Asphalt Concrete Thermal Stress Calculation (ACTS Calc) Software

Documentation and Tutorial

This software is the result of the application and optimization of earlier studies about pavement's temperature profile and pavement's thermal-stresses calculation. This tool was realized on "Jupyter" using a "python kernel", and then translated into a Graphical User Interface (GUI) developed with the "DearPyGui" library (https://github.com/hoffstadt/DearPyGui). The goal is to estimate the thermal stresses due to the environment's temperature oscillation.

In this "pre-release" version, the pavement thermal profile is calculated using a complex model, which includes the effect of (Gui, et al., 2007):

- Radiation: incoming solar energy and outcoming infrared radiation (albedo)
- Convection: heat exchanges with air, considering wind velocity and the occurrence of turbulence (if needed)
- Conduction: heat transfer into the ground (semi-infinite solid)

The implied model allows the obtention of an accurate thermal profile because the implementation of the different physical phenomena accounting for heat exchanges between the asphalt and its surroundings (Gui, et al., 2007). Thus, the thermal stresses results configure a better approximation than an eventual one based on a simplified thermal model

In the beginning, the software presents a "start page", on this, the "readme" and "about" sections are available on top of the screen. In the bottom of the page, the user can select a full calculation of thermal profile + thermal stresses or only the study of the thermal profile. To start the analysis, the user must pick in the start button. Figure 1 presents the start page of the software.

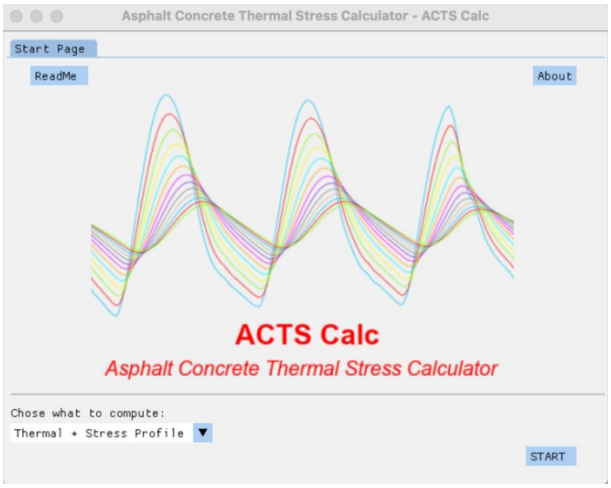


Figure 1. Start Page

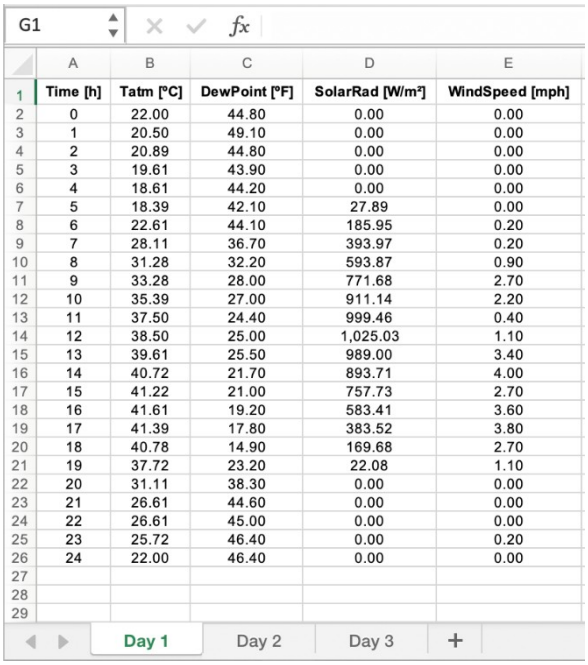
The different steps necessary to obtain the asphalt-pavement's stress profile are described below, alongside all mathematical equations and scientific background utilized.

I. Thermal Model

a. Thermal Model Inputs

To properly run the physical thermal model, various inputs are needed. First, the user is asked to fill an Excel table called "weather.xlsx" with weather data. This table needs meteorological measurements between 1 to 3 days (refer Figure 2). Each day must be filled in a separated tab of the file. The required data are:

- The measured time stamp [h]
- The atmospheric temperature [ºC]
- The dew point temperature [ºF]
- The solar irradiance [W/m²]
- The wind velocity [mph]



G1

	A	B	C	D	E
1	Time [h]	Tatm [ºC]	DewPoint [ºF]	SolarRad [W/m²]	WindSpeed [mph]
2	0	22.00	44.80	0.00	0.00
3	1	20.50	49.10	0.00	0.00
4	2	20.89	44.80	0.00	0.00
5	3	19.61	43.90	0.00	0.00
6	4	18.61	44.20	0.00	0.00
7	5	18.39	42.10	27.89	0.00
8	6	22.61	44.10	185.95	0.20
9	7	28.11	36.70	393.97	0.20
10	8	31.28	32.20	593.87	0.90
11	9	33.28	28.00	771.68	2.70
12	10	35.39	27.00	911.14	2.20
13	11	37.50	24.40	999.46	0.40
14	12	38.50	25.00	1,025.03	1.10
15	13	39.61	25.50	989.00	3.40
16	14	40.72	21.70	893.71	4.00
17	15	41.22	21.00	757.73	2.70
18	16	41.61	19.20	583.41	3.60
19	17	41.39	17.80	383.52	3.80
20	18	40.78	14.90	169.68	2.70
21	19	37.72	23.20	22.08	1.10
22	20	31.11	38.30	0.00	0.00
23	21	26.61	44.60	0.00	0.00
24	22	26.61	45.00	0.00	0.00
25	23	25.72	46.40	0.00	0.20
26	24	22.00	46.40	0.00	0.00
27					
28					
29					

Day 1 | Day 2 | Day 3 | +

Figure 2. Meteorological inputs in Excel

The user can validate if the information was uploaded correctly in the software (refer Figure 3).

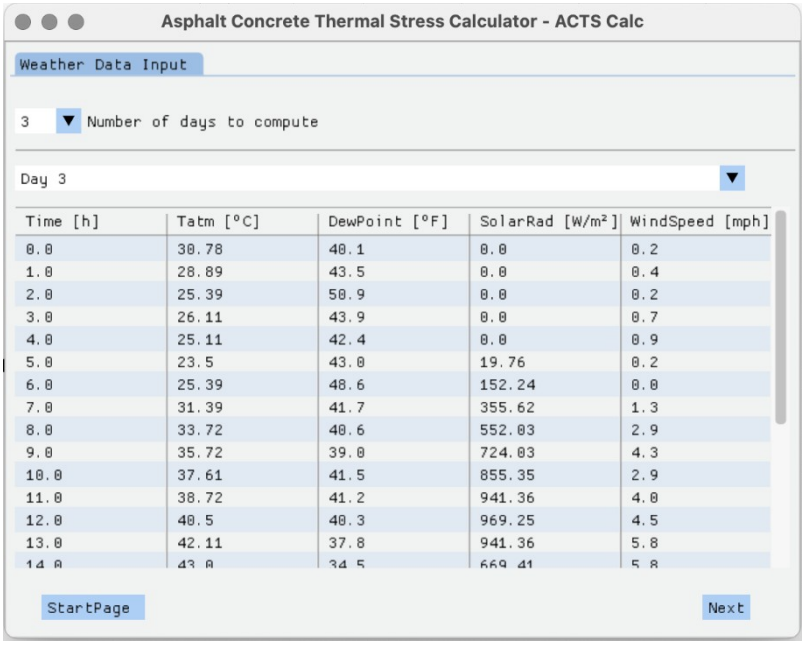


Figure 3. Uploaded meteorological inputs in the software

By clicking the "Next" button, the user validates the importation, and a verification process is realized onto the imported weather data to prepare it for future use in the thermal model. Initially, an approximation of the wind velocity data is done. A cubic polynomial fit is applied to the measured data to smooth it without losing relevant information. The results of the fitting process are displayed graphically (refer Figure 4), the acquired parameters as well as the R^2 error are shown for each of the 3 days.

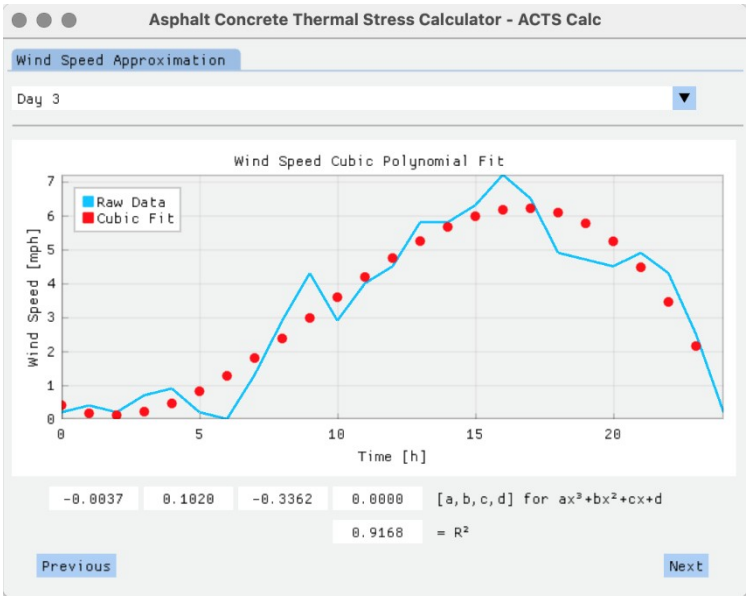


Figure 4. Graphical fitting process

Depending on the number of days to compute, a "stitching process" could be needed between the day-by-day data (refer Figure 5). This process is automatically done, and the results shown. In this time, the wind speed used is the one obtained after the cubic polynomial fit.

