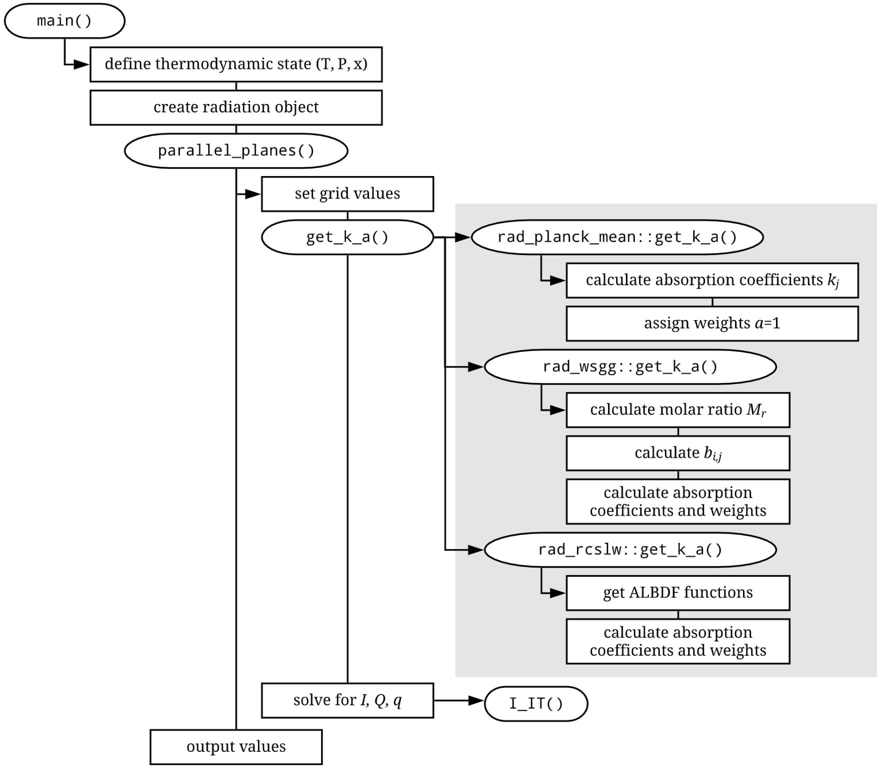
Motivation

- Radiative heat transfer is a key process in many combustion configurations, e.g., fire
- Accurate representation is challenging
  - Complex solution of the RTE
  - Complex representation of spectral gas and soot properties
  - Computationally expensive
  - Combustion simulation considers complex chemistry, turbulent flow, multicomponent mass transfer, soot formation
    - Burden on users/developers for submodel expertise
  - Availability of libraries that offload submodels facilitates code development and progress
    - Cantera, …

https://energy.sandia.gov/programs/nuclear-energy/nuclear-energy-safety-security/

RADLIB

- RADLIB: radiative heat transfer library
- Provides radiative properties for use in RTE solvers
- Current models:
  - RCSLW
  - WSGG
  - Planck Mean
- Species: *CO_2, H_2O, CO, soot*
- Includes a 1D RTE solver
- Multiple examples
- Open source
- C++, Python, Fortran interfaces
- Published, documented

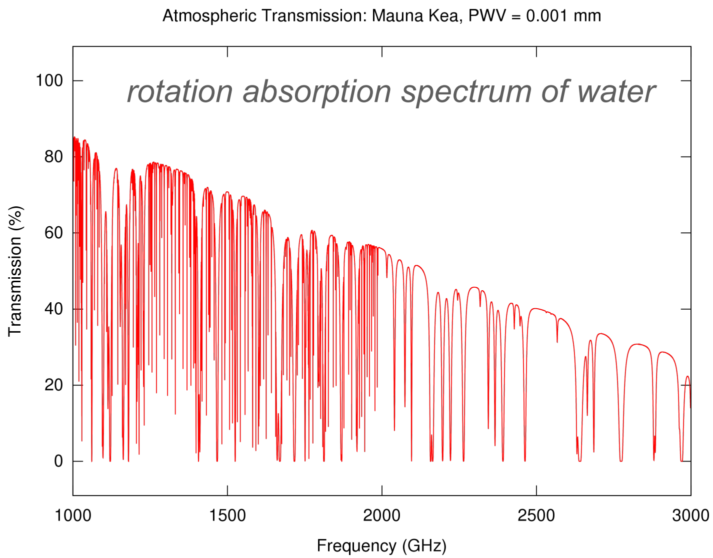


Models

- Spectral RTE (neglect scattering)

$$\frac{dI_\eta}{ds} = -\kappa_\eta I_\eta + \kappa_\eta I_{b,\eta}$$

- Absorption spectra include millions of spectral lines.
    - T, P, composition dependent
- Line-by-Line (LBL) solvers expensive, limited to simple configurations



https://commons.wikimedia.org/wiki/File:Atmospheric_terahertz_transmittance_at_Mauna_Kea_(simulated).svg

Global Models (WSGG)

- Spectrally integrated radiation properties
- Assume gray over wavenumber region j

$$\frac{dI_\eta}{ds} = -\kappa_\eta I_\eta + \kappa_\eta I_{b,\eta}$$

$$\int_{\eta\in\{\Delta_j\}}$$

$$\frac{dI_j}{ds} = -\kappa_j I_j + a_j \kappa_j I_b, \quad j = 0, 1, \ldots, n$$

$$a_j = \frac{\int_{\eta_j} I_{b,\eta} d\eta}{I_b} \qquad \sum_j a_j = 1$$

- Given spatial profiles of κ_j and a_j, solve for I_j profiles

Global Models (WSGG)

