Verification and Validation Plan for Solar Water Heating Systems Incorporating Phase Change Material

Maya Grab and Brooks MacLachlan

June 3, 2016

Contents

1 General Information

The following section provides an overview of the Verification and Validation (V&V) Plan for a solar water heating system incorporating phase change material simulator. This section explains the purpose of this document, the scope of the system, common definitions, acronyms and abbreviations that are used in the document, and an overview of the following sections.

1.1 Purpose

The main purpose of this document is to describe the verification and validation process that will be used to test a simulation for solar water heating systems incorporating PCM. This document is intended to be used as a reference for all future testing and will be used to increase confidence in the software implementation.

This document will be used as a starting point for the verification and validation report. The test cases presented within this document will be executed and the output will be analyzed to determine if the software is implemented correctly.

1.2 Scope

[No content... should this be here? —BM]

1.3 Acronyms, Abbreviations, and Symbols

symbol	description
QA	Quality assurance
SRS	Software requirements specification
V&V	Verification and validation
V&VP	Verification and validation plan
V&VR	Verification and validation report
PCM	Phase change material
SWHS	Solar Water Heating System
ϵ	10^{-2}

[The value for epsilon in the actual Matlab scripts is different. —BM]

1.4 Overview of Document

The following sections provide more detail about the V&V of a solar water heating simulator. Information about the testing process is provided, and the software specifications that were discussed in the SRS document are stated. The evaluation process that will be followed during testing is outlined, and test cases for both the system testing and unit testing are provided.

2 Plan

This section provides a description of the software that is being tested, the team that will perform the testing, the milestones for the testing phase, and the budget allocated to the testing.

2.1 Software Description

The software being tested is a simulator for a SWHS incorporating PCM. Given the physical parameters of the system, including dimensions, properties of the water and PCM, and relevant physical constants, the simulator calculates the changes in temperature and energy of the water and PCM over time.

2.2 Test Team

The team that will execute the test cases, write and review the V&VR consists of:

- Maya Grab

- Dr. Spencer Smith
- Thulasi Jegatheesan

2.3 Milestones

2.3.1 Location

The location where the testing will be performed is Hamilton, Ontario. The institution that will be performing the testing is McMaster University.

2.3.2 Dates and Deadlines

Test Case:
The creation of the test cases for both system testing and unit testing is scheduled to begin June 1st 2015. The deadline for the creation of the test cases is June 15th 2015.

Test Case Implementation:
Implementing code for the automation of the unit testing is scheduled to begin June 15th 2015. The implementation period is expected to last approximately two weeks and has a deadline of June 30th 2015.

Verification and Validation Report:
The writing of the V&VR is scheduled to begin July 1st 2015 and end on July 15th 2015.

2.4 Budget

The budget for the testing of this system is being funded by McMaster University and NSERC.

3 Software Specification

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

3.1 Functional Requirements

- Input the physical constants, properties and initial temperatures of water and PCM, and dimensions of the tank
- Verify that the inputs satisfy the required physical constraints

- Compute the calculated values required to solve the governing differential equations
- Calculate the temperatures and energy of water and PCM over time.

3.2 Nonfunctional Requirements

Priority nonfunctional requirements are correctness, understandability, reliability, and maintainability.

4 Evaluation

This section first presents the methods and constraints that are to be used during the evaluation process. This is followed by how the data obtained by the testing will be evaluated, which includes: how the data will be recorded, how to move from one test to the next, and how to determine if the test was successful.

4.1 Methods and Constraints

4.1.1 Methodology

The testing of the SWHS will be fully automated with the exception of one testing method, where a change to the ODE solver algorithm is required.

4.1.2 Extent of Testing

The extent of testing that will be employed is extensive. The unit test cases below provide complete code coverage and will increase confidence in the verification of the software. The system test cases increase confidence in the validation of the system.

4.1.3 Test Tools

A unit testing framework will be used to implement the unit test cases and run them automatically.