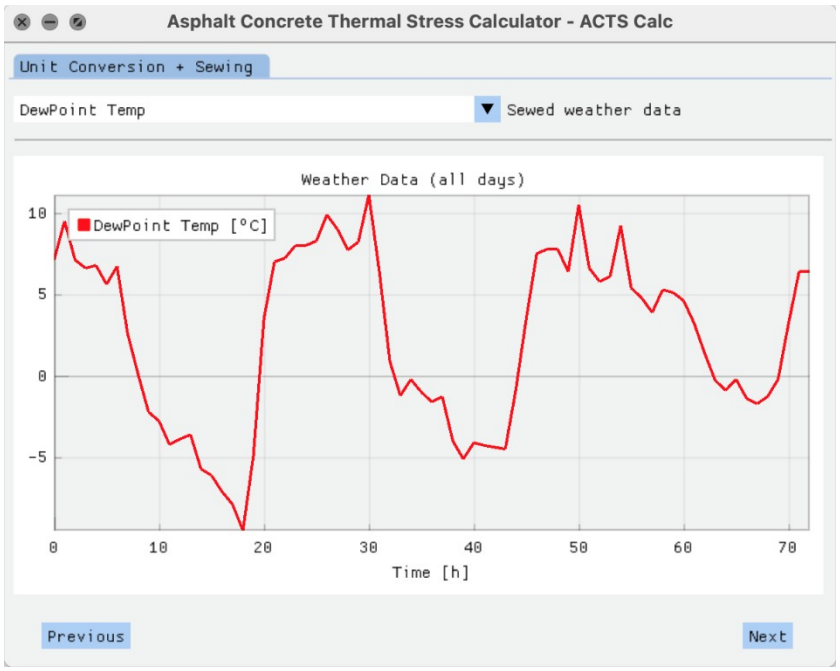


Figure 5. Sewed weather data

In the next step, the user needs to enter the calculation parameters to perform the thermal profile estimation (refer Figure 6). These are:

- The discrete time-step [s]
- The discrete spatial-step [m]
- The number of layers in the model [2-5]: There are at least two layers, the Hot Mix Asphalt (HMA) layer and the ground layer. However, the user can choose to include 3 more layers in-between those two (these could be old-HMA layers or other relevant soil information). The drop-down menu automatically updates the next windows, so the user can input material properties data for each of the layers
- The surface material and characteristic properties, including:
  - surface albedo (0-1)
  - surface emissivity (0-1)
  - sky view-factor (0-1)
  - solar view-factor (0-1)
  - characteristic length (for convection modelling) [m]
- The deep-ground properties, namely:
  - Deep-ground temperature [ºC]
  - Maximum ground depth [m]

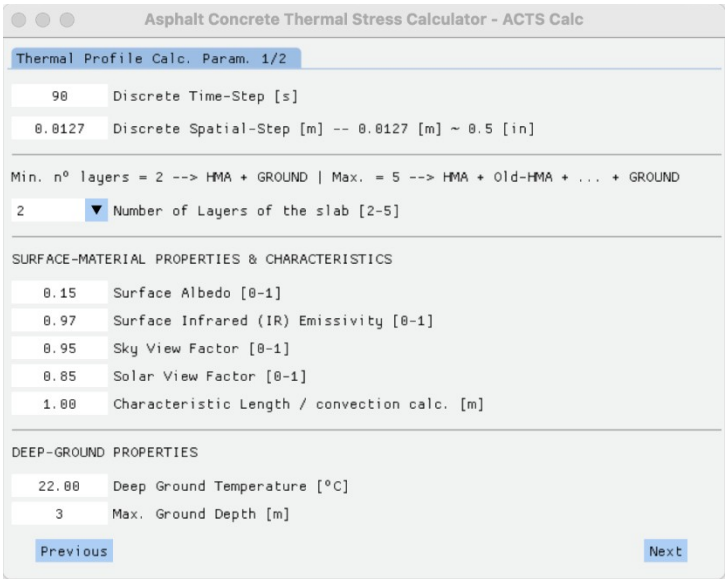


Figure 6. Thermal Calculation Parameters

As previously said, depending on the number of layers defined, a dynamic table appears to input the layer properties (refer Figure 7). The columns are organized left to right, from the surface layer to the subgrade. For all layers, the following properties are needed:

- Density [kg/m³]
- Specific heat capacity [J/kgºK]
- Thermal conductivity [W/mºK]

For non-ground layers, two more properties are need:

- The layer thickness [m]
- The thermal contact resistance between this layer and the adjacent one (0-1). The thermal contact resistance corresponds to the resistance between asphalt and subgrade.

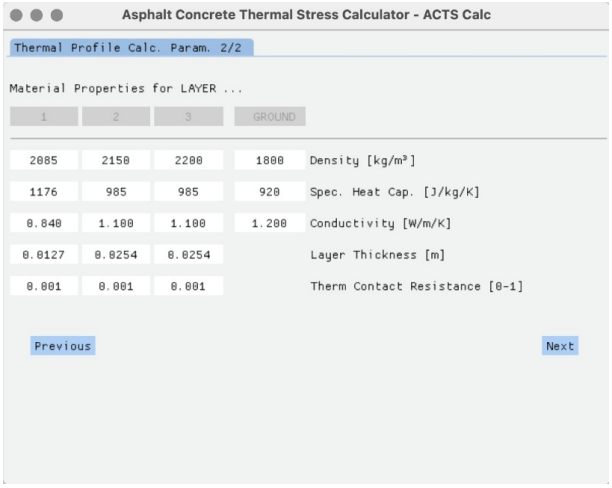


Figure 7. Layer Properties

b. Thermal Model - Calculation Process & Results

Once validated the model parameters and material properties, the calculation begins. The physical model is solved using a "*Forward-Time Centered-Space Finite Difference*" method. The system of equation used induces recurring terms, it means that some equations use temperature results calculated at the same time-step, which themselves depend on these equations. The thermal profile will converge to the definitive solution at each step. Thus, an iterative process is needed to solve the problem. Figure 8 shows how the calculation ongoing, and convergence is visualized numerically in the program, along with some "important event" notifications.

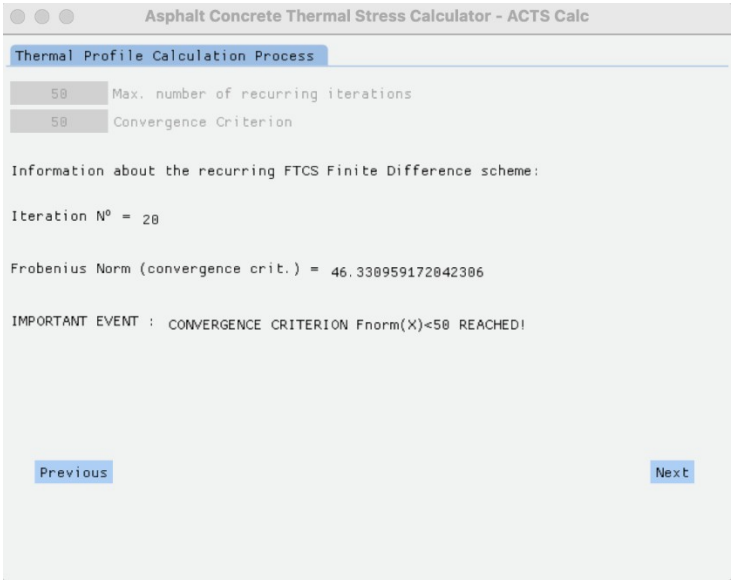


Figure 8. Visualization of the ongoing calculation and convergence

The mentioned "important events" can refer to the following:

- The occurrence of turbulent flow in the calculation process
- The end of the calculation without reaching the convergence criterion
- The convergence of the iterative algorithm (information on the calculation convergence is given in the "recurring_iterations_log.xlsx" file)
- The occurrence of a numerical instability and the cancellation of the calculation. In this case, the user is asked to change the value of the time- and/or space-step, to ensure the CFL-criterion validation, which will be explained later in this document

If the calculation is done without a problem ("convergence reached"), the user can visualize the obtained thermal profile with respect to time. Even if the calculation is realized on the whole profile, down to the maximum ground depth, only non-ground results are shown (refer Figure 9). This is realized to optimize the visualization and obtain faster graphical results. However, a "temp_profile.csv" file, which contains the full data, and a more workable file called

"temp_profile_subsample.xlsx", are available to the user in the directory "Results". The temperature values are expressed in [°C]. In this table, the rows are related to the time step [s] whereas the columns refer to the depth-step [m] (refer Figure 9).

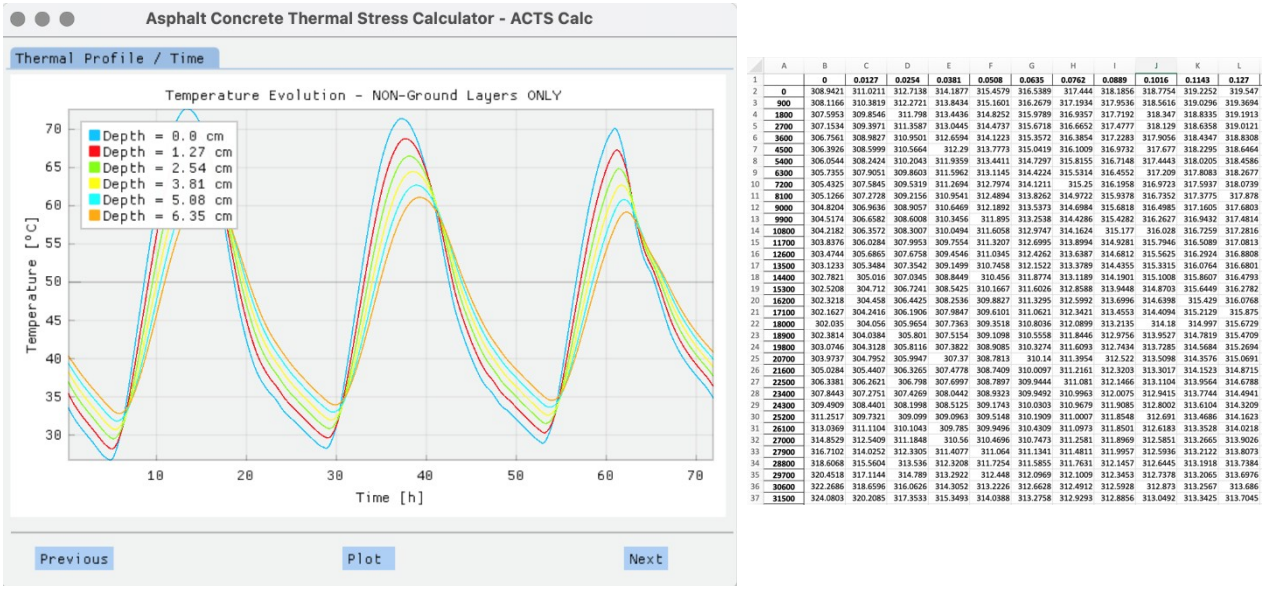


Figure 9. Thermal calculation results, plot, and Excel file.

The library used to plot the graphs is highly dynamical. It allows to change the position of the legend, toggle on/off some curves, etc., by entering the plot menu (right click > settings > legend). In this way, cleaner curves can be obtained, as shown in Figure 10.