- Spectrally integrated radiation properties
- Assume gray over wavenumber region j

$$\frac{dI_\eta}{ds} = -\kappa_\eta I_\eta + \kappa_\eta I_{b,\eta}$$

$$\int_{\eta\in\{\Delta_j\}}$$

$$\frac{dI_j}{ds} = -\kappa_j I_j + a_j \kappa_j I_b, \quad j = 0, 1, \ldots, n$$

$$a_j = \frac{\int_{\eta_j} I_{b,\eta} d\eta}{I_b} \qquad \sum_j a_j = 1$$

- Given spatial profiles of κ_j and a_j, solve for I_j profiles

- Total intensity

$$I = \int I_\eta d\eta = \sum_j I_j$$

- Heat flux (kW/m^2)

$$\mathbf{q} = \int_{4\pi} I(\mathbf{s})\mathbf{s} d\Omega$$

- Volumetric heat source (kW/m^3)

$$Q = -\nabla \cdot \mathbf{q}$$

RCSLW

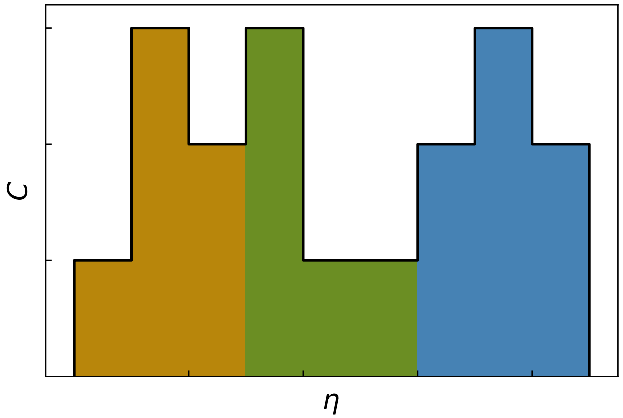
- Integrating over wavenumber bands Δ_j is limited because κ_η varies widely within the band
- Spectral Line WSGG (SLW) models integrate over a set of (non-sequential) η for which the absorption cross section C_η is uniform (to within the C_η-width of the band)
  - $\kappa_\eta = C_\eta N$ where N is molar density
- κ_j just corresponds to the given band
- a_j is the fraction of the black emission in Δ_j

$$a_j = F(\tilde{C}_j; \phi, T) - F(\tilde{C}_{j-1}; \phi, T)$$

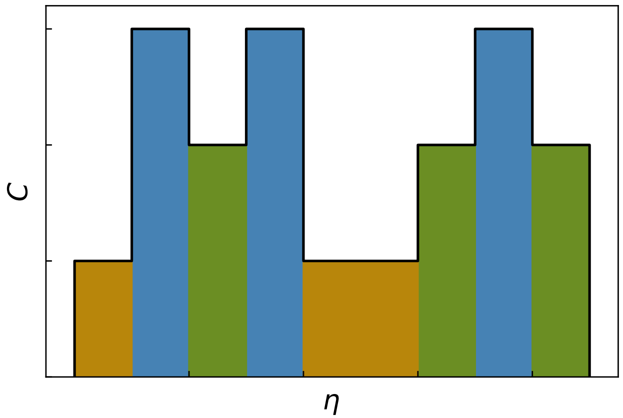
$$F(C; \phi, T) = \frac{1}{\sigma T^4} \int_{\{\eta : C_\eta(\eta,\phi) < C\}} E_b(\eta, T) d\eta$$

- F varies between 0 and 1 and is monotonic in C, so it can be inverted.

WSGG



SLW



Solovjov et al. J. Quant. Spectrosc. Radiat. Tranf. 197:26-44 (2017)

RCSLW

- The integrated RTE assumes Δ_j are independent of spatial position **x**, but C_η depends on the thermodynamic state ϕ, which depends on **x**.
- This is handled differently in variations of SLW models
- Rank correlation approach
  - Two positions with states ϕ_1 and ϕ_2, with cross section spectra $C_\eta(\eta,\phi_1)$, $C_\eta(\eta,\phi_2)$
  - Let $\hat{C}_1 \equiv C_\eta(\hat{\eta}, \phi_1)$ and $\hat{C}_2 \equiv C_\eta(\hat{\eta}, \phi_2)$ denote the cross sections at some wavenumber $\hat{\eta}$
  - Define

$$H_1 = \eta : C_\eta(\eta, \phi_1) < \hat{C}_1$$

$$H_2 = \eta : C_\eta(\eta, \phi_2) < \hat{C}_2$$

  - If $H_1=H_2$ for arbitrary $\hat{\eta}$ the spectra $C_\eta(\eta,\phi_1)$, $C_\eta(\eta,\phi_2)$ are *rank correlated.*

$$F(\hat{C}_1, \phi_1, T_r) = \frac{1}{\sigma T_r^4} \int_{H_1} E_b(\eta, T_r) d\eta$$

$$F(\hat{C}_2, \phi_2, T_r) = \frac{1}{\sigma T_r^4} \int_{H_2} E_b(\eta, T_r) d\eta$$

$$\longrightarrow \quad F(\hat{C}_1, \phi_1, T_r) = F(\hat{C}_2, \phi_2, T_r)$$

  - For a given state ϕ_1 and cross section $\hat{C}_1$, we can evaluate F, then invert to find $\hat{C}_2$ at another ϕ_2

RCSLW

- **Algorithm**
  - Specify a reference temperature T_r on the domain of interest (e.g. an average)
  - Specify a collection of fixed F_j points with boundaries $\tilde{F}_j$ (where 0<F<1)
  - At each position with state ϕ invert F_j and $\tilde{F}_j$ to compute C_j and $\tilde{C}_j$
  - Compute the $\kappa_j = C_j N$ for each j at each position
  - Compute the $a_j = F(\tilde{C}_j; \phi, T) - F(\tilde{C}_{j-1}; \phi, T)$ for each j at each position

