 4 (797 nm)
T1	x	x	x	x
T2	x		x	
T3	x			x
T4		x	x	
T5		x		x
T6			x	x

an open standard. The toolbox provides an easy entrance in this measurement technique with simple and transparent algorithms, built-in example databases and pre-implemented

published heat-transfer models. Modular materials databases in a standardized file format allow switching physical properties without changing the processing algorithms and allow applications of both, soot and synthetic nanoparticle sources. A common file format can act as a foundation for public reference measurement databases for standardized flames and synthetic nanoparticle source and should enable future inter-laboratory comparison studies in the context of the biennial LII workshop. The source code of the software is published under the GNU General Public License (GPL) to allow further development by all interested researchers within the scientific community.

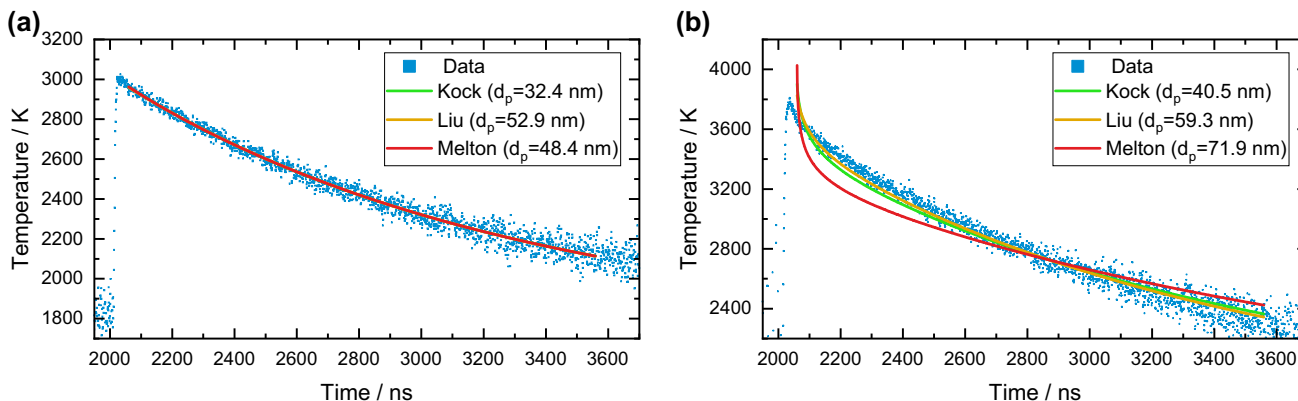


Fig. 12 Comparison of fitting results from three heat-transfer models (2060–3560 ns) for 0.5 mJ/mm² (a) and 1 mJ/mm² (b)

6.1 Summary of parameters

Property	LIISim variable	Unit	Description	
α_T	alpha_T_eff	–	Thermal accommodation coefficient (conduction)	
ϵ	eps	–	Total emissivity for radiation heat transfer	
γ	gamma	–	Heat capacity ratio calculated from heat capacity	
$\gamma(T)$	gamma_eqn	–	Heat capacity ratio provided as equation	
κ_a	therm_cond	W/m/K	Thermal conductivity of gas molecules	
ρ_p	rho_p	kg/m ³	Particle density	
θ_e	theta_e	–	Thermal accommodation coefficient (evaporation)	
$C_{p,mol}$	C_p_mol	J/mol/K	Molar heat capacity	
c_p	c_p_kg	J/kg/K	Specific heat capacity	
$E(m)$	Em	–	$E(m)$ absorption value at wavelength λ	
$E(m)$	Em_func	–	$E(m)$ absorption function	
ΔH_v	H_v	J/mol	Enthalpy of evaporation	
L	L	m	Mean free path of gas molecules	
M	molar_mass	kg/mol	Molar mass	
M_v	molar_mass_v	kg/mol	Molar mass of vapor species	
p_v	p_v	Pa	Vapor pressure function	
p_v^*	p_v_ref	Pa	Reference pressure for Clausius–Clapeyron	
T_v^*	T_v_ref	K	Reference temperature for Clausius–Clapeyron	
Constant	LIISim variable	Unit	Value (LIISim/NIST)	Description
c_0	c0	m/s	2.99792458×10^8	Speed of light in vacuum
h	h	Js	$6.62607004 \times 10^{-34}$	Planck constant
k_B	k_B	J/K	$1.38064852 \times 10^{-23}$	Boltzmann constant
N_A	N_A	1/mol	$6.02214085 \times 10^{23}$	Avogadro constant
R	R	J/mol/K	8.3144598	Molar gas constant
π	pi	–	3.14159265359	Pi
σ	sigma	W/m ² K ⁴	5.670367×10^{-8}	Stefan–Boltzmann constant
c_1	c_1	Wm ²	$c_1 = 2hc_0^2$	First radiation constant for spectral radiance
c_2	c_2	Km	$c_2 = hc_0/k_B$	Second radiation constant