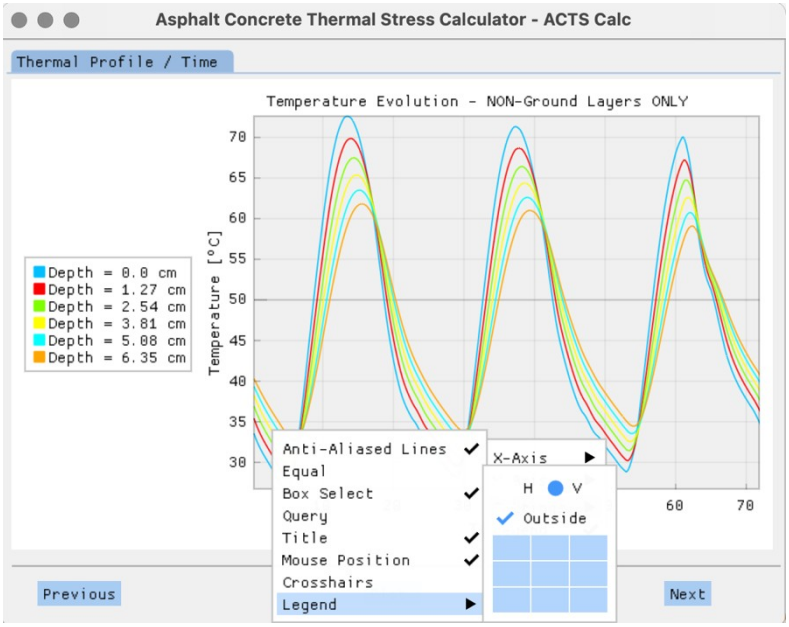


Figure 10. Optimized view of the thermal calculation results

Next, the user can select a given time-step and visualize thermal profile with respect to the pavement depth. As previously, the graph can be rework (rectangle-zoom, by clicking and sliding) to enhance the visualization quality. An export of the thermal profile to an Excel file is also possible (refer Figure 11).

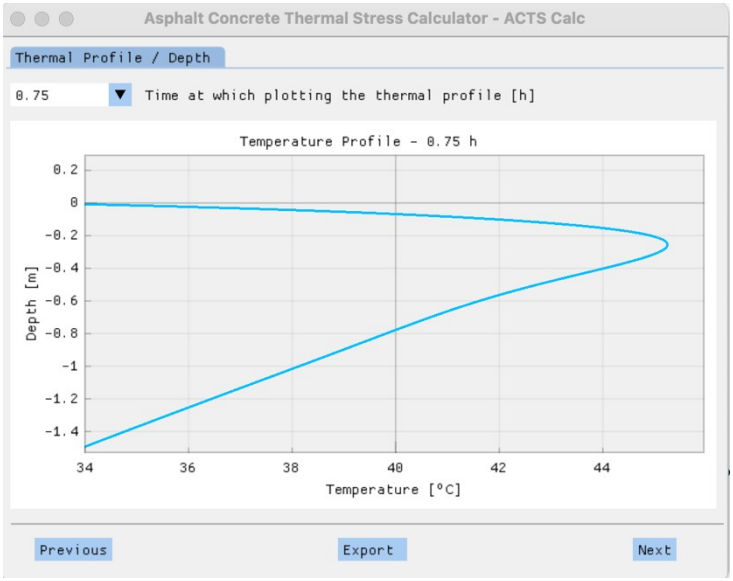


Figure 11. Thermal calculation results at a specific time

c. Thermal Model - Mathematical & Physical Description

- Outgoing radiation properties

To calculate the outgoing radiation from the surface to the horizon, a sky temperature needs to be estimated. This is realized through Equation 1.

$$T_{sky} = T_{atm}\left(0.004\,T_{dew} + 0.8\right)^{0.25} \qquad \text{Equation 1.}$$

Where:

$T_{sky}\ the\ sky\ temperature\ [^{\circ}K]$
$T_{atm}\ the\ atmospheric\ dry-bulb\ temperature\ [^{\circ}K]$
$T_{dew}\ the\ dew-point\ temperature\ [^{\circ}C]$

- Expressions used within the recurring scheme

The following equations show recurrent terms. These depend on the surface temperature, which is calculated using the results of the previous equations. These equations are integrated within the recurring scheme and solved iteratively until the defined convergence criterion is reached.

Outgoing radiation coefficient calculation:

$$h_{rad} = \Psi_{sky}\,\varepsilon\sigma\left(T_S^2 + T_{sky}^2\right)\left(T_S + T_{sky}\right) \qquad \text{Equation 2.}$$

Where:
- h_{rad} the abbreviated parameter for outgoing radiative heat transfer coefficient[]
- ψ_{sky} the sky view factor []
- ε the infrared emissivity of the surface []
- $\sigma = 5.67\,e^{-8}\left[W/m^2/^\circ K^4\right]$ the Stefan-Boltzmann constant
- $T_S\left[^\circ K\right]$ the pavement surface temperature

Convection heat transfer calculation:

The air film temperature depends on “surface” temperature.

$$T_{film} = \frac{T_{surface} + T_{atm}}{2} \qquad \text{Equation 3.}$$

T_{film} is then used together with the “Reference air film properties” to interpolated or extrapolate the “actual” air film properties:

- kinematic viscosity ν_{film} (T_{film}) [m²/s]
- conductivity κ_{film} (T_{film}) [W/m°K]
- diffusivity α_{film} (T_{film}) [m²/s]
- Prandtl number Pr (T_{film}) []

Reynolds number of air film:

Laminar or turbulent air flow?

$$R = \frac{U_{film} L}{\nu_{film}} \qquad \text{Equation 4.}$$

Where:
- Re the Reynolds number of air []
- U_{film} the wind velocity [m/s]
- L the characteristic length of the pavement [m]
- ν_{film} the kinematic viscosity of air [m²/s]

If Re $<5^5$, then the flow is laminar. Else it is turbulent.

Nusselt number of air film in function of laminar/turbulent flow:

For laminar flow:

$$Nu_{laminar} = 0.664\left[Pr^{1/3}\Re^{0.5}\right] \qquad \text{Equation 5.1}$$

For turbulent flow:

$$Nu_{turbulent} = 0.037\left[Pr^{1/3}\Re^{0.8}\right] \qquad \text{Equation 5.2}$$

Where:
- Nu_{xxx} the Nusselt number of air []
- Pr the Prandtl number of air []

Convective heat transfer coefficient of air (h_∞)

For laminar flow:

$$h_{laminar} = 0.664\left[k_{film} Pr^{1/3} \nu_{film}^{-0.5} L^{-0.5} U_{film}^{0.5}\right] \qquad \text{Equation 6.1}$$

$$h_{laminar} = Nu_{laminar}\frac{k_{film}}{L} \qquad \text{Equation 6.2}$$

For turbulent flow:

$$h_{turbulent} = 0.037\left[k_{film} Pr^{1/3} \nu_{film}^{-0.8} L^{-0.2} U_{film}^{0.8}\right] \qquad \text{Equation 6.1b}$$

$$h_{turbulent} = Nu_{turbulent}\frac{k_{film}}{L} \qquad \text{Equation 6.2b}$$

Where:
- $h_{xxx} = h_\infty$ the convective heat transfer coefficient of air []
- k_{film} the thermal conductivity of air [W/m°K]

d. Stability verification: Courant-Friedrichs-Lewy (CFL) criterion

Calculate Courant-Friedrichs-Lewy - CFL criterion at all time

$$\Delta t \leq \frac{\rho_S c_S \Delta x^2}{2\left(h_{rad}\Delta x + h_\infty \Delta x + k_S\right)} \qquad \text{Equation 7.}$$

With:
- ρ_S the density of the surface layer [kg/m³]
- c_S the specific heat capacity of the surface layer [J/kg°K]
- k_S the thermal conductivity of the surface layer [W/m°K]
- Δt the temporal discretization step [s]
- Δx the spatial discretization step [m]

e. Heat equations development using Finite Differences (FD)

A Forward-Time Central-Space (FTCS) Finite Difference scheme, also known as "explicit scheme", is used to solve the general 1-D diffusion equation:

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} \qquad \text{Equation 8.1}$$

Where α is the diffusivity [m²/s].

This translates into the following equation:

$$\frac{u_m^{t+1} - u_m^t}{\Delta t} = \frac{u_{m-1}^t - 2u_m^t + u_{m+1}^t}{\Delta x^2} \qquad \text{Equation 8.2}$$

Where:
- m represents the spatial position of the value of interest u
- t represents the temporal position of the value of interest u
- u_m^{t+1} is the sole unknown of the equation

The pattern associated to the FD-FTCS scheme is shown in the Figure 12:

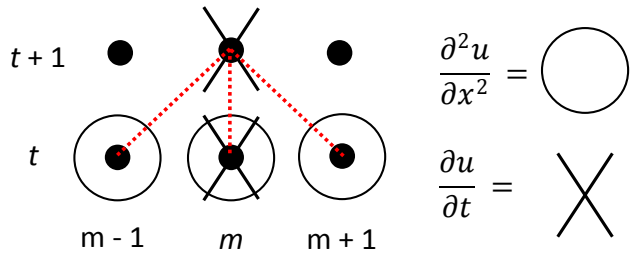


Figure 12. The associated pattern to the FD-FTCS scheme

To consider the different phenomena involved in the heat exchange between the pavement and its surroundings (i.e. environment, old asphalt layers, ground), the general 1-D diffusion equation is solved in a different way in function of the calculation depth of interest. This can be seen in the Figure 13 (Gui, et al., 2007).

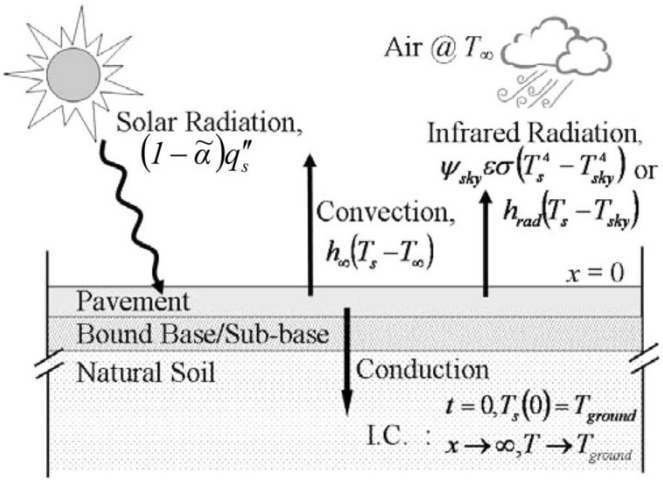


Figure 13. Heat exchange between the pavement and its surroundings

In this way, at the surface (depth = 0m), the Equation 8 is modified to consider the input solar radiation, the output infrared radiation of the pavement, the thermal convection as well as the thermal conduction. The resulting "Equation A" can be seen below.

On the other hand, for interior-nodes (depth > 0m), only thermal conduction must be considered. Thus, a simpler "Equation B" is developed and used.