RCSLW

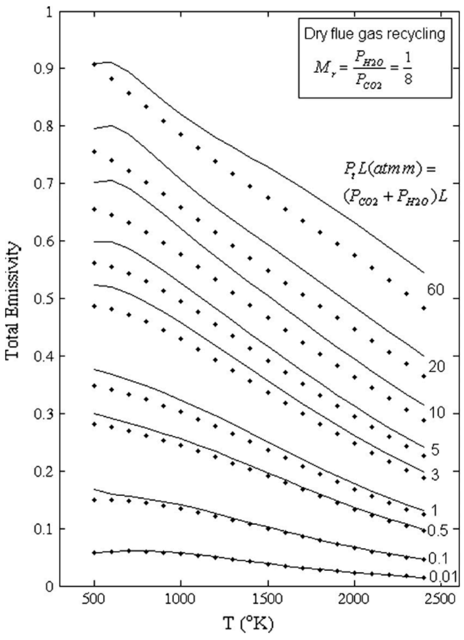
- **Algorithm**
  - Specify a reference temperature T_r on the domain of interest (e.g. an average)
  - Specify a collection of fixed F_j points with boundaries $\tilde{F}_j$ (where 0<F<1)
  - At each position with state ϕ invert F_j and $\tilde{F}_j$ to compute C_j and $\tilde{C}_j$
  - Compute the $\kappa_j = C_j N$ for each j at each position
  - Compute the $a_j = F(\tilde{C}_j;\phi,T) - F(\tilde{C}_{j-1};\phi,T)$ for each j at each position
- ALBDF data is available:
  - H2O, CO2, and CO
  - P=0.1-50 atm,
  - T=300-3000 K.
  - *Pearson et al. J. Quant. Spectrosc. Radiat. Transf. 143:83-91 (2018).*
- For mixtures: $C_\eta = x_{CO_2}C_{CO_2,\eta} + x_{H_2O}C_{H_2O,\eta} + x_{CO}C_{CO,\eta}$

$$F_\eta(C_\eta) = F_{CO_2,\eta}\left(\frac{C\eta}{x_{CO_2}}\right)F_{H_2O,\eta}\left(\frac{C\eta}{x_{H_2O}}\right)F_{CO,\eta}\left(\frac{C\eta}{x_{CO}}\right)$$

"multiplication method"

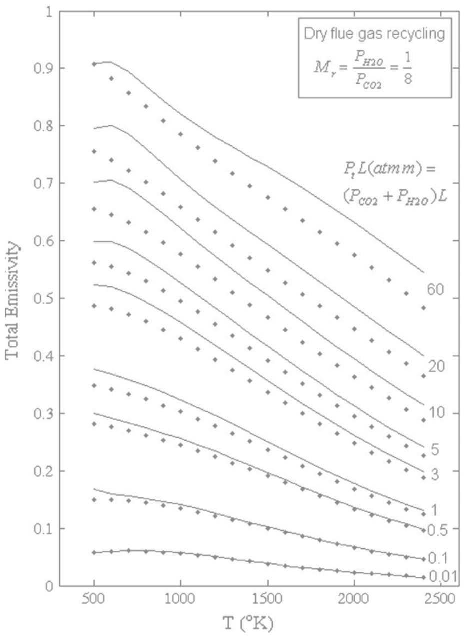
WSGG

- **WSGG model**
  - Bordbar et al., Combust. Flame, 161:2435-2445 (2014)
  - Bordbar et al. Int. Commun. Heat Mass Transf., 110:104400 (2020)
  - 4 gray + 1 clear gas
  - H_2O, CO_2
  - κ_j, a_j correlated to $M_r = x_{H2O}/x_{CO2}$,
    - 0.01 < Mr < 4 (interpolate to pure CO2, H2O outside)
  - T = 300-2400 K
  - Pressure-path length: 0.1-60 atm-m

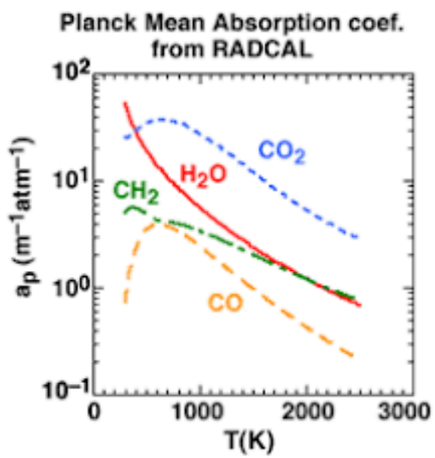


PM

- **WSGG model**
  - Bordbar et al., Combust. Flame, 161:2435-2445 (2014)
  - Bordbar et al. Int. Commun. Heat Mass Transf., 110:104400 (2020)
  - 4 gray + 1 clear gas
  - H_2O, CO_2
  - κ_j, a_j correlated to $M_r = x_{H2O}/x_{CO2}$,
    - 0.01 < Mr < 4 (interpolate to pure CO2, H2O outside)
  - T = 300-2400 K
  - Pressure-path length: 0.1-60 atm-m



- **Planck Mean model**
  - TNF Workshop, https://tnfworkshop.org/radiation/
  - H_2O, CO_2, CO, CH_4
  - T = 300-2300 K



Soot

- Included in all three models
- Unagglomerated spheres
- Rayleigh small particle limit
- Neglects scattering
- Plank Mean:
  - κ_s is added to the κ_{gas}
- WSGG:
  - κ_s is added to each κ_j
- RCSLW
  - Spectral treatment
  - $F_{s,\eta}$ computed and included in the mixture treatment (multiplication method)