Acknowledgements We gratefully thank Stanislav Musikhin (University of Duisburg-Essen, Germany) for testing the software and giving helpful feedback. We acknowledge funding through the German Research Foundation via SCHU1369/14 and SCHU1369/20.

Appendix A

Levenberg–Marquardt algorithm

For a given data vector y of the length m and for n parameters \mathbf{a} and a vector of standard deviations σ_i , the residuals for a given model $y_{mod}(x_i, \mathbf{a})$ (spectroscopic or heat-transfer model) can be described as

$$f_i(\mathbf{a}) = \frac{y_i - y_{mod}(x_i, \mathbf{a})}{\sigma_i} \quad i = 1, 2, \dots, m, \tag{11}$$

and

$$\mathbf{F}(\mathbf{a}) = \begin{bmatrix} f_1(\mathbf{a}) \\ \vdots \\ f_m(\mathbf{a}) \end{bmatrix} \in \mathbb{R}^m. \tag{12}$$

The goal is now to minimize the nonlinear least-squares problem for the parameters \mathbf{a}

$$\arg \min_{\mathbf{a}} f(\mathbf{a}) \tag{13}$$

with

$$f(\mathbf{a}) = \sum_{i=1}^m f_i(\mathbf{a})^2. \tag{14}$$

The gradient of $f(\mathbf{a})$ can be written in matrix notation as

$$\nabla f(\mathbf{a}) = 2\mathbf{J}(\mathbf{a})^T \mathbf{F}(\mathbf{a}) \in \mathbb{R}^m, \tag{15}$$

where $\mathbf{J}(\mathbf{a})$ is the Jacobian

$$\mathbf{J}(\mathbf{a}) = \begin{bmatrix} \frac{\partial f_1}{\partial a_1} & \dots & \frac{\partial f_1}{\partial a_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial a_1} & \dots & \frac{\partial f_m}{\partial a_n} \end{bmatrix} \in \mathbb{R}^{m \times n}, \tag{16}$$

and \mathbf{D} half of the Hessian matrix:

$$\nabla^2 f(\mathbf{a}) \approx 2\mathbf{J}(\mathbf{a})^T \mathbf{J}(\mathbf{a}) = 2\mathbf{D}. \tag{17}$$

For each iteration, the gradient of the parameters \mathbf{a} can be found by solving [51]:

$$(\mathbf{J}(\mathbf{a})^T \mathbf{J}(\mathbf{a}) + \lambda \mathbf{I}) \Delta \mathbf{a} = -\mathbf{J}(\mathbf{a}^k)^T \mathbf{F}(\mathbf{a}^k), \tag{18}$$

which can be transformed using the Cholesky decomposition to the form $\mathbf{LL}^T \mathbf{x} = \mathbf{b}$ with