A special case of this equation can be encountered when there is a thermal contact resistance R_{ij} between two different layers. Such specificity is considered thanks to "Equation C" that allows considering the phenomenon while ensuring heat flux continuity between the *i* and *j* layers:

$$k_i \frac{\mathrm{d}T_i}{\mathrm{d}x} = k_j \frac{\mathrm{d}T_j}{\mathrm{d}x} \qquad \text{Equation 9.1}$$

and

$$R_{ij} = \frac{T_{i,interface} - T_{j,interface}}{q\} \text{ rsub } \{\text{interface}\} ¿¿} \qquad \text{Equation 9.2}$$

Where:
- $T_{i,\text{interface}}$ and $T_{j,\text{interface}}$ the interface temperatures at the i^{th} and j^{th} layers, respectively
- $q\}$ rsub {interface ¿ the heat flux flowing through the interface

As the true initial thermal profile of the pavement is unknown, the "Initial Boundary Condition" (BC) $T(t=0, all\ \ depth)=T_{ground}$ is used. Additionally, the following BC is considered: $T(all\ \ time, \infty)=T_{ground}$.

Finally, it is worth noting that, due to the lack of initial thermal profile, an iterative recurring scheme is needed to compute the solution. This recurrence is realized applying $T(t=0, all\ \ depth)=T(t_{final}, all\ \ depth)$ at each iteration (except the first one, where the initial BC applies).

A matrixial "Frobenius" norm of the difference between thermal profile of iteration $i-1$ and iteration i is used in order to evaluate the convergence of the recurring algorithm and define a suitable stopping criterion.

$$X=[\text{ThProfile}]^{i-1}-[\text{ThProfile}]^{i} \qquad \text{Equation 10.1}$$

$$||X||_F=\sqrt{\sum_i \sum_j |x_{ij}|^2} \qquad \text{Equation 10.2}$$

*** **Equation (A):** Heat equation for surface-node S

$$T_s^{t+1}=\frac{2\Delta_t}{\rho_s c_s \Delta_x}¿$$

With:
- T_S^{t+1} the surface temperature at time $t+1$[°K]
- Ψ_{solar} the solar view factor []
- $\overline{\alpha}$ the surface albedo []
- q } rsub {s} rsup {t+1¿ the input solar radiation flux [W/m²]
- T_1^t the temperature at the first discretized depth at time t [°K]
- T_S^t the surface temperature at time t [°K]

*** **Equation (B):** Interior-node at depth m

$$T_m^{t+1}=\frac{k_i \Delta t}{\rho_i c_i \Delta x^2}\left[T_{m-1}^t-2T_m^t+T_{m+1}^t\right]+T_m^t$$

With:
- T_m^{t+1} the temperature at depth m and time $t+1$ [°K]

- ρ_i the density of layer i [kg/m³]
- c_i the specific heat capacity of layer i [J/kg°K]
- k_i the thermal conductivity of layer i [W/m°K]
- T_{m-1}^{t} the temperature at depth $m-1$ and time t [°K]
- T_{m}^{t} the temperature at depth m and time t [°K]
- T_{m+1}^{t} the temperature at depth $m+1$ and time t [°K]

*** **Equation (C):** Interface-node at depth $m=n$

$$T_n^{t+1}=\frac{1}{2}\left(T_{i,\text{interface}}+T_{j,\text{interface}}\right)$$

$$T_n^{t+1}=\frac{1}{2}\left[\frac{2\Delta x k_i+k_i k_j R_{ij}}{k_i k_j R_{ij}+\Delta x k_i+\Delta x k_j}\left(T_{n-1}^{t}\right)+\frac{2\Delta x k_j+k_i k_j R_{ij}}{k_i k_j R_{ij}+\Delta x k_i+\Delta x k_j}\left(T_{n+1}^{t}\right)\right]$$

With:
- T_n^{t+1} the temperature at depth n and time $t+1$ [°K]
- $k_i \wedge k_j$ the thermal conductivity of layer $i \wedge j$, respectively [W/m°K]
- $R_{ij}$the thermal contact resistance between layers $i \wedge j$ (0-1)
- T_{n-1}^{t} the temperature at depth $n-1$ and time t [°K]
- T_{n+1}^{t} the temperature at depth $n+1$ and time t [°K]

- Coefficient calculations for non-surface depths:

For each i^{th} layer, some parameters are defined. These are directly related to the heat equations that govern the heat energy in- and out-take, as well as their diffusion within the pavement. They are computed in order to simplify the understanding and lecture of the developed code in python.

Some parameters can be computed outside of the iterated recurring scheme as they are fixed. This is the case for the parameters $\delta_i, A_i \wedge B_i$,associated with the interior-node heat Equation B.

$$T_m^{t+1}=A_i T_m^{t}+B_i\left(T_{m-1}^{t}+T_{m+1}^{t}\right) \qquad \text{Equation 11.}$$

Where:
- $\delta_i=\frac{2\Delta t}{\rho_i c_i \Delta x}$
- $A_i=1-\left[\delta_i \frac{k_i}{\Delta x}\right]$
- $B_i=\frac{\delta_i k_i}{2\Delta x}$

Identically, parameters $C_{ij} \wedge D_{ij}$, related to the interface-node Equation C applied on the i^{th} and j^{th} layers, are computed in advance.

$$T_n^{t+1} = C_{ij}\, T_{n-1}^t + D_{ij} T_{n+1}^t \qquad \text{Equation 12.}$$

Where:

- $C_{ij} = \frac{1}{2}\left[\frac{2\Delta x k_i + k_i k_j R_{ij}}{k_i k_j R_{ij} + \Delta x k_i + \Delta x k_j}\right]$
- $D_{ij} = \frac{1}{2}\left[\frac{2\Delta x k_j + k_i k_j R_{ij}}{k_i k_j R_{ij} + \Delta x k_i + \Delta x k_j}\right]$

- Coefficient calculations for surface

In the case of the surface-node, in Equation A, only 2 of the 4 parameters can be calculated outside the iterative loop: $\delta_S \wedge B_S$.

$$T_S^{t+1} = A_S^{t+1} T_S^t + B_S T_1^t + C_S^{t+1} \qquad \text{Equation 13.}$$

Where:

- $\delta_S = \frac{2\Delta t}{\rho_S c_S \Delta x}$
- $A_S^{t+1} = 1 - \delta_S\left[h_\infty^{t+1} + h_{rad}^{t+1} + \frac{k_S}{\Delta x}\right]$
- $B_S = \delta_S \frac{k_S}{\Delta x}$
- $C_s^{t+1} = \delta_s$ ¿

However, parameters $A_S^{t+1} \wedge C_S^{t+1}$ need to be included into the recurring scheme as they depends on the convective $(h_\infty) \wedge outgoin\, gradiative\, (h_{rad})$ coefficients, which themselves depend on the surface temperature T_S^t.

II. Stresses Model

The stress calculation is realized only on the 1st pavement layer. Thus, only this part of the thermal profile previously calculated is considered in the following. This approach was chosen since the thermal cracking phenomenon mostly affect the top part of the pavement structure.

The stress profile within the surface layer is enough to estimate the pavement's cracking potential. Reducing the calculation to this sole portion of the pavement allows to decrease the computational burden of the calculation. Focusing the calculation of stresses in the first layer avoids the stress-discontinuity at the layers' interface due to the different thermal contraction coefficients.

It should be noted that the time-step use in the following is 10x higher than the one employed in the thermal calculation. Indeed, if a small time-step was needed in order to ensure the stability of the "explicit scheme", it is not the case anymore. The use of a bigger time-step allows a faster software responsiveness while maintaining a good accuracy.

a. Stresses Model Inputs

- Coefficient of Thermal Contraction

In this part of the software, the user can choose between 3 solutions (refer Figure 14).

1. The user knows the linear Coefficient of Thermal Contraction of the asphalt concrete mixture and directly enters it: CTC_{mix} $[\frac{mm^{-5}}{^{\circ}C}]$

2. The user only partially knows the mixture properties. CTC_{mix} thus, needs to be computed using the following user-inputs:

- CTC_{binder} = Linear coefficient of thermal contraction of the binder $[\frac{mm^{-5}}{^{\circ}C}]$
- CTC_{agg} = Linear coefficient of thermal contraction of the aggregates $[\frac{mm^{-5}}{^{\circ}C}]$
- VMA = Voids in Mineral Aggregate [%]
- V_{agg} = Aggregate volume in mixture [%]

CTC_{mix} calculation is performed using Equation 14.1 (Lytton, et al., 1993).

$$CTC_{mix} = \frac{VMA \times CTC_{binder} + V_{agg} \times CTC_{agg}}{3 \times V_{total}} \qquad \text{Equation 14.1}$$

Where:

$3 \times V_{total} = 3 \times 100[\%] = 300$

3. The user does not know any of the mixture properties. He then must enter the following information:

- CTC_{binder} = Linear coefficient of thermal contraction of the binder $[\frac{mm^{-5}}{^{\circ}C}]$
- CTC_{agg} = Linear coefficient of thermal contraction of the aggregates $[\frac{mm^{-5}}{^{\circ}C}]$
- *Gsb* = Aggregate bulk specific gravity []
- *Gb* = Binder specific gravity []
- *Gmm* = Asphalt mixture maximum specific gravity []
- *AV*= Air volume [%]
- *BC* = Binder content [%]

The software automatically calculates $VMA \wedge V_{agg}$, using Equations 14.4 and 14.5 respectively (NCAT, 2009) (NCAT, 2012). The results are then used in Equation 14.1 to recover CTC_{mix}.

$$Gmb = \left(1 - \frac{AV}{100}\right) Gmm \qquad \text{Equation 14.2}$$

$$Gse = \frac{100 - BC}{\frac{100}{Gmm} - \frac{BC}{Gb}} \qquad \text{Equation 14.3}$$

$$VMA = 100 - Gmb \cdot \frac{100 - BC}{Gsb} \qquad \text{Equation 14.4}$$

$$V_{agg} = Gmb \cdot \left(\frac{100 - BC}{Gse}\right) \qquad \text{Equation 14.5}$$

Where:
- *Gmb* = Asphalt mixture bulk specific gravity []
- *Gse* = Aggregate effective specific gravity []

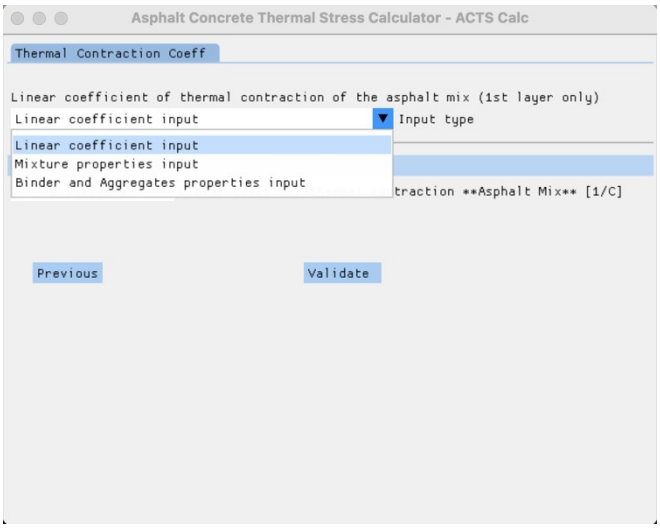


Figure 14. CTC Input alternatives

Once the user "validates" (see button in figures above) the chosen method, the calculation(s) is (are) realized, and the results for CTC_{mix} actualized and shown in the shaded-out case "Coefficient used for calculation".

b. Mechanical Properties

The last step before performing the stress calculation consists in entering the mechanical properties of the material, and numerically manipulating them in order to obtain the Relaxation Modulus Master Curve (ErelMC).