$$\kappa_{s,\eta} = C_0 f_v \eta$$

$$\kappa_s = 3.72 \frac{f_v C_0 T}{C_2} = 1817 f_v T \ \ m^{-1}$$

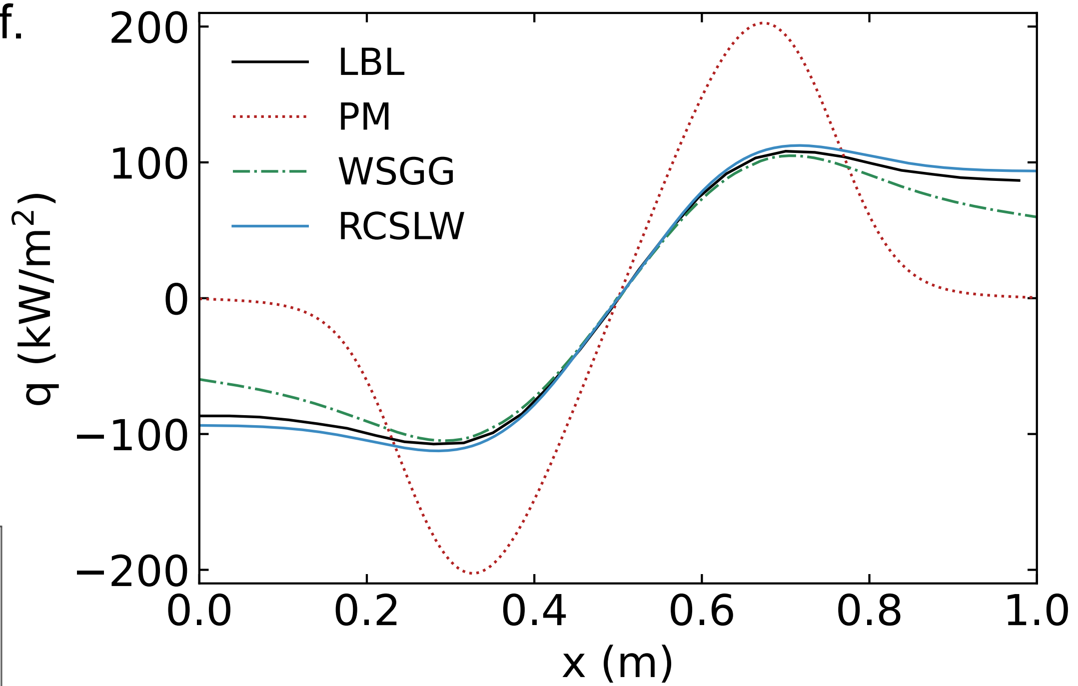
$$C_0 = \frac{36\pi n k}{(n^2 - k^2 + 2)^2 + 4n^2k^2}$$

$$C_2 = 0.014388 \ \ m{\cdot}K$$

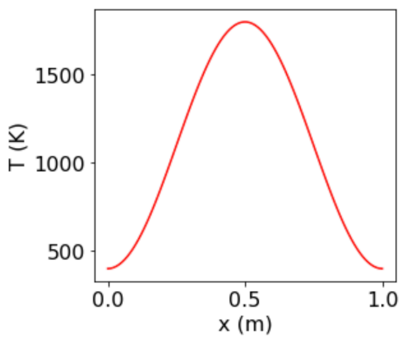
- n, k are real, imaginary parts of the complex index of refraction
- n=1.75, k=1.03 → C_0=7.03
  - *Williams et al. Int. J. Heat Mass Transf. 50:1616-1630 (2007)*

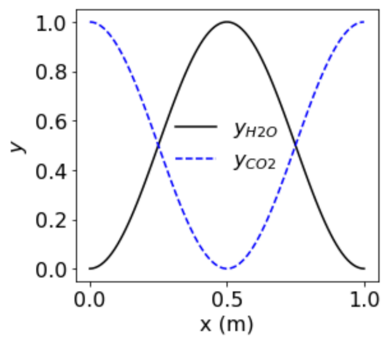
Examples

- 1D planar
- P = 1 atm, vary T, composition prof.
- 7 configurations
- Compare results with LBL simulations.
- Ray tracing: 1000 pts, 101 angles



Example B3



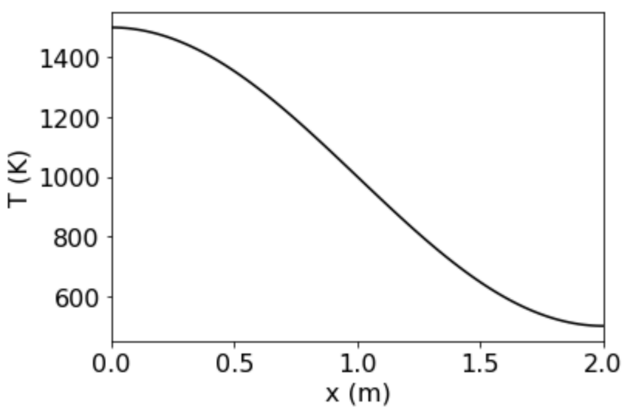


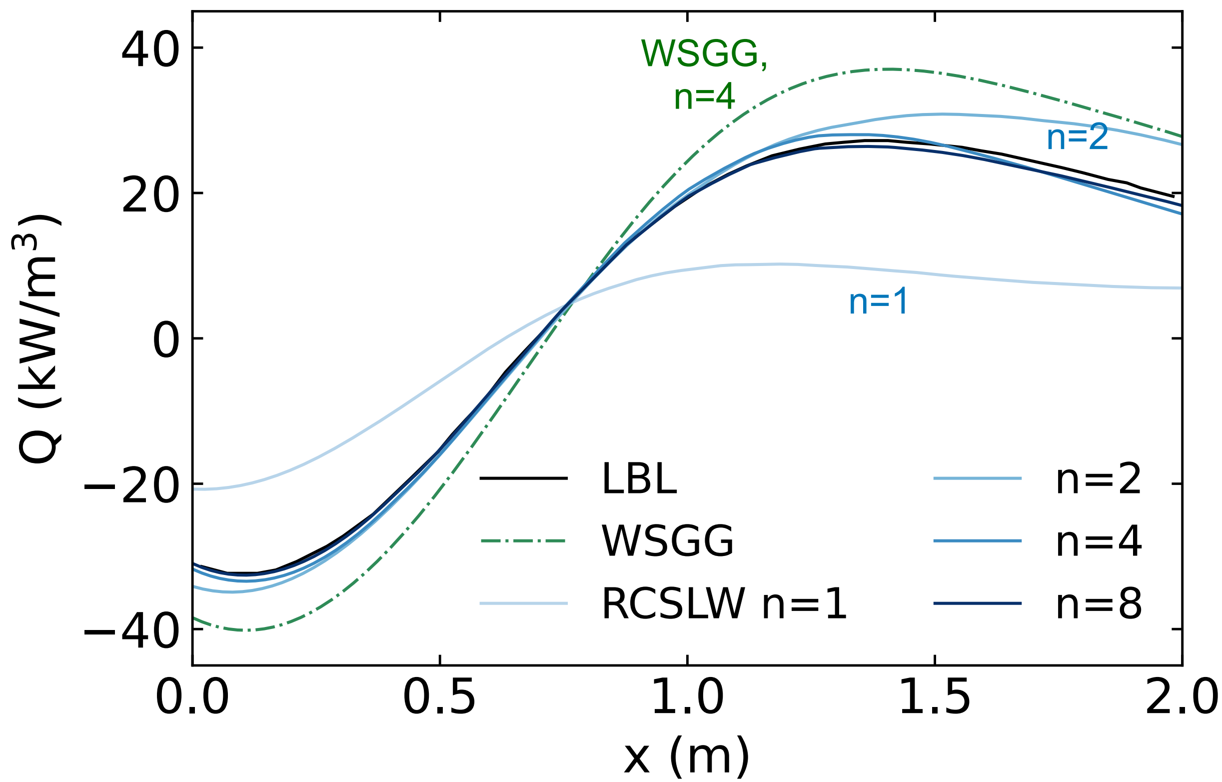
- PM gives poor results
- WSGG is good except near walls
- RCSLW is nearly LBL