The following equation will be implemented in a script in order to compare the outputs of different implementation:

$$\Delta_{\text{relative}} = \frac{\text{True} - \text{Calculated}}{\text{True}}$$

4.1.4 Testing Constraints

There are currently no anticipated limitations on the testing.

4.2 Data Evaluation

4.2.1 Testing Criteria

Test criteria are divided into two categories:

1. **Numerical**: testing results will be compared to expected results within an allowable margin of error ϵ.

2. **Error catching**: testing will ensure that faulty and unrecommended inputs are caught in exceptions and warnings.

4.2.2 Testing Data Reduction

The results of the test data will be evaluated on a PASS/FAIL basis. If the actual results match the expected results the test will be considered a PASS, otherwise the test is considered a FAIL.

5 System Test Description

5.1 Faulty Input

5.1.1 Input

The input will be based on the Data Constraints on Input table provided in the appendix of this document (borrowed from Input Variables table in the SRS document). Each test will correspond to one entry from the physical constraints column, altering a specific input variable to a non-permissible value. The list of inputs is in order with the entries in the table, though note there are several cases tested for each constraint described.

Table 1: Faulty Input Cases

No.	Input	Expected Outcome	MsgID
01	$L = -2$	error: Tank length must be > 0	input:L
02	$L = 0$	error: Tank length must be > 0	input:L
03	$D = -2$	error: Tank diameter must be > 0	input:diam
04	$D = 0$	error: Tank diameter must be > 0	input:diam
05	$V_P = -0.05$	error: PCM volume must be > 0	input:Vp

06	$V_P = 0$	error: PCM volume must be > 0	input:Vp
07	$L = 0.5$ $D = 0.5$ $Vp = 0.5$	error: Tank volume must be $>$ PCM volume	input:VpVt
08	V_P $= 0.199974938771605$ (tank volume) A_P $= 2.208137511613965$ (tank surface area)	error: Tank volume must be $>$ PCM volume	input:VpVt
09	$A_P = -1.5$	error: PCM area must be > 0	input:Ap
10	$A_P = 0$	error: PCM area must be > 0	input:Ap
11	$\rho_P = -1000$	error: rho_p must be > 0	input:rho_p
12	$\rho_P = 0$	error: rho_p must be > 0	input:rho_p
13	$T^P_{\text{melt}} = -10$	error: Tmelt must be > 0 and $<$ Tc	input:Tmelt
14	$T^P_{\text{melt}} = 0$	error: Tmelt must be > 0 and $<$ Tc	input:Tmelt
15	$T^P_{\text{melt}} = 45$ $T_C = 40$	error: Tmelt must be > 0 and $<$ Tc	input:Tmelt
16	$C^S_P = -1000$	error: C_ps must be > 0	input:C_ps
17	$C^S_P = 0$	error: C_ps must be > 0	input:C_ps
18	$C^L_P = -1000$	error: C_pl must be > 0	input:C_pl
19	$C^L_P = 0$	error: C_pl must be > 0	input:C_pl
20	$H_f = -200000$	error: Hf must be > 0	input:Hf
21	$H_f = 0$	error: Hf must be > 0	input:Hf
22	$A_C = -0.12$	error: Ac must be > 0	input:Ac