$$\mathbf{LL}^T = (\mathbf{D} + \lambda \mathbf{I})$$

$$\mathbf{x} = \Delta \mathbf{a} \quad (19)$$

$$\mathbf{b} = -\mathbf{J}(\mathbf{a}^k)^T \mathbf{F}(\mathbf{a}^k).$$

Now \mathbf{x} can be found by forward $\mathbf{L}\mathbf{y} = \mathbf{b}$ and backward substitution $\mathbf{L}^T \mathbf{x} = \mathbf{y}$, which gives the new parameter approximation for the next iteration k :

$$\mathbf{a}^{k+1} = \mathbf{a}^k + \Delta \mathbf{a}. \quad (20)$$

In LIISim, the parameters \mathbf{a}^k are visualized for the temperature fit in “AnalysisTools Temperature Fit” and for the heat-transfer modeling in the FitCreator module.

Appendix B

Implemented heat-transfer models from literature [22]. The heat transfer rates are defined in the HeatTransferModel child classes in the “calculations/models/” folder of the source code.

The following materials and gas mixture properties are calculated for all models according:

Name	Variable	Symbol (original)	Symbol (LIISim)	Equation	Unit
Specific heat capacity of the particle	c_p_kg	c_s	c_p	$c_p = C_{p,\text{mol}}/M_p$	$\text{J kg}^{-1} \text{K}^{-1}$
Thermal velocity of gas molecules	c_tg	$c_{\text{tg}}(T_g)$	$c_{\text{tg}}(T_g)$	$c_{\text{tg}} = \left(\frac{8 k_B N_A T_g}{\pi M_{\text{mix}}} \right)^{\frac{1}{2}}$	m s^{-1}
Molar heat capacity of gas mixture	C_p_mol	–	$C_{p,\text{mix}}$	$C_{p,\text{mix}} = \sum_i x_i C_{p,g,i}$	$\text{J mol}^{-1} \text{K}^{-1}$
Heat capacity ratio	gamma	$\gamma(T_g)$	$\gamma(T_g)$	$\gamma(T_g) = \frac{C_{p,\text{mix}}}{C_{p,\text{mix}} - R}$	–
Molar mass of gas mixture	molar_mass	–	M_{mix}	$M_{\text{mix}} = \sum_i x_i M_{g,i}$	kg mol^{-1}

Kock model

Materials properties (Soot_Kock)

Name	Variable	Type	Symbol (original)	Symbol (LIISim)	a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	Unit	Comment
Accommodation coefficient	alpha_T_eff	const	α_T	α_T	0.23	–	–	–	–	–	–	–	–	–	–
Accommodation coefficient	theta_e	const	α_M	θ_e	1.0	–	–	–	–	–	–	–	–	–	–
Molar heat capacity	C_p_mol	poly2 ^a	–	$C_{p,\text{mol}}$	22.5566	0.0013	–	–	–	-1.8195×10^6	–	–	–	$\text{J mol}^{-1} \text{K}^{-1}$	Calculated from given c_s
Total emissivity	eps	const	ϵ	ϵ	1.0	–	–	–	–	–	–	–	–	–	–
Molar mass	molar_mass	const	W_s	M_p	0.012011	–	–	–	–	–	–	–	–	kg mol^{-1}	–
Molar mass of vapor	molar_mass_v	const	W_v	M_v	0.036033	–	–	–	–	–	–	–	–	kg mol^{-1}	–
Density	rho_p	const	ρ_s	ρ_p	1860	–	–	–	–	–	–	–	–	kg m^{-3}	–
Enthalpy of evaporation	H_v	const	ΔH_v	ΔH_v	790776.6	–	–	–	–	–	–	–	–	J mol^{-1}	–

Name	Variable	Type	Symbol (original)	Symbol (LIISim)	a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	Unit	Comment
Reference pressure	p_v_ref	const	p_{ref}	p_v^*	61.5	-	-	-	-	-	-	-	-	Pa	Clausius–Clapeyron
Reference temperature	T_v_ref	const	T_{ref}	T_v^*	3000	-	-	-	-	-	-	-	-	K	Clausius–Clapeyron

$${}^a f(T) = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^{-1} + a_5 T^{-2}$$