Examples

Example S5

- Black walls at 1500, 500 K
- Uniform x_{H2O} = 0.1
- Vary # Gray Gases

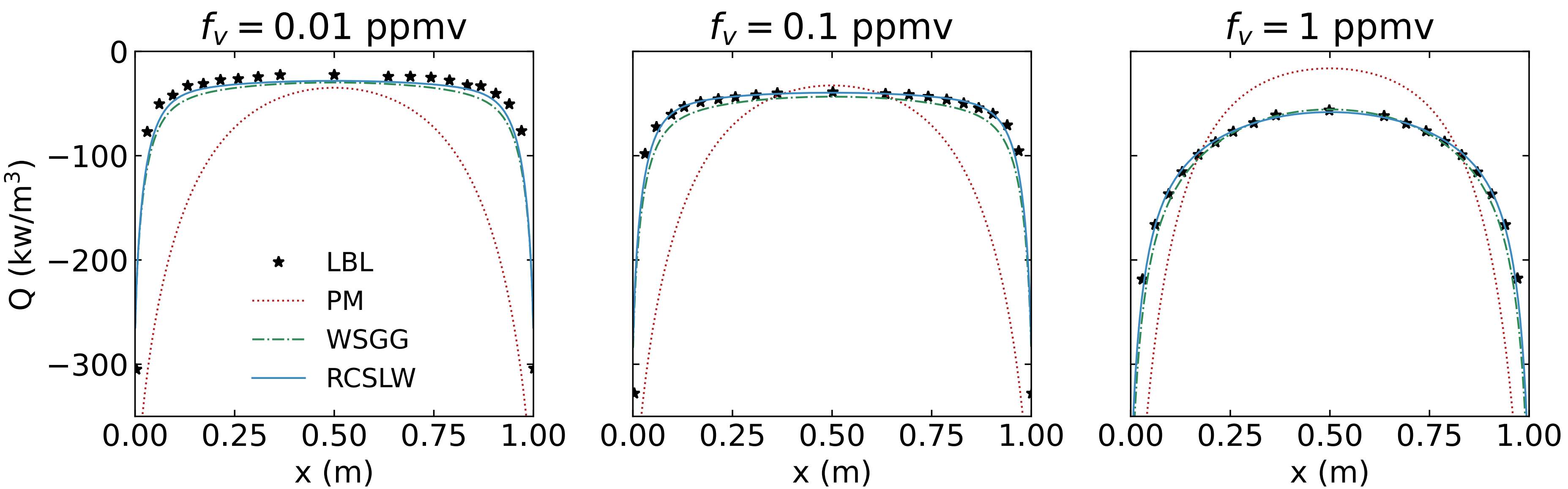




Examples

Example SB1

- Cold black walls
- Uniform T, composition
- T = 1000 K
- x_{H2O} = 0.2, x_{CO2} = 0.1



Publication

Computer Physics Communications 272 (2022) 108227

Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

RadLib: A radiative property model library for CFD ☆,☆☆

Victoria B. Stephens, Sally Jensen, Isaac Wheeler, David O. Lignell *

Department of Chemical Engineering, Brigham Young University, Provo, UT 84602, United States

A R T I C L E   I N F O

Article history:
Received 3 January 2021
Received in revised form 25 October 2021
Accepted 7 November 2021
Available online 12 November 2021