23	$A_C = 0$	error: Ac must be > 0	input:Ac
24	$T_C = -50$	error: Tmelt must be > 0 and < Tc	input:Tmelt
25	$T_C = 0$	error: Tmelt must be > 0 and < Tc	input:Tmelt
26	$T_C = 100$	error: Tc must be > 0 and < 100	input:Tc
27	$T_C = 110$	error: Tc must be > 0 and < 100	input:Tc
28	$\rho_W = -1000$	error: rho_w must be > 0	input:rho_w
29	$\rho_W = 0$	error: rho_w must be > 0	input:rho_w
30	$C_W = -4000$	error: C_w must be > 0	input:C_w
31	$C_W = 0$	error: C_w must be > 0	input:C_w
32	$h_C = -1000$	error: hc must be > 0	input:hc
33	$h_C = 0$	error: hc must be > 0	input:hc
34	$h_P = -1000$	error: hp must be > 0	input:hp
35	$h_P = 0$	error: hp must be > 0	input:hp
36	$T_{\text{init}} = -5$	error: Tinit must be > 0 and < 100	input:Tinit
37	$T_{\text{init}} = 0$	error: Tinit must be > 0 and < 100	input:Tinit
38	$T_{\text{init}} = 100$	error: Tc must be > Tinit	input:TcTinit
39	$T_{\text{init}} = 110$	error: Tc must be > Tinit	input:TcTinit
40	$T_{\text{init}} = 45$	error: Tinit must be < Tmelt	input:TinitTmelt
41	$T_{\text{init}} = 50$	error: Tc must be > Tinit	input:TcTinit
42	$T_{\text{init}} = 60$	error: Tc must be > Tinit	input:TcTinit
43	$t_{\text{final}} = 0$	error: tfinal must be > 0	input:tfinal

44	$t_{\text{final}} = -50000$	error: tfinal must be > 0	input:tfinal

5.1.2 Preparation and Procedure

Input files are to be generated for each test case and be stored in an appropriate directory. The automated test should include a setup procedure that would add the directory to the Matlab path. Each test case runs main.m on an input file and checks for the appropriate MsgId.

5.2 Unrecommended Input

5.2.1 Input

The input will be based on the Data Constraints on Input table provided in the appendix of this document (borrowed from Input Variables table in the SRS document). Each test will correspond to one entry from the physical constraints column, altering a specific input variable to a non-advisable value. The list of inputs is in order with the entries in the table, though note there are several cases tested for each constraint described.

Table 2: Unrecommended Input Cases

No.	Input	Expected Outcome	MsgID
01	$L = 0.01$ $V_P = 0.001$	It is recommended that 0.1 <= L <= 50	inputwarn:L
02	$L = 55$	It is recommended that 0.1 <= L <= 50	intputwarn:L
03	$L = 30$ $D = 0.03$ $V_P = 0.001$	It is recommended that 0.002 <= D/L <= 200	inputwarn:diam
04	$D = 400$	It is recommended that 0.002 <= D/L <= 200	inputwarn:diam
05	$L = 15$ $D = 4.12$ $V_P = 0.00006$	It is recommended that Vp be >= 0.0001% of Vt	inputwarn:VpVt
06	$A_P = 0.04$	It is recommended that Vp <= Ap <= 2*Vp/0.001	inputwarn:VpAp

07	$A_P = 110$	It is recommended that Vp <= Ap <= 2*Vp/0.001	inputwarn:VpAp
08	$\rho_P = 450$	It is recommended that 500 < rho_p < 20000	inputwarn:rho_p
09	$\rho_P = 20005$	It is recommended that 500 < rho_p < 20000	inputwarn:rho_p
10	$C_P^S = 90$	It is recommended that 100 < C_ps < 4000	inputwarn:C_ps
11	$C_P^S = 5000$	It is recommended that 100 < C_ps < 4000	inputwarn:C_ps
12	$C_P^L = 90$	It is recommended that 100 < C_pl < 5000	inputwarn:C_lp
13	$C_P^L = 5005$	It is recommended that 100 < C_pl < 5000	inputwarn:C_pl
14	$H_f = \min$		
15	$H_f = \max$		
16	$A_C = 0.7$	It is recommended that Ac <= pi*D/2	inputwarn:Ac
17	$\rho_W = 900$	It is recommended that 950 < rho_w <= 1000	inputwarn:rho_w
18	$\rho_W = 1010$	It is recommended that 950 < rho_w <= 1000	inputwarn:rho_w
19	$C_W = 4160$	It is recommended that 4170 < C_w < 4210	inputwarn:C_w
20	$C_W = 4220$	It is recommended that 4170 < C_w < 4210	inputwarn:C_w
21	$h_C = 9$	It is recommended that 10 < hc < 10000	inputwarn:hc
22	$h_C = 10001$	It is recommended that 10 < hc < 10000	inputwarn:hc
23	$h_P = 9$	It is recommended that 10 < hp < 10000	inputwarn:hp
24	$h_P = 10001$	It is recommended that 10 < hp < 10000	inputwarn:hp
25	$t_{\text{final}} = 86500$	It is recommended that 0 < tfinal < 86400	inputwarn:tfinal