Gas mixture properties (Kock-Nitrogen-100%)

Composition:

Gas	Variable	Symbol	Fraction
Nitrogen_Kock	x	x	1.0

Gas properties (Nitrogen_Kock)

Name	Variable	Type	Symbol (original)	Symbol (LIISim)	a_0	a_1	a_2	a_3	a_4	a_5	Unit
Molar mass	molar_mass	const	M_g	M_g	0.028014	-	-	-	-	-	kg mol ⁻¹
Molar heat capacity ^a	C_p_mol	poly2 ^b	$C_{mp,g}$	$C_{p,g}$	28.58	0.00377	-	-	-	-50,000	J mol ⁻¹ K ⁻¹

^aFor nitrogen from [52]

$${}^b f(T) = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^{-1} + a_5 T^{-2}$$

Heat-transfer model (HTM_KockSoot)

Evaporation:

$$\dot{Q}_{evap} = -\frac{\Delta H_v}{M_v} \dot{u}_{evap}, \tag{21}$$

$$\dot{u}_{evap} = -\theta_e \frac{1}{4} \pi d_p^2 c_{tv} \rho_v, \tag{22}$$

$$c_{tv} = \left(\frac{8 k_B N_A T_p}{\pi M_v} \right)^{\frac{1}{2}}, \tag{23}$$

$$\rho_v = \frac{p_v M_v}{R T_p}, \tag{24}$$

$$p_v = p_v^* \exp\left(-\frac{\Delta H_v}{R} \left(\frac{1}{T_p} - \frac{1}{T_v^*}\right)\right). \tag{25}$$

Conduction:

$$\dot{Q}_{cond, fm} = \frac{\alpha_T \pi d_p^2 p_g c_{tg}}{8} \left(\frac{\gamma + 1}{\gamma - 1}\right) \left(\frac{T_p}{T_g} - 1\right), \tag{26}$$

$$c_{tg} = \left(\frac{8 k_B N_A T_g}{\pi M_{mix}}\right)^{\frac{1}{2}}. \tag{27}$$

Radiation:

$$\dot{Q}_{rad} = \pi d_p^2 \varepsilon \sigma (T_p^4 - T_g^4). \tag{28}$$

Liu model

Materials properties (Soot_Liu)

Name	Variable	Type	Sym- bol (orig- inal)	Sym- bol (Liu- Sim)	a_0	a_1	a_2	a_3	a_4	a_5	a_6	Unit	Comment
Accommodation coefficient	alpha_T_eff	const	α_T	α_T	0.37	-	-	-	-	-	-	-	-
Accommodation coefficient	theta_e	const	α_M	θ_e	0.77	-	-	-	-	-	-	-	-
Molar heat capacity	C_p_mol	poly ^a	$C_{p,mol}$		3.54288	3.55694×10^{-2}	-2.55018×10^{-5}	9.83713×10^{-9}	-2.10385×10^{-12}	2.35752×10^{-16}	-1.07879×10^{-20}	$J mol^{-1} K^{-1}$	Valid from 1200 to 5500 K; calculated from given c_s
Total emissivity	eps	const	ϵ	ϵ	0.4	-	-	-	-	-	-	-	-
Molar mass	molar_mass	const	W_v	M_p	0.012011	-	-	-	-	-	-	$kg mol^{-1}$	-
Molar mass of vapor	molar_mass_v	const	W_1	M_v	17.179×10^{-3}	6.8654×10^{-7}	2.9962×10^{-9}	-8.5954×10^{-13}	1.0486×10^{-16}	-	-	$kg mol^{-1}$	-
Density	rho_p	const	ρ_s	ρ_p	1860	-	-	-	-	-	-	$kg m^{-3}$	-
Enthalpy of evaporation	H_v	poly ^a	Δh_v	ΔH_v	2.05398×10^5	7.366×10^2	-0.40713	1.1992×10^{-4}	-1.7946×10^{-8}	1.0717×10^{-12}	-	$J mol^{-1}$	-
Vapor pressure	p_v	exppo-ly ^b	p_v	p_v	-	$101,325$ (unit conversion)	-122.96	9.0558×10^{-2}	-2.7637×10^{-5}	4.1754×10^{-9}	-2.4875×10^{-13}	Pa	Original unit: [atm] from fits to data