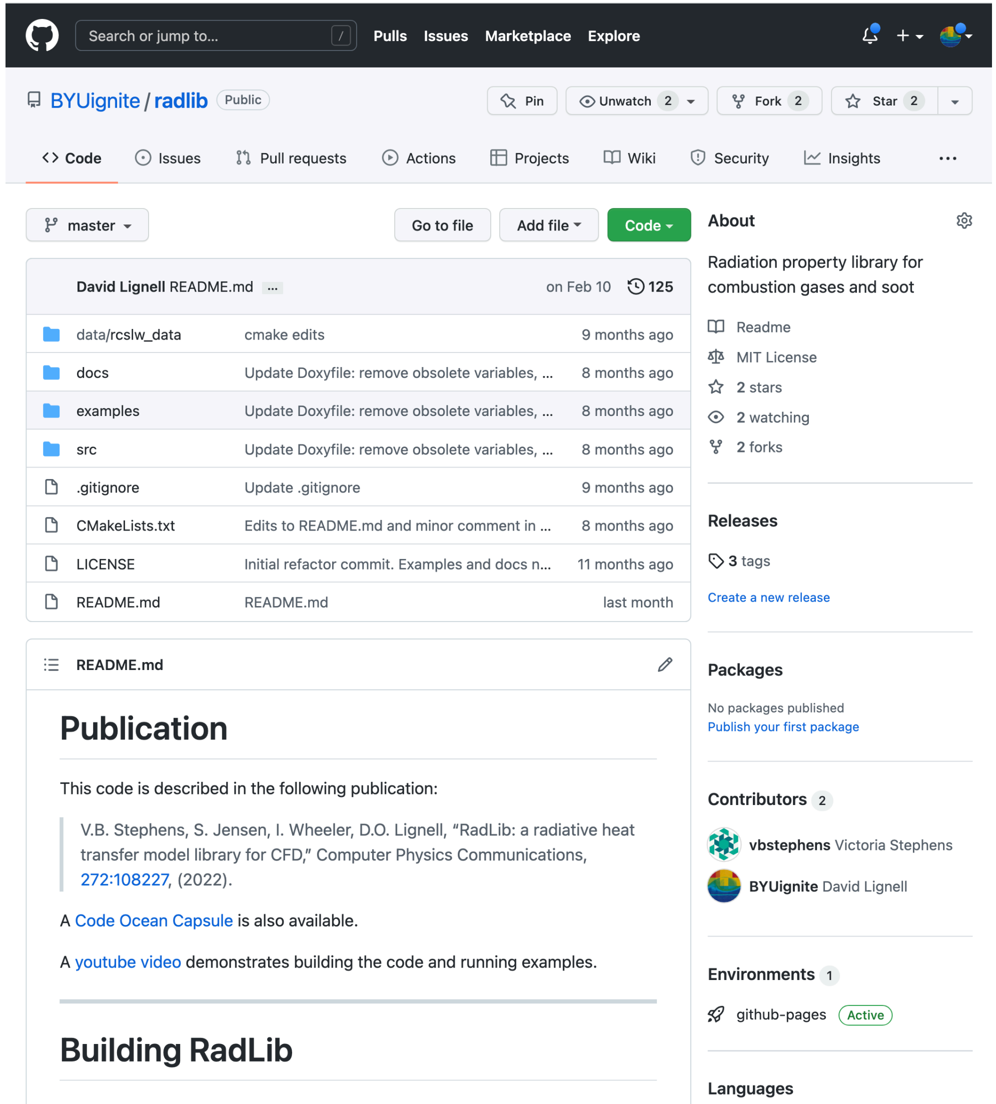
Keywords:
Radiative heat transfer
Reacting flows
CFD
WSGG
RCSLW

A B S T R A C T

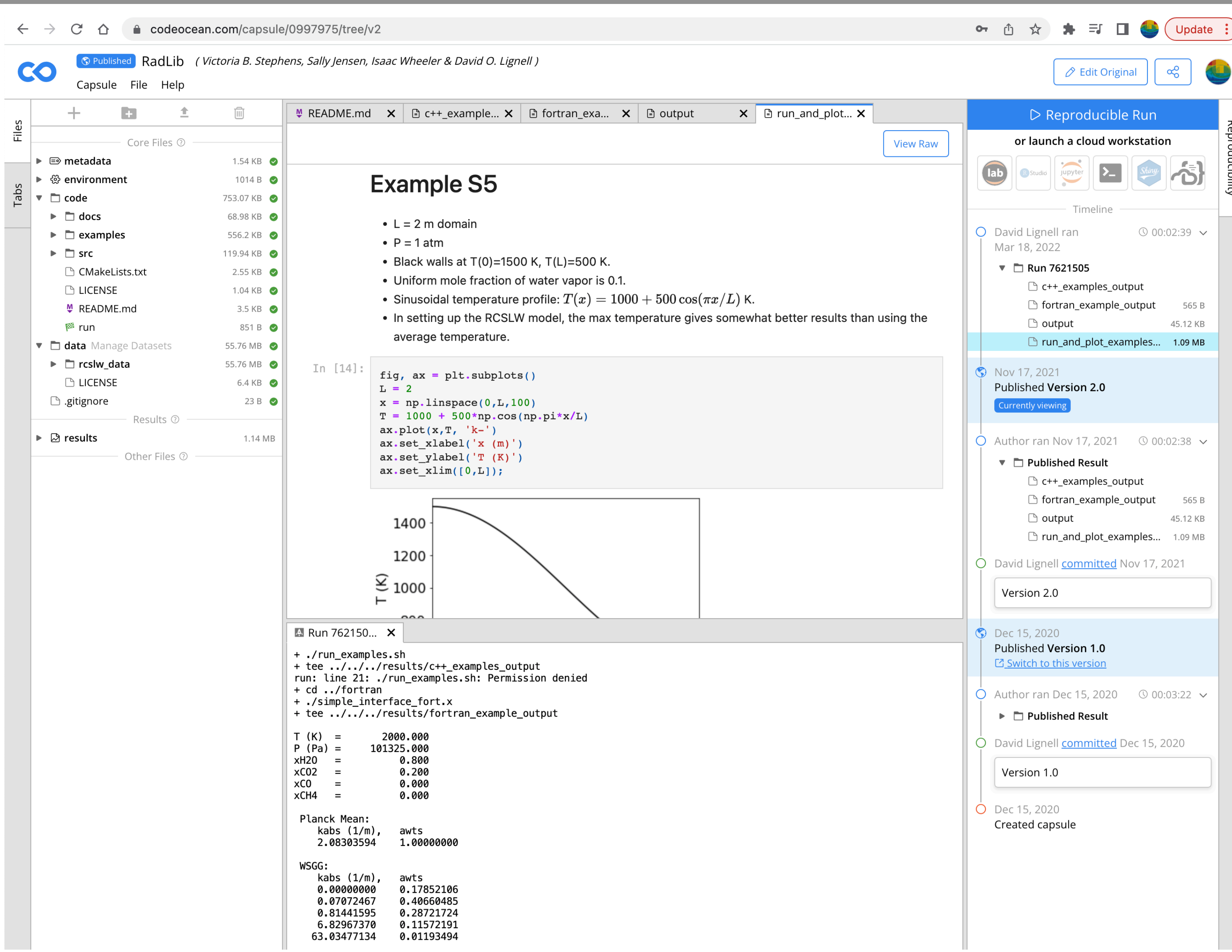
RadLib is a C++ library of radiation property models that can be applied to variety of systems involving radiative heat transfer, including CFD simulations. RadLib includes three major radiation property models—Planck Mean (PM) absorption coefficients, the weighted sum of gray gases (WSGG) model, and the rank-correlation spectral line weighted-sum-of-gray-gases (RCSLW) model. RadLib includes C++, Python, and Fortran interfaces and can be expanded to include additional models. Several example cases illustrate use of the models with an included ray-tracing solver, compare their accuracy relative to line-by-line (LBL) solutions, and examine their computational costs. Additionally, an integrated CFD example of an ethylene burner configuration using Fire Dynamics Simulator (FDS) is provided. RadLib provides researchers with convenient access to validated radiation property models and a framework for further development.