To do so, the user has the choice between entering the “Creep Compliance” (CC) data obtained experimentally at different temperatures, or to inform the “Relaxation Modulus” (Erel) data for various testing temps. The data is imported from Excel tables that can be found in the “DATA” directory under the name of "creep_compliance.xlsx" and "relax_modulus.xlsx", respectively. Both tables are made up of 3 columns:

1. Testing temperature [ºC];
2. Testing time [s] or frequency [Hz];
3. Mechanical property value [1/GPa] or [GPa] respectively.

Through a drop-down menu, the user can choose to read one or another of the Excel files. Depending on the chosen option, 2 different scenarios emerge.

Creep Compliance Properties Entry Alternative:

- Material Input

If the user chose to enter CC measurements, the software will show the related data on the screen. This way, the user can verify that the importation process took place in a good manner (refer Figure 15).

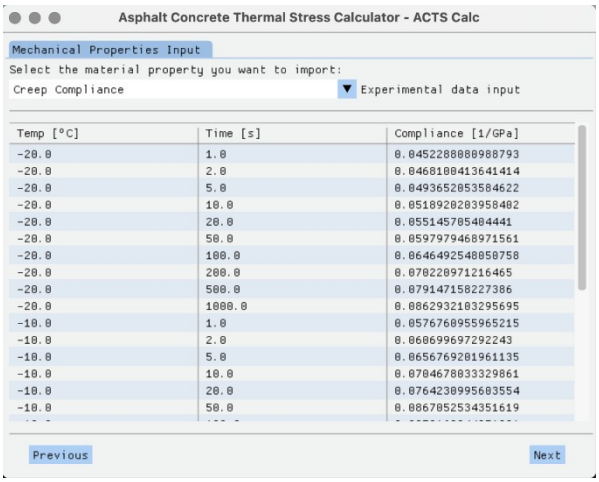
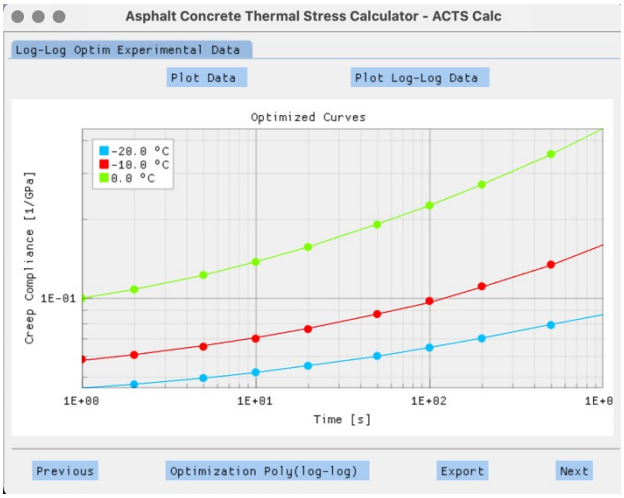


Figure 15. Uploaded Creep Compliance data from the lab

Once the user validates the data, the software will plot the raw data, and the user can choose to observe the data on a log-log scale.

Figure 16 shows a second order polynomial function, which has been fitted to each log-log creep compliance curve (resulting in a polynomial for each testing temperature). The parameters of those functions can be exported in a text file clicking on “Export” and found in the “Result” directory under the name "polymial_optim_material.txt".




```
Polynomial fit "a.x²+b.x+c" for temperature = -20.0 ºC
Results :
a = 0.016762007692050378 +/- 0.0005908969293155514
b = 0.04407328468372216 +/- 0.0018424940962699133
c = -1.344916809247515 +/- 0.001189557991796078
R² = 0.9998060648489542

Polynomial fit "a.x²+b.x+c" for temperature = -10.0 ºC
Results :
a = 0.03243472811496347 +/- 0.0017247587169646238
b = 0.046507823900791764 +/- 0.0053780234903025815
c = -1.2346139293533427 +/- 0.0034721794664184466
R² = 0.9992986054868952

Polynomial fit "a.x²+b.x+c" for temperature = 0.0 ºC
Results :
a = 0.03841515267268902 +/- 0.0005233612311514409
b = 0.10020482701830132 +/- 0.0016319088302037229
c = -1.001511981280431 +/- 0.0010535989534306409
R² = 0.9999708182973699
```

Figure 16. Creep Compliance Plot and Fitted Polynomial Parameters

- CCMC Construction

With this information, the Creep Compliance Master Curve (CCMC) can then be built. The Master Curve is obtained thanks to the application of the time-temperature superposition principle (TTSP) that stands for thermo-rheologically simple materials. In short, this principle states that the behavior of a material at low temperature is equivalent to its response to short time (or high frequency) excitations. Conversely, the material behavior at high temperature is assimilated to its response to long time (low frequency) excitations (Roylance, 2001). Thus, it is possible to calculate a shift-factor a_T for each testing temperature, to estimate a "reduced time" $\xi = \frac{t}{a_T}$ corresponding to their equivalent long- or short-time excitations at a constant temperature (Olsen, et al., 2001).

Here, the construction of the Master Curve is realized following a technique developed by Witczak, that allows an automatic horizontal shifting of the non-reference curves (Witczak & Bari, 2004). It consists in calculating discrete shift-factor $\log(a_T)$ using the polynomial functions previously optimized. This process is closed to another method, called "equivalent slope", that shows significantly better results that traditional shifting techniques (Saboo & Kumar, 2018).

To obtain the CCMC of its material, the user just needs to choose a reference temperature. Depending on this choice, results might be accurate. Thus, the user is invited to test different reference temperature in order to find the most adequate (refer Figure 17).

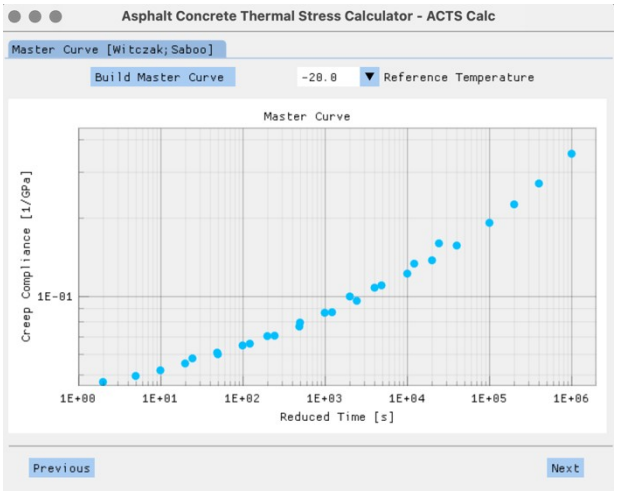


Figure 17. CCMC with a specific reference temperature

The discrete shift-factors $\log(a_T)$ found during the construction of the master curve are accessible to the user. Those can be found under the "Results/discrete-shift.xlsx" file.

- Continuous Shift-Factor

Once the CCMC and the associated discrete shift-factors are obtained, the user can choose between the *William-Landel-Ferry (WLF)* and the *Arrhenius law* to obtain a continuous evolution of the shift-factor with the temperature. The user is invited to take a close look to the obtained data, since the WLF law was initially developed in a semi-empirical fashion for plastic materials (amorphous polymers), considering α-transitions as the main relaxation mechanism. Those transitions occur when the material is slightly above the glass transition temperature ~ $\left[T_g ; T_g + 100 ºC\right]$. On the contrary, the Arrhenius law considers β-transformations, that are more common when the study temperature is well below T_g (Roylance, 2001).

$$\log\left(a_T\right)\left(T, T_{ref}\right) = \frac{-C_1\left(T - T_{ref}\right)}{C_2 + \left(T - T_{ref}\right)} \qquad \text{Equation 15.}$$

$$\log\left(a_T\right)\left(T, T_{ref}\right) = \frac{E_a}{\ln(10) R}\left(\frac{1}{T} - \frac{1}{T_{ref}}\right) \qquad \text{Equation 16}$$

Where:

- T the working temperature [ºC]
- T_{ref} the reference temperature [ºC]
- a_T the shift-factor at temperature T with reference temperature T_{ref}
- C_1 and C_2 the William-Landel-Ferry parameters
- E_a the material activation energy [J/mol]
- R the ideal gas constant with $R \approx 8.314$ [J/(K.mol)].

Depending on the material and working temperatures, as well as on the optimization process itself using the bounded “Trust Region Reflective” algorithm from the scientific computing tools for Python (SciPy, 2021), the parameters obtained when using the WLF law through Equation 15 are not always optimal and can eventually lead to erroneous results. On the contrary, the sole parameters of the Arrhenius law using Equation 16, generally gives satisfying results, and stands on stronger scientific grounds, since it relies on the material chemical and physical properties.

Results obtained for the same set of discrete shift-factors with both methods are shown in Figure 18.

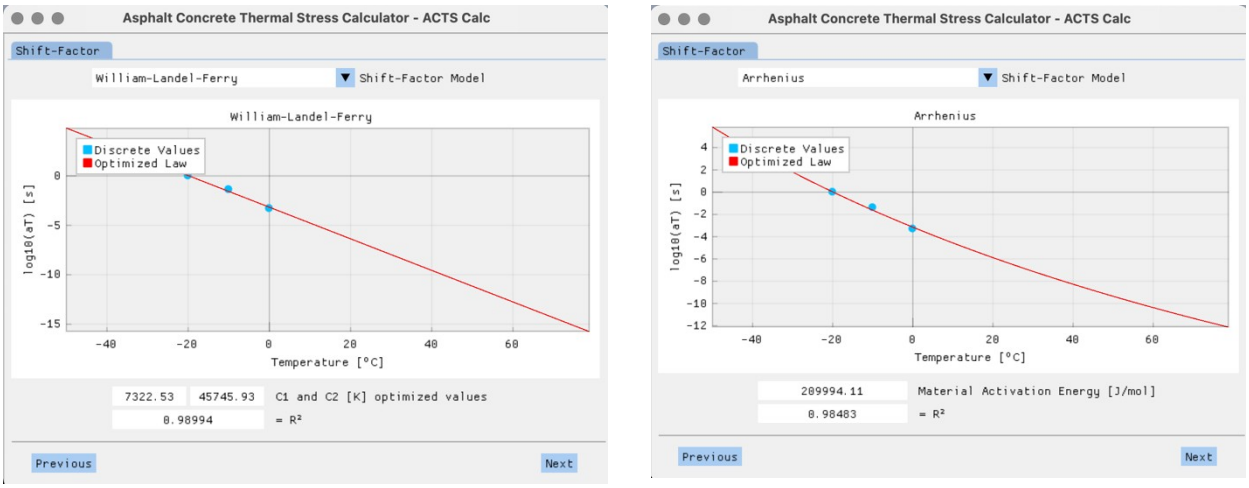


Figure 18. WLF Optimized Law and Arrhenius Optimized Law respectively

Optimized parameters for one or another of the models are accessible to the user in the “Results” directory, in the Excel files named “optim-wlf.xlsx" or "optim-arrhenius.xlsx". They are also directly visible on the software screen, along with the R^2resulting from the optimization process.

