Test Report: EOMEE

Gabriela Sánchez Díaz

December 12, 2020

1 Revision History

Date	Version	Notes
12-12-2020	1.0	First draft.

2 Symbols, Abbreviations and Acronyms

symbol	description
T	Test
EOMEE	Equation-of-motion for excited states
SRS	Software Requirements Specification
VnV	Verification and Validation
MG	Module Guide
MIS	Module Interface Specification

[symbols, abbreviations or acronyms – you can reference the SRS tables if needed —SS]

For more details about symbols, abbreviations and acronyms see Sections 1.2 and 1.4 in **?**.

Contents

List of Tables

List of Figures

This document gathers the outcomes of the functional and nonfunctional tests described in the System Verification and Validation Plan for EOMEE (VnV) [?]. It also serves the purpose of a to-do list of tasks that were pending to be completed within the time frame set by the course CAS741.

3 Functional Requirements Evaluation

Tests ST1-ST4 (involving the verification of the input formats and calculations) were designed to corroborate the functional requirements R1-R5 specified in the Software Requirements Specification (SRS) (see Section 5.1 in VnV and SRS respectively). All tests were completed successfully.

4 Nonfunctional Requirements Evaluation

4.1 Usability

Section 5.2.1 of the VnV Plan describes the usability test (ST6) [?]. This verification remains to be performed by some member of the review team (Section 4.1 of the VnV)

4.2 Accuracy

The verification of this nonfunctional requirement is covered by the system tests ST3, ST4 and ST8 (Sections 5.1.2 and 5.2.3 of the VnV [?]). The first two test cases were completed successfully, but ST8 remains to be done by Gabriela Sánchez-Díaz.

4.3 Reusability

The test designed to verify the modular decomposition of the project is described in the VnV Plan section 5.2.2 [?]. It's completion is left as a pending task for the author, however, a modification performed in the Integrlas modules during the unit testing (reported in Section 7) could be a preliminary validation of the design.

4.4 Portability

This requirement is addressed by test ST5 in the Section 5.2.1 of the VnV Plan [?]. The author installed the package in the Ubuntu 20.04 (Focal Fossa) platform following the installation instruction in the project's repository cas741. After installation the test module was ran. An issue that came up at this point is the relative paths of the data files declared in the input files. This problem is left ot be resolved by the Gabriela Sánchez-Díaz.

5 Comparison to Existing Implementation

To the author's knowledge there aren't existing implementations of EOMEE. However, there are electronic structure packages that offer similar features to EOMEE (exited states calculation) such as Gaussian [?] (commercial) and Psi4 [?] (open-source). Comparisons with the results given by these software remain to be done.

6 Unit Testing

All modules described in the Module Guide (MG) and Module Interface Specification (MIS) [?, ?] were tested with the exception of the Hardware-Hiding module (M1). The unit tests of EOMEE are implemented under the folder *tests/* and can be found in the project's GitHub repository cas741

7 Changes Due to Testing

No major issues were detected by the tests. The only change introduced was to facilitate the unit testing, in particular this involved the Integrals module (Section 8 of MIS). The valid input format of the module was extended to accept not only a file path, but also arrays.

This modification to the interface of the Integrals module didn't affect the remaining modules, therefore it could be considered as a positive indication of the modular decomposition of the project (related to the Reusability non-functional requirement), though this wasn't the actual test designed for this requirement (see section 5.2.2 in the VnV Plan [?]).

8 Automated Testing

The unit testing framework used was Pytest. Also, a Travis-CI .yaml file was set up for automated testing, however currently the buildup fails due to code formatting standards errors detected by the Python linters (specified in section 4.5 in the VnV Plan [?]).

9 Trace to Requirements

The original table for the trace between requirements and tests can be found in section 5.3 of the VnV Plan [?].

	ST1	ST2	ST3	ST4	ST5	ST6	ST7	ST8
R1	X							
R2		X						
R3		X						
R4			X	X				
R5			X	X				
Reusability							X	
Usability					X	X		
Portability					X			
Accuracy			X	X				X

Table 1: Traceability for system tests and requirements

10 Trace to Modules

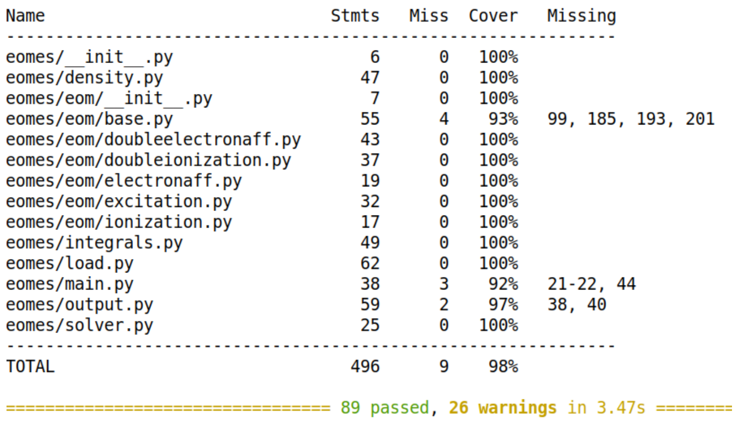
Table 2 shows the trace between EOMEE modules and unit tests. The modules outside of the unit testing scope are discussed in Section 6.1 of the VnV Plan [?].

	M1	M2	M3	M4	M5	M6-M11	M12	M13
UT1-UT2			X					
UT3-UT5				X				
UT6-UT9					X			
UT10							X	
UT11								X

Table 2: Traceability for modules and unit test sections

11 Code Coverage Metrics

Figure 1 shows the coverage statistics for EOMEE. The "Missing" column indicates lines of code left untested, these involve abstract methods of the EOM module interface and some conditional statements related to the selection of the EOM methods in the Control and Output modules. These lines were not deemed critical for the successful execution of the program and that is way were left uncovered. It remains as a future task to get 100% code coverage.



```
Name                              Stmts   Miss  Cover   Missing
---------------------------------------------------------------
eomes/__init__.py                     6      0   100%
eomes/density.py                     47      0   100%
eomes/eom/__init__.py                 7      0   100%
eomes/eom/base.py                    55      4    93%   99, 185, 193, 201
eomes/eom/doubleelectronaff.py       43      0   100%
eomes/eom/doubleionization.py        37      0   100%
eomes/eom/electronaff.py             19      0   100%
eomes/eom/excitation.py              32      0   100%
eomes/eom/ionization.py              17      0   100%
eomes/integrals.py                   49      0   100%
eomes/load.py                        62      0   100%
eomes/main.py                        38      3    92%   21-22, 44
eomes/output.py                      59      2    97%   38, 40
eomes/solver.py                      25      0   100%
---------------------------------------------------------------
TOTAL                               496      9    98%

================================ 89 passed, 26 warnings in 3.47s ========
```

Figure 1: Code Coverage of EOMEE tests