^a $f(T) = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4 + a_5 T^5 + a_6 T^6 + a_7 T^7 + a_8 T^8$

^b $f(T) = a_0 + a_1 \exp(a_2 + a_3 T + a_4 T^2 + a_5 T^3 + a_6 T^4 + a_7 T^5)$

Gas mixture properties (Liu_Flame)

Composition:

Gas	Variable	Symbol	Fraction
FlameAir_Liu	x	x	1.0

Properties (manually set for composition):

Name	Variable	Type	Symbol (original)	Sym- bol (LIISim)	a_0	a_1	a_2	a_3	a_4	Unit
Heat capacity ratio ^a	gamma_eqn	poly ^b	γ	γ	1.4221163416	$-1.8636002383 \times 10^{-4}$	$8.0783894569 \times 10^{-8}$	$-1.6425082302 \times 10^{-11}$	$1.2750021975 \times 10^{-15}$	-

^aFor flame mixture from [53]

^b $f(T) = a_0 + a_1T + a_2T^2 + a_3T^3 + a_4T^4 + a_5T^5 + a_6T^6 + a_7T^7 + a_8T^8$

Gas properties (FlameAir_Liu)

Name	Variable	Type	Symbol (original)	Symbol (LIISim)	a_0	Unit
Molar mass	molar_mass	const	-	M_g	0.02874	kg mol ⁻¹

Heat-transfer model (HTM_Liu)

Evaporation:

$$\dot{Q}_{\text{evap}} = -\frac{\Delta H_v}{M_v} \dot{m}_{\text{evap}}, \tag{29}$$

$$\dot{m}_{\text{evap}} = -\frac{\pi d_p^2 M_v \theta_c p_v}{RT_p} \left(\frac{RT_p}{2\pi M_v} \right)^K, \tag{30}$$

with $K = 0.5$.

Conduction:

$$\dot{Q}_{\text{cond}} = \frac{\pi d_p^2 \alpha_T p_0}{2T_g} \sqrt{\frac{RT_g}{2\pi M_{\text{mix}}}} \left(\frac{\gamma^* + 1}{\gamma^* - 1} \right) (T_p - T_g), \tag{31}$$

$$\frac{1}{\gamma^* - 1} = \frac{1}{T - T_0} \int_{T_0}^T \frac{1}{\gamma(T') - 1} dT'. \tag{32}$$

This heat-transfer model uses polynomial fitting coefficients for calculation of $\gamma(T)$. These are provided through the “gamma_eqn” property of the LIISim implementation in the GasMixture database. If $\gamma(T)$ is not defined, the heat capacity of the gas mixture $C_{p,\text{mix}}(T)$ is used to calculate $\gamma(T)$ according to:

$$\gamma(T) = \frac{C_{p,\text{mix}}(T)}{C_{p,\text{mix}}(T) - R}. \tag{33}$$

Radiation:

$$\dot{Q}_{\text{rad}} = \frac{199\pi^3 d_p^3 (k_B T)^5 \epsilon}{h(hc)^3}. \tag{34}$$

Melton model

Materials properties (Soot_Melton(workshop))