Program summary
Program Title: RadLib
CPC Library link to program files: `https://doi.org/10.17632/rs5kvnr86r.1`
Developer's repository link: `https://github.com/BYUignite/radlib`
Code Ocean capsule: `https://codeocean.com/capsule/0997975/tree`
Licensing provisions: MIT
Programming language: C++, Fortran, Python
Nature of problem: Radiative heat transfer in combustion CFD problems is typically neglected or oversimplified, resulting in significant error and inaccuracy in combustion simulations. Radiation modeling, however, often requires a high degree of specialization due to its complexity and the challenges associated with implementation in existing CFD codes, which presents a substantial obstacle to researchers who wish to address the inaccuracies introduced by insufficient radiative heat transfer modeling in combustion simulations.
Solution method: We present RadLib, an open-source C++ library of validated radiation property models that can be applied alongside various RTE solution methods to ease some of the obstacles associated with radiation modeling for combustion CFD. The package includes C++, Fortran, and Python interfaces, several illustrative examples with a provided ray-tracing solver and an integrated example using Fire Dynamics Simulator (FDS). RadLib can also be expanded to include additional radiation property models and interfaces.
Additional comments including restrictions and unusual features: The library is intended to be used in Linux-like terminal applications.

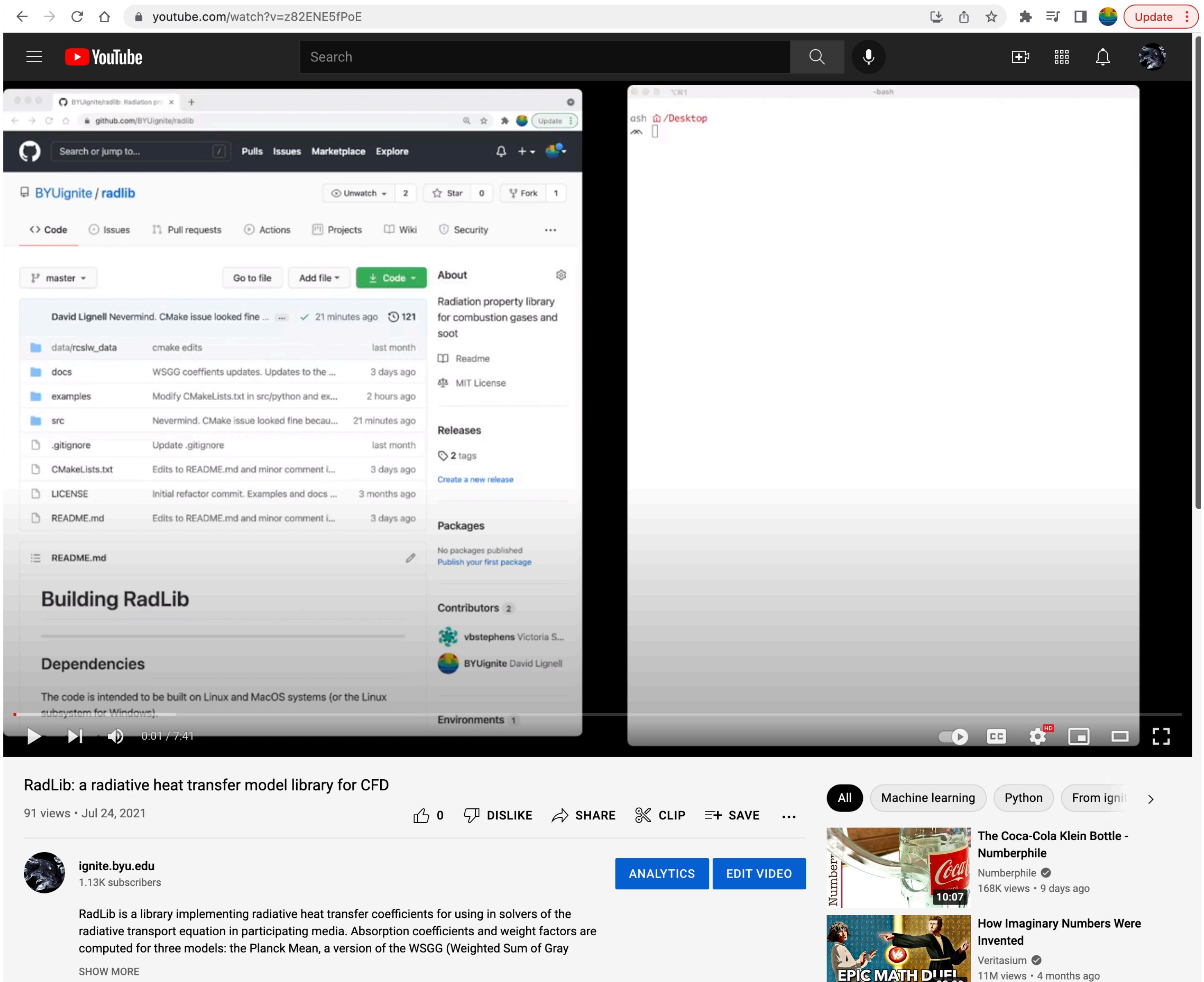
Github


Search or jump to...
Pulls
Issues
Marketplace
Explore
BYUignite / radlib
Public
Pin
Unwatch 2
Fork 2
Star 2
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
master
Go to file
Add file
Code
David Lignell README.md ...
on Feb 10
125
data/rcslw_data
cmake edits
9 months ago
docs
Update Doxyfile: remove obsolete variables, ...
8 months ago
examples
Update Doxyfile: remove obsolete variables, ...
8 months ago
src
Update Doxyfile: remove obsolete variables, ...
8 months ago
.gitignore
Update .gitignore
9 months ago
CMakeLists.txt
Edits to README.md and minor comment in ...
8 months ago
LICENSE
Initial refactor commit. Examples and docs n...
11 months ago
README.md
README.md
last month
README.md
Publication
This code is described in the following publication:
V.B. Stephens, S. Jensen, I. Wheeler, D.O. Lignell, “RadLib: a radiative heat transfer model library for CFD,” Computer Physics Communications, 272:108227, (2022).
A Code Ocean Capsule is also available.
A youtube video demonstrates building the code and running examples.
Building RadLib
About
Radiation property library for combustion gases and soot
Readme
MIT License
2 stars
2 watching
2 forks
Releases
3 tags
Create a new release
Packages
No packages published
Publish your first package
Contributors 2
vbstephens Victoria Stephens
BYUignite David Lignell
Environments 1
github-pages
Active
Languages