- G-MPower Serie Optimization on CCMC

Once the shift-factor model is chosen and its parameters optimized on the discrete data, a Generalized Modified Power Serie (G-MPower) is used to obtain a continuous evolution of the CCMC in function of the reduced time ¿). This function is particularly suited to obtain a smooth and accurate evolution of the CCMC (Park & Kim, 2001).

$$D(\xi) = D_0 + \sum_{i=1}^{N} \frac{D_i}{\left(1 + \frac{\tau_i}{\xi}\right)^k}$$ Equation 17.

Where:

- $D(\xi)$ the creep compliance at reduced time (ξ ¿ [1/GPa]
- D_0 the glassy creep compliance [1/GPa] - at $\xi \rightarrow 0^{+¿¿}$
- D_i the local analogous creep compliance values [1/GPa]
- τ_i the local analogous relaxation time [s]
- k the power-law factor.

The user can choose the number of "branches" of the Power Serie, and to directly visualize the results of the optimization for the selection (refer Figure 19).

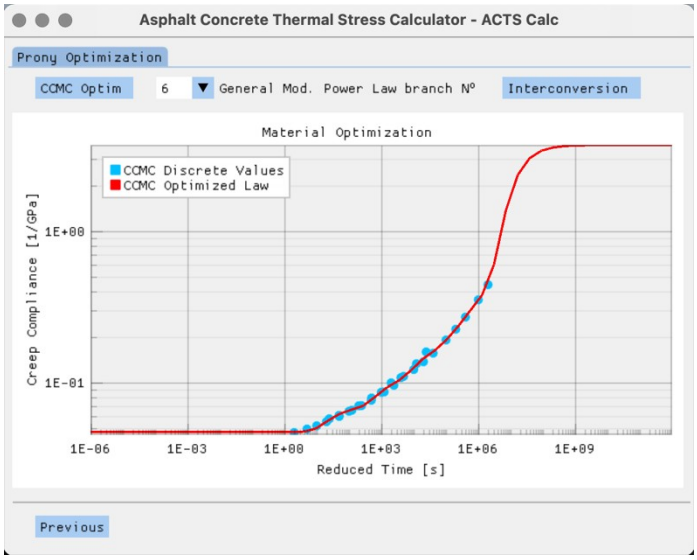


Figure 19. CCMC - G-MPower Optimization

The optimized parameters can, once again, be found in the "*Results/optim-GenModPower_CCMC.xlsx" file (refer Figure 220).

	A	B	C	D	E
1		**0**			
2	**D_0**	0.047352			
3	**D_1**	1697.634			
4	**tau_1**	0.068528			
5	**D_2**	7.681806			
6	**tau_2**	3.207072			
7	**D_3**	5378.967			
8	**tau_3**	0.040187			
9	**D_4**	0.477004			
10	**tau_4**	0.022448			
11	**D_5**	0.012505			
12	**tau_5**	0.194324			
13	**D_6**	596.8267			
14	**tau_6**	0.122907			
15	**k**	107.1704			

Figure 20. CCMC - G-M Power Optimization Coefficients

- CCMC Interconversion to ErelMC

In order to calculate the thermal stress resulting from the temperature profile previously defined, the known as Relaxation Modulus Master Curve is needed. In this case, it can be obtained through a process called “interconversion”, since the relaxation modulus *Erel* is the conjugate of the creep compliance (Alavi, 2014) (Alavi, et al., 2017).

The CCMC being modelled with a Power Serie, it is not possible to use an analytical development of the interconversion process, as shown in (Park & Kim, 1999). Thus, a general "discretized" approach of the interconversion is implemented on the simulated points obtained in the previous step, following Equations 18 to 20.

$$n = ¿\frac{d\log F(\tau)}{d\log \tau} \vee ¿ \qquad \text{Equation 18.}$$

$$\alpha = \left(\frac{\sin n\pi}{n\pi}\right)^{1/n} \qquad \text{Equation 19.}$$

$$Erel(\xi) = \frac{1}{D\left(\frac{\xi}{\alpha}\right)} \qquad \text{Equation 20.}$$

Where:

- n the local log-log slope of the source function at $\tau = t$
- $F(\tau)$ the source function (note that n is now a function of t)
- α the expression for substitution
- D *the* Creep Compliance
- ξ the Reduce time

Figure 21 shows the results for the ErelMC.

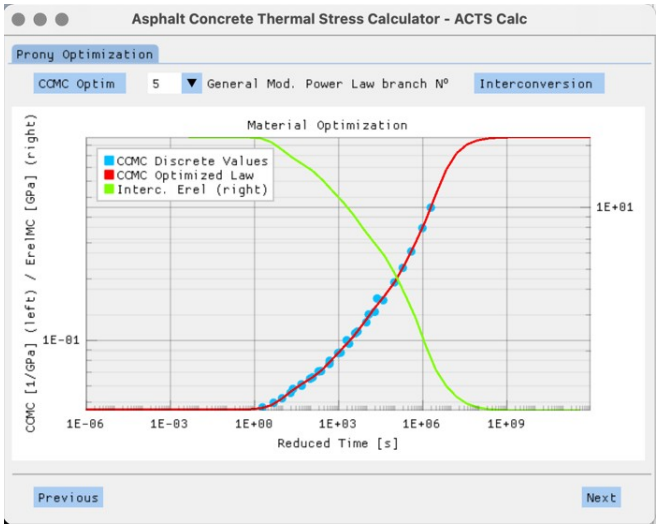


Figure 21. CCMC Interconversion to ErelMC

- Prony Serie Optimization on ErelMC

Once the interconverted ErelMC is obtained, a new Prony Serie is optimized onto it. This time, the Prony function is adapted to a Generalized Maxwell Model (GMM), also known as Wiechert Model. The associated mechanistic scheme is shown in Figure 22 (Adamczak & Bochnia, 2016).

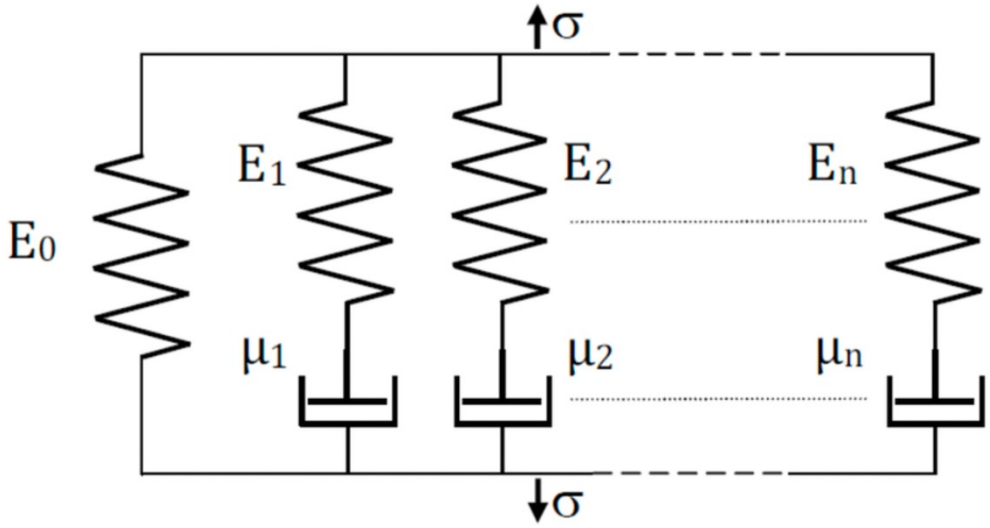


Figure 22. Generalized Maxwell Model for Relaxation Modulus

The associated equations are presented below:

$$Erel(\xi) = E_0 + \sum_{i=1}^{N} E_i\left(e^{-\xi/10^{\rho_i}}\right) \qquad \text{Equation 21.}$$

$$Erel(\xi) = E_0 + \sum_{i=1}^{N} E_i + \sum_{i=1}^{N} E_i\left(e^{-\xi/10^{\rho_i}} - 1\right) \qquad \text{Equation 22.}$$

$$Erel(\xi) = E_{glassy} - \sum_{i=1}^{N} E_i\left(1 - e^{-\xi/10^{\rho_i}}\right) \qquad \text{Equation 23.}$$

Where:

- $Erel(\xi)$ the relaxation modulus at reduced time ξ [GPa];
- E_0 the equilibrium relaxation modulus [GPa] - at $\xi \to +\infty$
- E_i local relaxation modulus values [GPa]
- $\rho_i = \log(\tau_i)$ the log of the relaxation time [s]
- E_{glassy} the glassy relaxation modulus [GPa] - at $\xi \to 0^{+¿¿}$

Once more, the user can choose the number of "branches" of the GMM, and to directly visualize the results of the optimization for the selection (refer Figure 23).

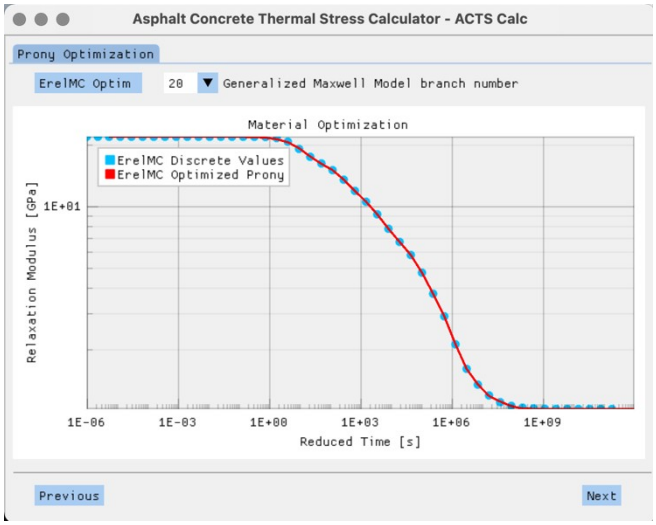


Figure 23. ErelMC - Prony Optimization

The optimized parameters can be found in the "Results/optim-prony_ErelMC.xlsx" file (refer Figure 24).