Name	Variable	Type	Symbol (original)	Symbol (LIISim)	a_0	Unit	Comment
Accommodation coefficient	alpha_T_eff	const	α_T	α_T	0.3	–	
Accommodation coefficient	theta_e	const	α_M	θ_e	1.0	–	
Molar heat capacity	C_p_mol	const	–	$C_{p,mol}$	22.8	$\text{J mol}^{-1} \text{K}^{-1}$	^a calculated from given c_s
Molar mass	molar_mass	const	W_s	M_p	0.012	kg mol^{-1}	
Molar mass of vapor	molar_mass_v	const	W_v	M_v	0.036	kg mol^{-1}	
Density	rho_p	const	ρ_s	ρ_p	2260	kg m^{-3}	
Enthalpy of evaporation	H_v	const	ΔH_v	ΔH_v	7.78×10^5	J mol^{-1}	
Reference pressure	p_v_ref	const	p_{ref}	p_v^*	100,000	Pa	Clausius–Clapeyron
Reference temperature	T_v_ref	const	T_{ref}	T_v^*	3915	K	Clausius–Clapeyron

$${}^a C_{p,mol} = c_{s,Melton} M_p$$

Gas mixture properties (Melton-Nitrogen-100%)

Composition:

Gas	Variable	Symbol	Fraction
Nitrogen_Melton	x	x	1.0

Properties (manually set for composition)

Name	Variable	Type	Symbol (original)	Symbol (LIISim)	a_0	a_1	Unit	Comment
Thermal conductivity	therm_cond	const	κ_a	κ_a	0.1068	–	W/m/K	Original unit W/cm/K
Mean free path	L	poly ^a	L	L	–	2.355×10^{-10}	m	Original unit: cm

$${}^a f(T) = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4 + a_5 T^5 + a_6 T^6 + a_7 T^7 + a_8 T^8$$

Gas properties (Nitrogen_Melton)

Name	Variable	Type	Symbol (original)	Symbol (LIISim)	a_0	Unit	Comment
Molar heat capacity	C_p_mol	const	–	$C_{p,g}$	36.0295	$\text{J mol}^{-1} \text{K}^{-1}$	^a calculated from given $\gamma(1800 \text{ K}) = 1.3$

$${}^a \gamma(1800 \text{ K}) = 1.3 = \frac{C_p}{C_p - R} \Rightarrow C_p = \frac{1.3}{0.3} R$$

Heat-transfer model (HTM_Melton)

Evaporation:

$$\dot{Q}_{\text{evap}} = - \frac{\Delta H_v}{M_p} \dot{u}_{\text{evap}}. \quad (35)$$

This model uses molar mass of solid species in Eq. (35).

$$\dot{u}_{\text{evap}} = - \frac{\pi d_p^2 M_v \theta_c p_v}{RT_p} \left(\frac{RT_p}{2M_v} \right)^{0.5}, \quad (36)$$

$$p_v = p_v^* \exp \left(- \frac{\Delta H_v}{R} \left(\frac{1}{T_p} - \frac{1}{T_v^*} \right) \right). \quad (37)$$

Conduction:

$$\dot{Q}_{\text{cond}} = \frac{2\kappa_a \pi d_p^2}{d_p + GL(T_g)} (T_p - T_g), \quad (38)$$

$$G = \frac{8f}{\alpha_T(\gamma + 1)}, \quad (39)$$

$$f = \frac{9\gamma - 5}{4}. \quad (40)$$

Radiation:

Not included in this model.