Code Ocean


codeocean.com/capsule/0997975/tree/v2
Update
Published RadLib (Victoria B. Stephens, Sally Jensen, Isaac Wheeler & David O. Lignell)
Capsule File Help
Edit Original
Files
Tabs
Core Files
metadata 1.54 KB
environment 1014 B
code 753.07 KB
docs 68.98 KB
examples 556.2 KB
src 119.94 KB
CMakeLists.txt 2.55 KB
LICENSE 1.04 KB
README.md 3.5 KB
run 851 B
data Manage Datasets 55.76 MB
rcslw_data 55.76 MB
LICENSE 6.4 KB
.gitignore 23 B
Results
results 1.14 MB
Other Files
README.md
c++_example...
fortran_exa...
output
run_and_plot...
View Raw
Example S5
L = 2 m domain
P = 1 atm
Black walls at T(0)=1500 K, T(L)=500 K.
Uniform mole fraction of water vapor is 0.1.
Sinusoidal temperature profile: $T(x) = 1000 + 500\cos(\pi x/L)$ K.
In setting up the RCSLW model, the max temperature gives somewhat better results than using the average temperature.
In [14]:
fig, ax = plt.subplots()
L = 2
x = np.linspace(0,L,100)
T = 1000 + 500*np.cos(np.pi*x/L)
ax.plot(x,T, 'k-')
ax.set_xlabel('x (m)')
ax.set_ylabel('T (K)')
ax.set_xlim([0,L]);
1400
1200
1000
T (K)
Run 762150...
+ ./run_examples.sh
+ tee ../../../results/c++_examples_output
run: line 21: ./run_examples.sh: Permission denied
+ cd ../fortran
+ ./simple_interface_fort.x
+ tee ../../../results/fortran_example_output
T (K) = 2000.000
P (Pa) = 101325.000
xH2O = 0.800
xCO2 = 0.200
xCO = 0.000
xCH4 = 0.000
Planck Mean:
kabs (1/m), awts
2.08303594 1.00000000
WSGG:
kabs (1/m), awts
0.00000000 0.17852106
0.07072467 0.40660485
0.81441595 0.28721724
6.82967370 0.11572191
63.03477134 0.01193494
Reproducible Run
or launch a cloud workstation
Reproducibility
Timeline
David Lignell ran Mar 18, 2022 00:02:39
Run 7621505
c++_examples_output
fortran_example_output 565 B
output 45.12 KB
run_and_plot_examples... 1.09 MB
Nov 17, 2021
Published Version 2.0
Currently viewing
Author ran Nov 17, 2021 00:02:38
Published Result
c++_examples_output
fortran_example_output 565 B
output 45.12 KB
run_and_plot_examples... 1.09 MB
David Lignell committed Nov 17, 2021
Version 2.0
Dec 15, 2020
Published Version 1.0
Switch to this version
Author ran Dec 15, 2020 00:03:22
Published Result
David Lignell committed Dec 15, 2020
Version 1.0
Dec 15, 2020
Created capsule

YouTube


youtube.com/watch?v=z82ENE5fPoE
Update
YouTube
Search
github.com/BYUignite/radlib
Search or jump to...
Pulls Issues Marketplace Explore
BYUignite / radlib
Unwatch 2 Star 0 Fork 1
Code Issues Pull requests Actions Projects Wiki Security
master
Go to file Add file Code
David Lignell Nevermind. CMake issue looked fine ... 21 minutes ago 121
data/rcslw_data cmake edits last month
docs WSGG coefficients updates. Updates to the ... 3 days ago
examples Modify CMakeLists.txt in src/python and ex... 2 hours ago
src Nevermind. CMake issue looked fine becau... 21 minutes ago
.gitignore Update .gitignore last month
CMakeLists.txt Edits to README.md and minor comment i... 3 days ago
LICENSE Initial refactor commit. Examples and docs ... 3 months ago
README.md Edits to README.md and minor comment i... 3 days ago
README.md
Building RadLib
Dependencies
The code is intended to be built on Linux and MacOS systems (or the Linux subsystem for Windows).
About
Radiation property library for combustion gases and soot
Readme
MIT License
Releases
2 tags
Create a new release
Packages
No packages published
Publish your first package
Contributors 2
vbstephens Victoria S...
BYUignite David Lignell
Environments 1
~/Desktop
0:01 / 7:41
RadLib: a radiative heat transfer model library for CFD
91 views • Jul 24, 2021
0 DISLIKE SHARE CLIP SAVE
ignite.byu.edu
1.13K subscribers
ANALYTICS EDIT VIDEO
RadLib is a library implementing radiative heat transfer coefficients for using in solvers of the radiative transport equation in participating media. Absorption coefficients and weight factors are computed for three models: the Planck Mean, a version of the WSGG (Weighted Sum of Gray
SHOW MORE
All Machine learning Python From igni
The Coca-Cola Klein Bottle - Numberphile
Numberphile
168K views • 9 days ago
10:07
How Imaginary Numbers Were Invented
Veritasium
11M views • 4 months ago
EPIC MATH DUEL
23:29

https://github.com/**BYUignite/radlib**

BRIGHAM YOUNG UNIVERSITY
FOUNDED
BYU
1875
PROVO, UTAH