	A	B	C	D	E
1		**0**			
2	**E_0**	21.887714			
3	**E_1**	0.7369317			
4	**log(tau_1)**	2.3448441			
5	**E_2**	0.4015705			
6	**log(tau_2)**	1.1384852			
7	**E_3**	0.6416397			
8	**log(tau_3)**	1.1384857			
9	**E_4**	1.3813758			
10	**log(tau_4)**	2.3448428			
11	**E_5**	0.6807771			
12	**log(tau_5)**	6.6945394			
13	**E_6**	0.7917301			
14	**log(tau_6)**	2.3448436			
15	**E_7**	0.3990238			
16	**log(tau_7)**	1.1384857			

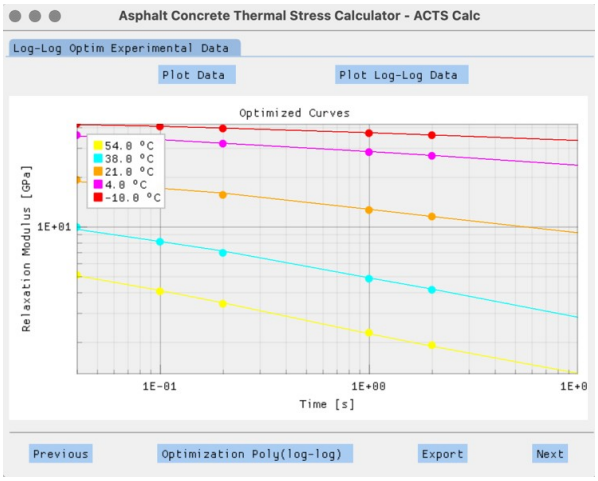
Figure 24. ErelMC - Prony Optimization Coefficients

Despite the higher number of steps and the complexity behind the "presmoothing" process, the obtained results are of greater value compared to a dual application of Prony Series.

Relaxation Modulus Properties Entry Alternative:

As explained earlier, the second scenario consists in directly entering the relaxation modulus data for different testing temperatures. After selecting the input type, the user can verify that the importation process worked as expected.

In this case, there is no need to optimize a Prony Serie on a CCMC, or to realize an interconversion process, since we can directly obtain the ErelMC. To do so, a process like the one presented above is employed (Witczak & Bari, 2004). However, the polynomial functions used in this case to fit the experimental data are of first order. This was decided due to the linear behavior of the log-log plot. The parameters of the linear functions can be exported in a text file and found in the “Result” directory under the name "polymial_optim_material.txt" (refer Figure 25).



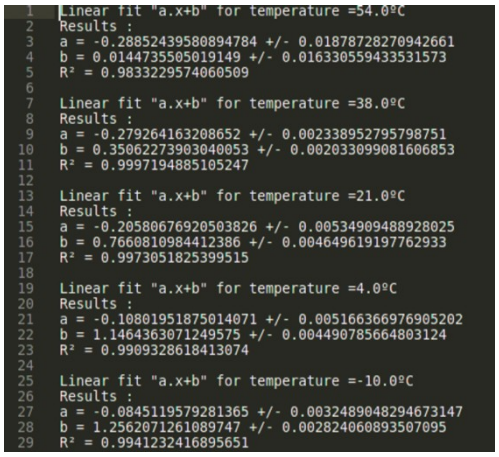


Figure 25. Relaxation Modulus Optimization Plot and Fitted Polynomial Parameters

After this step, the Erel Master Curve is constructed, and a shift-factor model (WLF or Arrhenius) selected.

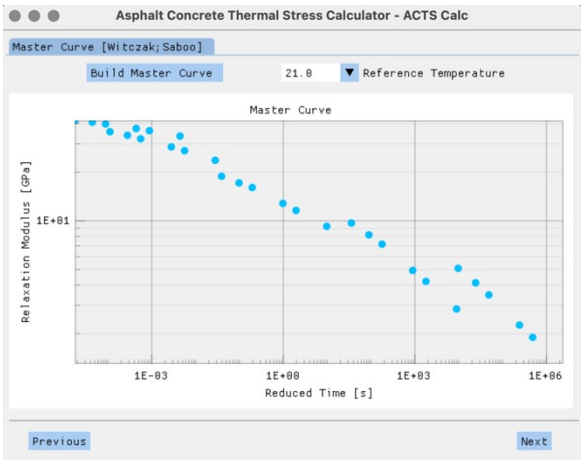


Figure 26. ErelMC with a Specific Reference Temperature

Finally, a Prony Serie is optimized onto a Generalized Maxwell Model in order to obtain a continuous evolution of the Relaxation Modulus in function of the reduced time. However, an additional smoothing process is realized using a sigmoid function according Equation 24. The simulated results obtained with the optimized sigmoid curve are the one used to obtain the Prony Coefficients.

$$\log ¿\, Erel(\xi) \vee ¿\, \delta + \frac{\alpha}{1+e^{\beta+\gamma \log \xi}} \qquad \text{Equation 24.}$$

Where:

- $Erel(\xi)$ the relaxation modulus at reduced time ξ [GPa]
- $\alpha, \beta, \gamma \wedge \delta$ are the parameters of the sigmoid law here employed

It should be noted that a higher number of Prony branches are needed in this case to obtain a smooth behavior at high reduced times. This leads to a higher computational resource and time demand. The optimized parameters are accessible to the user in the "Results" directory as "Results/optim-prony_ErelMC.xlsx" Excel file. Figure 27 shows the result of the Prony optimization.

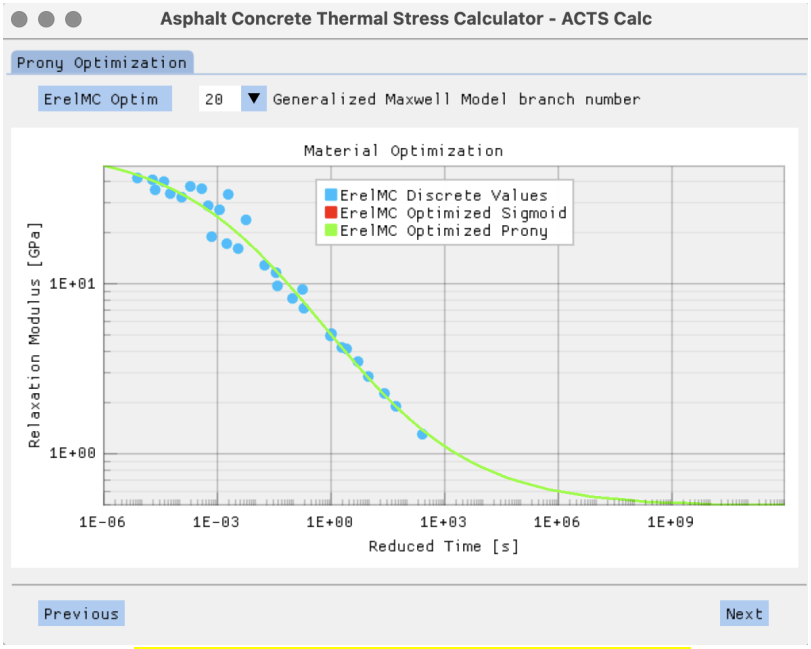


Figure 27. Result of the Prony optimization

c. Thermal Stress Calculation

Various techniques allow solving the ordinary differential equation (ODE) that arise from the Generalized Maxwell Model. In the next paragraphs, we will describe two of them, since both have been implemented and tested in the present software.

Finite Difference Method (FDM):

The stress in the $i-th\,arm\,of\,the\,GMM\,model$, σ_i, is given by Equation 25:

$$\frac{\mathrm{d}\sigma_i}{\mathrm{d}t}+\frac{1}{\tau_i}\sigma_i=E_i\frac{\mathrm{d}\varepsilon}{\mathrm{d}t} \qquad \text{Equation 25.}$$

Where ε is both the global and arm deformation, since $\varepsilon_{total}=\varepsilon_i=\varepsilon$ for the Wiechert Model.

Equation 25 can be re-written in backward finite difference form (implicit scheme) as:

$$\frac{\sigma_i^t-\sigma_i^{t-1}}{\Delta t}+\frac{1}{\tau_i}\sigma_i^t=E_i\frac{\varepsilon^t-\varepsilon^{t-1}}{\Delta t} \qquad \text{Equation 26.}$$

Where the superscripts $t\wedge t-1$ indicate stress and deformation values before and after a small-time increment Δt.

.

Starting from Equation 26, it is possible to separate σ_i^t and then to sum the results over all arms of the GMM and to add the stress in the equilibrium spring, following the pattern of Equation 21. This leads to the general Equation 27 that can be used to calculate the stress at any time t and for any thermal excitation, in an incremental fashion (Roylance, 2001).

$$\sigma^t=E_0\varepsilon^t+\sum_i\frac{E_i\left(\varepsilon^t-\varepsilon^{t-1}\right)+\sigma_j^{t-1}}{1+\left(\Delta t/\tau_i\right)} \qquad \text{Equation 27.}$$

Where, due to the TTSP, $t=\xi\wedge\Delta t=\Delta\xi$

Despite the simplicity of such process and its easy numerical implementation, it could be observed a strong dependence of the results on tiny fluctuations of $E(\xi)$. Indeed, and as shown in previous figures, the optimization of a Prony Serie on the interconverted or experimental-shifted values of ErelMC induces a more or less pronounce waviness of $E(\xi)$, which in turn generates strong fluctuations in the calculation of σ^t at certain steps.

To solve this problem, the time-step Δt was artificially decreased to a small reasonable value using a linear interpolation scheme on the thermal profile, to ensure that Equation 27 was solved in a time-stable fashion. Nevertheless, this adjustment does not allow the complete correction of the observed phenomenon, as seen below in Figure 28.

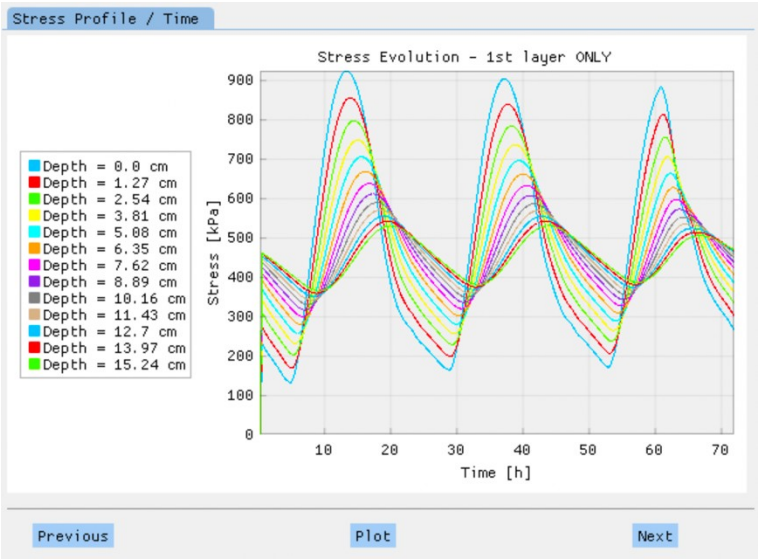


Figure 28. Plot of the Stress Calculation Using Finite Difference Method (FDM)

Another calculation scheme, relying on a more stable implementation, was thus implemented and tested.