References

- C. Schulz, B.F. Kock, M. Hofmann, H. Michelsen, S. Will, B. Bougie, R. Suntz, G. Smallwood, Appl. Phys. B **83**(3), 333–354 (2006)
- H.A. Michelsen, C. Schulz, G.J. Smallwood, S. Will, Prog. Energy Combust. Sci. **51**, 2–48 (2015)
- T. Dreier, C. Schulz, Powder Technol. **287**, 226–238 (2016)
- A.V. Filippov, M.W. Markus, P. Roth, J. Aerosol Sci. **30**(1), 71–87 (1999)
- R.L. Vander Wal, T.M. Ticich, J.R. West, Appl. Opt. **38**(27), 5867–5879 (1999)
- Y. Murakami, T. Sugatani, Y. Nosaka, The Journal of Physical Chemistry A **109**(40), 8994–9000 (2005)
- T. Lehre, R. Suntz, H. Bockhorn, Proc. Combust. Inst. **30**(2), 2585–2593, (2005)
- A. Eremin, E. Gurentsov, C. Schulz, J. Phys. D Appl. Phys. **41**(5), 055203 (2008)
- S. Maffi, F. Cignoli, C. Bellomunno, S. De Iuliis, G. Zizak, Spectrochim. Acta Part B Atomic Spectrosc. **63**(2), 202–209 (2008)
- T.A. Sipkens, R. Mansmann, K.J. Daun, N. Petermann, J.T. Titanah, M. Karttunen, H. Wiggers, T. Dreier, C. Schulz, Appl. Phys. B **116**(3), 623–636 (2014)
- K. Daun, J. Menser, R. Mansmann, S.T. Moghaddam, T. Dreier, C. Schulz, Journal of Quantitative Spectrosc. Radiat. Transfer **197**, 3–11 (2017)
- J. Menser, K. Daun, T. Dreier, C. Schulz, Appl. Phys. B **122**(11), 277 (2016)
- T.A. Sipkens, N.R. Singh, K.J. Daun, Appl. Phys. B **123**(1), 14 (2017)
- F. Cignoli, S. De Iuliis, V. Manta, G. Zizak, Appl. Opt. **40**(30), 5370–5378 (2001)
- D.R. Snelling, G.J. Smallwood, F. Liu, Ö.L. Gülder, W.D. Bachalo, Appl. Opt. **44**(31), 6773–6785 (2005)
- T. Lehre, B. Jungfleisch, R. Suntz, H. Bockhorn, Appl. Opt. **42**(12), 2021–2030 (2003)
- F. Goulay, P.E. Schrader, X. López-Yglesias, H.A. Michelsen, Appl. Phys. B **112**(3), 287–306 (2013)
- R. Mansmann, T. Dreier, C. Schulz, Appl Opt **56**(28), 7849–7860 (2017)
- F. Goulay, P.E. Schrader, H.A. Michelsen, Appl. Phys. B **100**(3), 655–663 (2010)
- L.A. Melton, Appl. Opt. **23**(13), 2201–2208 (1984)
- B. Kock, B. Tribalet, C. Schulz, P. Roth, Combust. Flame **147**(1–2), 79–92 (2006)
- H.A. Michelsen, F. Liu, B.F. Kock, H. Bladh, A. Boiarciuc, M. Charwath, T. Dreier, R. Hedef, M. Hofmann, J. Reimann, S. Will, P.E. Bengtsson, H. Bockhorn, F. Foucher, K.P. Geigle, C. Mouñaïm-Rousselle, C. Schulz, R. Stirn, B. Tribalet, R. Suntz, Appl. Phys. B **87**(3), 503–521 (2007)
- C. Schulz, Laser-induced incandescence: quantitative interpretation, modelling, application, in *Proceedings international bunsen discussion meeting and workshop*, Duisburg, Germany, 25–28 September 2005, CEUR-WS.org. <http://ceur-ws.org/Vol-195>. Accessed 20 Jan 2018
- R. Suntz, H. Bockhorn, Laser-induced incandescence: quantitative interpretation, modelling, applications in *Proceedings 2nd international discussion meeting and workshop*, Bad Herrenalb, Germany, 2–4 August 2006, CEUR-WS.org. <http://ceur-ws.org/Vol-211>. Accessed 20 Jan 2018
- P. Desgroux, Laser-induced incandescence 2012, in *Proceedings 5th international workshop on laser-induced incandescence*, Le Touquet, France, 8–11 May 2012, CEUR-WS.org. <http://ceur-ws.org/Vol-865>. Accessed 20 Jan 2018
- A. Nanthaamornphong, K. Morris, D.W.I. Rouson, H.A. Michelsen, A case study: agile development in the community laser-induced incandescence modeling environment (CLiME), in *2013 5th international workshop on software engineering for computational science and engineering (SE-CSE)* (2013), pp. 9–18
- A. Nanthaamornphong, J.C. Carver, K. Morris, H.A. Michelsen, D.W.I. Rouson, Comput. Sci. Eng. **16**(3), 36–46 (2014)
- M. Hofmann, B. Kock, C. Schulz, A web-based interface for modeling laser-induced incandescence (LIISim), in *Proceedings of the European Combustion Meeting*, (Kreta, 2007)
- A.M. Hofmann, PhD dissertation, Heidelberg University, 2006
- M. Hofmann, B.F. Kock, T. Dreier, H. Jander, C. Schulz, Appl. Phys. B **90**(3–4), 629–639 (2007)
- M. Leschowski, PhD dissertation, University of Duisburg-Essen, 2016
- E. Cenker, G. Bruneaux, T. Dreier, C. Schulz, Appl. Phys. B **119**(4), 745–763 (2015)
- E. Cenker, G. Bruneaux, T. Dreier, C. Schulz, Appl. Phys. B **118**(2), 169–183 (2014)
- E. Cenker, K. Kondo, G. Bruneaux, T. Dreier, T. Aizawa, C. Schulz, Appl. Phys. B **119**(4), 765–776 (2015)
- S. Maffi, S. De Iuliis, F. Cignoli, G. Zizak, Appl. Phys. B **104**(2), 357–366 (2011)
- S. De Iuliis, M. Urciuolo, A. Cammarota, S. Maffi, R. Chirone, G. Zizak, *XXXIV Meeting of the Italian Section of the Combustion Institute* (Rome, 2011)

