The Journal of Open Source Software

lbh15: a Python package for standard use and implementation of physical data of heavy liquid metals used in nuclear reactors

Gabriele Ottino[1¶], **Daniele Panico**[1], **Daniele Tomatis**[1], and **Pierre-Alexandre Pantel**[2]

1 newcleo Srl, via Giuseppe Galliano 27, 10129 Torino, Italy **2** newcleo SA, 9 Rue des Cuirassiers, 69003 Lyon, France ¶ Corresponding author

DOI: 10.21105/joss.06383

Software

- Review
- Repository
- Archive

Editor: Kelly Rowland

Reviewers:

- @damar-wicaksono
- @LukeSeifert

Submitted: 08 January 2024
Published: 05 April 2024

Summary

lbh15 is a Python package that provides function correlations for the physical properties of the liquid metals used as coolant in GEN-IV liquid metal fast reactors (*LMFR*), such as those cooled by molten lead and lead-bismuth eutectic alloy. The package implements the correlations contained in the reference handbook edited by OECD/NEA (Fazio, 2015), also offering the possibility of adding new customized properties with minimal effort for the user. The properties of the liquid metal are uniquely defined by its thermodynamic state, namely by the temperature and pressure values. Alternatively, the physical properties can be used at the liquid metal object's instantiation, provided that the inverse of the corresponding correlation has at least one root in the validity range (*injective function* property).

lbh15 package is released under the *GNU Lesser General Public License v3.0*.

Statement of Need

Thermal-hydraulic analysis is a key factor for the design and safety studies of *LMFR*s, involving the implementation and use of several numerical methods and physical data that are employed in different computational tools. A standardization of the methods is necessary to guarantee homogeneity, reproducibility, and comparability of the numerical results. This standardization is particularly important considering the growing community of users with robust quality assurance needs. This is an essential point to ensure effective and successful projects in both industrial and research environments, especially for nuclear science and engineering. Additionally, *new*cleo pursues efforts for data standardization to develop new units of lead-cooled fast reactors (*LFR*).

In this context, standard libraries providing the correlations of physical properties for thermal-hydraulic computational tools are needed, such as CFD, system, and sub-channel codes concerning heavy liquid metals.

Implementation

lbh15 package takes inspiration from the iapws (Romera, 2021) Python package, which implements the water-related *IAPWS* full standard (International Association for the Properties of Water and Steam, 2018). However, lbh15 follows a different implementation approach.

The efficiency and the effectiveness are assured by the *Object-Oriented* design and the *Dynamic Loading* approach, which have been applied throughout the entire development process. lbh15 relies on the abstract liquid metal class; all classes describing the different metals inherit from it. The abstract class does not directly implement the property correlations, but it instead instantiates the property objects and provides the property values. In other words, the abstract liquid metal class acts as both *factory* of the property objects and *proxy* of the property values (Giridhar, 2016). This allows the user to add new custom properties without modifying the existing implementation of the liquid metal class.

Use

There are two main ways to use the package: either by instantiating a liquid metal object to access all its properties, or by instantiating an object for each specific property. The former approach provides a single entry point to all the liquid metal properties, which are evaluated at the specified thermodynamic state after checking that such state is valid (temperature between the melting and the boiling values, and positive pressure). In addition, this approach allows users to select the default correlations of the properties by means of the available class methods. The latter approach is best suited to cases where only a few specific properties are required for an individual thermodynamic state, since it offers faster instantiation and evaluation of the correlation functions.

Implemented Properties

The properties implemented so far can be subdivided into two groups:

- *thermo-physical*: saturation vapor pressure, surface tension, density, thermal expansion coefficient, speed of sound, isentropic compressibility, specific heat capacity, specific enthalpy, dynamic viscosity, electrical resistivity, thermal conductivity, and Prandtl number;
- *thermo-chemical*: diffusivity and solubility of oxygen and of the impurities in the liquid metals, oxygen partial pressure, molar enthalpy, molar entropy, Gibbs free energy, and oxygen concentration range assuring corrosion-protective oxide layer on metallic structure.

Implementation History

The release of version *1.1.0* of the package `lbh15` was described by Panico & Tomatis (2023). This version implemented only the thermo-physical properties.

The current version *2.1.0* implements the thermo-chemical properties and updates the documentation accordingly, improving the overall understanding. Moreover, solutions have been adopted to improve performance and usability of the code such as enforcing vectorization over the whole implementation and using the Horner scheme to evaluate polynomials (Hildebrand, 1974). Great attention is paid to the code quality. *PEP8* guidelines[1] are ensured by *pycodestyle*[2], and the automatic static analysis has been performed by applying *pylint*[3]. The documentation of the current version includes a tutorial focusing on a volume of lead that is subjected to time-varying thermal loads, where the oxygen concentration is controlled to fall in the range where the protective oxide layer formation is assured (Fazio, 2015).

The implementation of irradiated liquid metals' properties with new tutorials is planned as future improvement.

[1] https://www.python.org/dev/peps/pep-0008/ - *Style Guide for Python Code. PEP 8.* - G. van Rossum, B. Warsaw, and Coghlan - 2001

[2] https://pypi.org/project/pycodestyle/ - pycodestyle 2.11.1

[3] https://pypi.org/project/pylint/ - Pylint 3.1.0

Documentation

The documentation of `lbh15` is generated by `Sphinx` and published on `lbh15` *Github Pages* at the following address:

`https://newcleo-dev-team.github.io/lbh15/index.html`.

It is composed of parts addressed separately to the developers and to the users. The documentation contains examples for users, from basic use to short tutorials for more advanced applications.

Authors Contribution with CRediT

- Gabriele Ottino: Software, Validation, Writing - original draft, Writing - review & editing
- Daniele Panico: Conceptualization, Software, Supervision, Writing - review & editing
- Daniele Tomatis: Conceptualization, Project Administration, Supervision, Writing - original draft, Writing - review & editing
- Pierre-Alexandre Pantel: Writing - review & editing

References

Fazio, C. et al. (2015). *Handbook on lead-bismuth eutectic alloy and lead properties, materials compatibility, thermal-hydraulics and technologies*. OECD/Nuclear Energy Agency (NEA), Paris, France. https://doi.org/10.1787/42dcd531-en

Giridhar, C. (2016). *Learning python design patterns*. Packt Publishing.

Hildebrand, F. B. (1974). *Introduction to numerical analysis*. McGraw-Hill.

International Association for the Properties of Water and Steam. (2018). *Revised release on the IAPWS formulation 1995 for the thermodynamic properties of ordinary water substance for general and scientific use* (IAPWS R6-95(2018)).

Panico, D., & Tomatis, D. (2023). lbh15: A python package implementing lead, bismuth, and lead-bismuth eutectic thermophysical properties for fast reactor applications. *Proceedings of 20th International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-20), Washington DC, USA*, 1–12. https://doi.org/10.13182/nureth20-40554

Romera, J. J. G. (2021). *Jjgomera/iapws:* (Version v1.5.2). Zenodo. https://doi.org/10.5281/zenodo.4744318