5.2.2 Preparation and Procedure

Input files are to be generated for each test case and be stored in an appropriate directory. The automated test should include a setup procedure that would add the directory to the Matlab path. Each test case runs main.m on an input file and checks for the appropriate MsgId.

5.3 Alternative Algorithm

5.3.1 Means of Control

The program will be manually compared to an implementation substituting the ode45 algorithm with ode23.

5.3.2 Input

The input will be the standard working input, run once through the ode45 implementation and once through the ode23 implementation.

5.3.3 Expected Output

The delta between the output vectors must be $< \epsilon$.

5.3.4 Preparation and Procedure

The ode45 function will be manually substituted with ode23 algorithm in main.m in order to compare the implementations. Both functions will be run on the input file and results will be compared.

5.4 Comparison to Original Implementation

5.4.1 Means of Control

Valid output from the current Matlab implementation will be compared to output from the original Fortran implementation.

5.4.2 Input

The standard input file (outlined in Table 12) for Matlab and the corresponding file in Fortran format will be run through their respective implementations.
The standard input file will be modified to five different variations, as described in the table below.

Table 3: Comparison to Original Implementation Cases

No.	Purpose	Δ Input
01		Standard
02	Set coil temperature just above melting temperature of PCM.	$T_{\mathrm{C}} = 44.21°C$ $T_{\mathrm{melt}} = 44.2°C$
03	Set t at exactly the initial melting time according to No.1 (Fortran implementation).	$t = 20570$s
04	Set t just above the initial melting time.	$t = 20580$s
05	Remove PCM	$V_P = 10^-7$ $A_P = 10^-7$ $\rho_P = 1$

5.4.3 Expected Output

The delta between the output vectors should be $< \epsilon$.

5.4.4 Procedure

5.4.5 Preparation

[Should these sections be blank? —BM]

6 Temperature Modules

6.1 Module Information

This testing suite will test the three temperature modules handling the governing differential equations for temperature of water and temperature of PCM.

- Temperature1.m handles the case where $T_{\mathrm{P}} < T_{\mathrm{melt}}$.
- Temperature2.m handles the case where $T_{\mathrm{P}} = T_{\mathrm{melt}}$.
- Temperature3.m handles the case where $T_{\mathrm{P}} > T_{\mathrm{melt}}$.

6.1.1 Module Inputs

The Temperature modules take as input:

- a time t;
- a temperature vector T where $T(1) = T_{\mathrm{W}}$ and $T(2) = T_{\mathrm{P}}$;
- an input parameters structure containing the typical simulator parameters

6.1.2 Module Outputs

The Temperature modules output a vector dTdt where:

- dTdt(1) $= \frac{dT_{\mathrm{W}}}{dt}$
- dTdt(2) $= \frac{dT_{\mathrm{P}}}{dt}$
- dTdt(3) $= \frac{dQ_{\mathrm{P}}}{dt}$ (temperature2.m only)

6.1.3 Related Modules

The module load_params.m was used in testing in order to load the input parameters structure into each function.