37. S. De Iuliis, F. Cignoli, G. Zizak, *Appl. Opt.* **44**(34), 7414–7423 (2005)
38. T.A. Sipkens, P.J. Hadwin, S.J. Grauer, K.J. Daun, *Appl Opt* **56**(30), 8436–8445 (2017)
39. C.F. Bohren, D.R. Huffman, *Absorption and scattering of light by small particles*. (John Wiley & Sons, 2008)
40. The Qt Framework, July 2015. <http://www.qt.io>
41. C. Boost, Libraries, November 2013. <http://www.boost.org/>
42. Qwt library, June 2016. <http://qwt.sourceforge.net/>
43. J.J. Moré, *Numerical analysis* (Springer, Berlin, Heidelberg, 1978), pp. 105–116
44. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical recipes 3rd edition: the art of scientific computing*. (Cambridge University Press, Cambridge, 2007)
45. ÖL. Gülder, D.R. Snelling, R.A. Sawchuk, *Symp. (Int.) Combust.* **26** (2), 2351–2358, (1996)
46. B. Quay, T.W. Lee, T. Ni, R.J. Santoro, *Combust. Flame* **97**(3), 384–392 (1994)
47. J.R. Dormand, P.J. Prince, *J. Comput. Appl. Math.* **6**(1), 19–26 (1980)
48. R. Lemaire, M. Mobtil, *Appl. Phys. B* **119**(4), 577–606 (2015)
49. R. Mansmann, K. Thomson, G. Smallwood, T. Dreier, C. Schulz, *Opt. Express* **25**(3), 2413 (2017)
50. D.R. Snelling, K.A. Thomson, F. Liu, G.J. Smallwood, *Appl. Phys. B* **96**(4), 657–669 (2009)
51. R.C. Aster, B. Borchers, C.H. Thurber, *Parameter estimation and inverse problems*. (Elsevier Acad. Press, Amsterdam, 2005)
52. B.F. Kock, *Zeitaufgelöste Laserinduzierte Inkandeszenz (TR-LII): Partikelgrößenmessung in einem Dieselmotor und einem Gasphasenreaktor*. (Cuvillier, Göttingen, 2006)
53. F. Liu, K.J. Daun, D.R. Snelling, G.J. Smallwood, *Appl. Phys. B* **83**(3), 355–382 (2006)