6.2 Test Data

6.2.1 Inputs

Table 4: Temperature Tests Input

Test	Input
temperature 1	$t = 100$ $T = [40.7, 40.5]$ params
temperature 2a	$t = 3000$ $T = [44.2, 44.2]$ params
temperature 2b	$t = 4000$ $T = [45, 44.2]$ params
temperature 3	$t = 25000$ $T = [47, 46.5]$ params

6.2.2 Expected Outputs

Table 5: Temperature Tests Expected Outputs

Test	Input
temperature 1	[0.00139536, 0.002708315]
temperature 2a	[0.001108642, 0, 0]
temperature 2b	[-0.000573435, 0, 960]
temperature 3	[-0.00038229, 0.005249596]

7 Energy Modules

7.1 Module Information

This testing suite will test the three energy modules handling the governing equations for energy of water and PCM.

- energy1.m handles the case where $T_{\mathrm{P}} < T_{\mathrm{melt}}$.
- energy2.m handles the case where $T_{\mathrm{P}} = T_{\mathrm{melt}}$.
- energy3.m handles the case where $T_{\mathrm{P}} > T_{\mathrm{melt}}$.

7.1.1 Module Inputs

The Energy modules take as input:

- a temperature matrix T where $T(:,1) = T_{\mathrm{W}}$, $T(;,2) = T_{\mathrm{P}}$ and (energy2 only) $T(:,3) = Q_{\mathrm{P}}$
- an input parameters structure containing the typical simulator parameters

7.1.2 Module Outputs

The Energy modules output two vectors: E_{W} and E_{P}.

7.2 Test Data

7.2.1 Inputs

Table 6: Energy Tests Input

Test	Input
energy 1	$T = [40:44; 40:44]'$ params
energy 2	$T = [44.2:0.1:44.6; 44.2, 44.2, 44.2, 44.2, 44.2; 372000:51000:576000]'$ params
energy 3	$T = [45:49; 45:49]'$ params

[In these tables throughout the document, values are expressed using Matlab's syntax. Is that okay, or do we want this document to be completely independent from the choice of software?" —BM]

7.2.2 Expected Outputs

Table 7: Energy Tests Output

Test	E_{W}	E_{P}
energy 1	0 627795.0938 1255590.188 1883385.281 2511180.375	0 88616 177232 265848 354464
energy 2	2636739.394 2699518.903 2762298.413 2825077.922 2887857.431	744187.2 795187.2 846187.2 897187.2 948187.2
energy 3	3138975.469 3766770.563 4394565.657 5022360.75 5650155.844	11117682.8 11231977.3 11346271.8 11460566.3 11574860.8

8 Event Modules

8.1 Module Information

This testing suite will test the two Event modules, which handle the switch between the cases $T_{\mathrm{P}} < T_{\mathrm{melt}}$, $T_{\mathrm{P}} = T_{\mathrm{melt}}$ and $T_{\mathrm{P}} > T_{\mathrm{melt}}$.

8.1.1 Module Inputs

The Event modules take as input:

- a time t;
- a temperature vector T where $T(1) = T_{\mathrm{W}}$, $T(2) = T_{\mathrm{P}}$ and (event 2 only) $T(3) = Q_{\mathrm{P}}$;
- an input parameters structure containing the typical simulator parameters

8.1.2 Module Outputs

The Event modules output three values: [value, isterminal, direction].

8.1.3 Related Modules

The module load_params.m was used in testing in order to load the input parameters structure into each function.

8.2 Test Data

8.2.1 Inputs

Table 8: Event Tests Input

Test	Input
event 1a	$t = 100$ $T = [41, 40.9]$ params
event 1b	$t = 3000$ $T = [44.2, 44.2]$ params
event 2a	$t = 3000$ $T = [44.2, 44.2, 0]$ params
event 2b	$t = 4000$ $T = [45, 44.2, 600000]$ params
event 2c	$t = 20570$ $T = [44.7, 44.2, 10654060]$ params

8.2.2 Expected Outputs

Table 9: Event Tests Expected Output

Test	Output
event 1a	[3.3, 1, 0]
event 1b	[0, 1, 0]
event 2a	[-1, 1, 0]
event 2b	[-0.943683441, 1, 0]
event 2c	[0, 1, 0]

9 Output Verification Module

9.1 Module Information

This testing suite will test the Output Verification module, which verifies that the energy outputs obey the law of conservation of energy.

9.1.1 Module Inputs

The Output Verification module takes as input:

- an input parameters structure containing the typical simulator parameters;
- a vector of times, t;
- a matrix with at least two columns of temperature values, T where $T(:,1) = T_{\mathrm{W}}$, and $T(:,2) = T_{\mathrm{P}}$;
- a vector of values for energy in the water, Ew;
- a vector of values for energy in the PCM, Ep;

9.1.2 Module Outputs

The Output Verification Module has no outputs.

9.1.3 Related Modules

The module load_params.m was used in testing in order to load the input parameters structure into each function.

9.2 Test Data

9.2.1 Input

Table 10: Output Verification Tests Input

Test No.	Input
1	params t = [0;10;20;30] T = [40 40; 42 41.9; 44 43.8; 46 45.7;] Ew = [0;1000;2000;19800] Ep = [0;1000;2000;5400]
2	params t = [0;10;20;30] T = [40 40; 42 41.9; 44 43.8; 46 45.7;] Ew = [0;1000;2000;19800] Ep = [0;1000;2000;3000]
3	params t = [0;10;20;30] T = [40 40; 42 41.9; 44 43.8; 46 45.7;] Ew = [0;1000;2000;3000] Ep = [0;1000;2000;5400]
4	params t = [0;10;20;30] T = [40 40; 42 41.9; 44 43.8; 46 45.7;] Ew = [0;1000;2000;3000] Ep = [0;1000;2000;3000]

9.2.2 Expected Output

Table 11: Output Verification Tests Expected Output

Test No.	Expected Warning	MsgID
1	None	N/A
2	'There is greater than 0.00001 relative error between the Ep output and the expected output based on the law of conservation of energy'	'output:Ep'

3	'There is greater than 0.00001 relative error between the Ew output and the expected output based on the law of conservation of energy'	'output:Ew'
4	'There is greater than 0.00001 relative error between the Ew output and the expected output based on the law of conservation of energy' and 'There is greater than 0.00001 relative error between the Ep output and the expected output based on the law of conservation of energy'	'output:Ew' and 'output:Ep'

10 Appendix

Table 12: Standard Input Variables

Var	**Typical Value**
L	1.5 m
D	0.412 m
V_P	0.05 m^3
A_P	1.2 m^2
ρ_P	1007 $\mathrm{kg/m}^3$
T_{melt}^P	44.2 °C
C_P^S	1760 J/(kg °C)
C_P^L	2270 J/(kg °C)
H_f	211600 J/kg
A_C	0.12 m^2
T_C	50 °C
ρ_W	1000 $\mathrm{kg/m}^3$
C_W	4186 J/(kg °C)
h_C	1000 W/(m^2 °C)
h_P	1000 W/(m^2 °C)
T_{init}	40 °C
t_{final}	50000 s
AbsTol	10^{-10}
RelTol	10^{-10}

Table 13: Data Constraints on Input

Var	**Physical Constraints**
L	$L > 0$
D	$D > 0$
V_P	$V_P > 0$ (*)
	$V_P < \pi(D/2)^2 L$
A_P	$A_P > 0$ (*)
ρ_P	$\rho_P > 0$
T_{melt}^P	$0 < T_{\text{melt}}^P < T_C$
C_P^S	$C_P^S > 0$
C_P^L	$C_P^L > 0$
H_f	$H_f > 0$
A_C	$A_C > 0$ (*)
T_C	$T_C > 0$ (+)
ρ_W	$\rho_W > 0$
C_W	$C_W > 0$
h_C	$h_C > 0$
h_P	$h_P > 0$
T_{init}	$0 < T_{\text{init}} < T_{\text{melt}}$ (+)
t_{final}	$t_{\text{final}} > 0$

(*) These quantities cannot be equal to zero, or there will be a divide by zero in the model.

(+) These quantities cannot be zero, or there would be freezing.