Boltzman Superposition Integral:

Starting from the GMM and Equation 25, it is possible to transform the ODE into an integral equation. Indeed, integrals consist in continuously summing operations, and it is thus possible to express the material viscoelasticity at a given time as the continuous sum (integral) of the responses to excitations imposed at all previous times. Therefore, such type of integral is also known as "hereditary integral", since the resolution of it at time *t*receives from past "experience". In the present case, integral Equation 28 is obtained (Roylance, 2001).

$$\sigma(t) = \int_{0^{+}¿}^{t} Erel(t-\xi)\,\mathrm{d}\,\varepsilon = \int_{0^{+}¿}^{t} Erel(t-\xi)\frac{\mathrm{d}\,\varepsilon(\xi)}{\mathrm{d}\,\xi}\,\mathrm{d}\,\xi \qquad \text{Equation 29.}$$

When facing complex excitation, as in the present case, finding a solution to this hereditary integral can be challenging, even using Fourier or Laplace integral transforms. A way to solve it is to use correspondence principles, that allow converting the stress-strain integral relationship into a simple elastic-like (i.e. linear) stress-strain relation (Schapery, 1984) (Schapery, 1999). In

particular, to tackle viscoelastic problems, Schapery proposed the use of known as “pseudo-variables” (Schapery, 1975) (Schapery, 1975).

In short, the pseudo-variables method states that stresses in an elastic and viscoelastic body are the same, and thus, that it is possible to solve a viscoelastic problem using the set of Equations 29 and 30 considering a Generalized Maxwell Model as stated in Equation 23.

$$\sigma = E^R \varepsilon^R \qquad \text{Equation 29.}$$

$$\varepsilon^R = \frac{1}{E^R} \int_{0^+ ¿}^{t} \left[E_{glassy} - \sum_{i=1}^{N} E_i \left(1 - e^{\frac{(t-\xi)}{\tau_i}} \right) \right] \frac{\mathrm{d}\,\varepsilon(\xi)}{\mathrm{d}\,\xi} \mathrm{d}\,\xi \qquad \text{Equation 30.}$$

A numerical implementation for isotropic material proposed by Hinterhoelzl and Schapery has been employed to integrate the equation discreetly in time, through the calculation of pseudo-variables increments (Hinterhoelzl & Schapery, 2004) (Ozer, 2020) (Ozer, 2014) (Ozer, 2020). This method showed better stability regardless of the time increment, as much as a high reliability. This technique is thus the one employed to compute all thermal stress calculations presented below.

- Stress vs. Time at all depth plot

Once the user validates the obtained Prony Serie for $E(\xi)$, the stress calculation is automatically realized, and the user is invited to “Plot” the results. The results are here shown “only for the 1st layer”, since the stress calculation is realized solely for this part of the pavement (refer Figure 29).

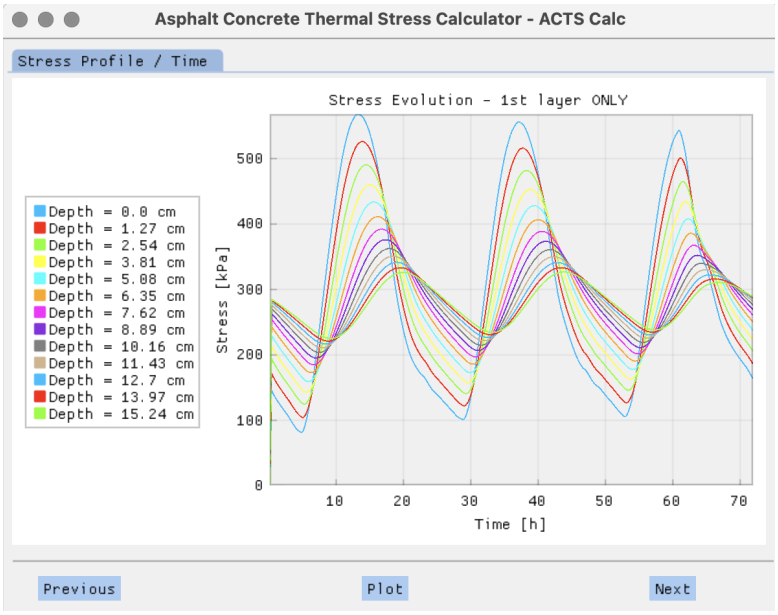


Figure 29. Stress Profile Plot for the full pavement depth

As for the thermal profile plot, the preceding one is highly interactive: the legend can be shown or hidden in few steps (right click on the plot); the stress evolution at a given depth can be hidden by clicking on it in the legend; figure and/or axes can be zoomed in and out.

As the plot is showed, the program produces an Excel table containing all the results shown in the original plot (i.e., including all depths), with the stress values expressed in [GPa]. In this table, the rows are related to the time step [s] whereas the columns refer to the depth-step [cm] (refer Figure 30).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1		**0**	**1.27**	**2.54**	**3.81**	**5.08**	**6.35**	**7.62**	**8.89**	**10.16**	**11.43**	**12.7**	**13.97**	**15.24**
2														
3	**900**	0	0	0	0	0	0	0	0	0	0	0	0	0
4	**1800**	0.000145	0.000171	0.000194	0.000213	0.000229	0.000242	0.000253	0.000263	0.00027	0.000276	0.00028	0.000282	0.000284
5	**2700**	0.00014	0.000166	0.000189	0.000208	0.000225	0.000239	0.00025	0.00026	0.000267	0.000273	0.000278	0.000281	0.000282
6	**3600**	0.000135	0.000161	0.000184	0.000204	0.000221	0.000235	0.000247	0.000257	0.000265	0.000271	0.000275	0.000279	0.000281
7	**4500**	0.000131	0.000156	0.000179	0.000199	0.000217	0.000231	0.000244	0.000254	0.000262	0.000268	0.000273	0.000277	0.000279
8	**5400**	0.000127	0.000152	0.000175	0.000195	0.000213	0.000228	0.00024	0.000251	0.000259	0.000266	0.000271	0.000275	0.000277
9	**6300**	0.000123	0.000148	0.000171	0.000191	0.000209	0.000224	0.000237	0.000248	0.000257	0.000264	0.000269	0.000273	0.000275
10	**7200**	0.00012	0.000145	0.000167	0.000188	0.000205	0.000221	0.000234	0.000245	0.000254	0.000261	0.000267	0.000271	0.000274
11	**8100**	0.000116	0.000141	0.000164	0.000184	0.000202	0.000217	0.000231	0.000242	0.000251	0.000259	0.000264	0.000269	0.000272
12	**9000**	0.000113	0.000137	0.00016	0.00018	0.000198	0.000214	0.000227	0.000239	0.000248	0.000256	0.000262	0.000267	0.00027
13	**9900**	0.000109	0.000134	0.000156	0.000177	0.000195	0.000211	0.000224	0.000236	0.000246	0.000254	0.00026	0.000265	0.000268
14	**10800**	0.000105	0.00013	0.000153	0.000173	0.000191	0.000207	0.000221	0.000233	0.000243	0.000251	0.000257	0.000262	0.000266
15	**11700**	0.000101	0.000127	0.000149	0.00017	0.000188	0.000204	0.000218	0.00023	0.00024	0.000248	0.000255	0.00026	0.000264
16	**12600**	9.68E-05	0.000123	0.000146	0.000166	0.000185	0.000201	0.000215	0.000227	0.000237	0.000246	0.000253	0.000258	0.000262
17	**13500**	9.28E-05	0.000119	0.000142	0.000163	0.000181	0.000198	0.000212	0.000224	0.000235	0.000243	0.00025	0.000256	0.00026
18	**14400**	8.88E-05	0.000115	0.000138	0.000159	0.000178	0.000195	0.000209	0.000222	0.000232	0.000241	0.000248	0.000254	0.000258
19	**15300**	8.57E-05	0.000111	0.000135	0.000156	0.000175	0.000191	0.000206	0.000219	0.000229	0.000238	0.000246	0.000252	0.000256
20	**16200**	8.34E-05	0.000108	0.000131	0.000152	0.000171	0.000188	0.000203	0.000216	0.000227	0.000236	0.000243	0.000249	0.000254
21	**17100**	8.16E-05	0.000106	0.000128	0.000149	0.000168	0.000185	0.0002	0.000213	0.000224	0.000233	0.000241	0.000247	0.000252
22	**18000**	8.01E-05	0.000104	0.000126	0.000146	0.000165	0.000182	0.000197	0.00021	0.000221	0.000231	0.000239	0.000245	0.00025
23	**18900**	8.41E-05	0.000103	0.000124	0.000144	0.000162	0.000179	0.000194	0.000207	0.000219	0.000228	0.000236	0.000243	0.000248
24	**19800**	9.22E-05	0.000107	0.000124	0.000142	0.00016	0.000177	0.000191	0.000205	0.000216	0.000226	0.000234	0.000241	0.000246
25	**20700**	0.000103	0.000112	0.000126	0.000142	0.000159	0.000174	0.000189	0.000202	0.000214	0.000223	0.000232	0.000238	0.000244
26	**21600**	0.000115	0.00012	0.00013	0.000143	0.000158	0.000173	0.000187	0.0002	0.000211	0.000221	0.000229	0.000236	0.000242
27	**22500**	0.00013	0.000129	0.000136	0.000146	0.000159	0.000172	0.000185	0.000198	0.000209	0.000219	0.000227	0.000234	0.00024
28	**23400**	0.000148	0.000141	0.000143	0.00015	0.00016	0.000172	0.000184	0.000196	0.000207	0.000217	0.000225	0.000232	0.000238
29	**24300**	0.000167	0.000155	0.000152	0.000155	0.000163	0.000173	0.000184	0.000195	0.000205	0.000215	0.000223	0.00023	0.000236
30	**25200**	0.000187	0.00017	0.000162	0.000162	0.000167	0.000175	0.000184	0.000194	0.000204	0.000213	0.000221	0.000228	0.000234
31	**26100**	0.000208	0.000186	0.000174	0.00017	0.000172	0.000178	0.000186	0.000194	0.000203	0.000212	0.00022	0.000226	0.000232
32	**27000**	0.000229	0.000202	0.000187	0.000179	0.000178	0.000181	0.000187	0.000195	0.000203	0.000211	0.000218	0.000225	0.00023
33	**27900**	0.000251	0.00022	0.0002	0.000189	0.000185	0.000186	0.00019	0.000196	0.000203	0.00021	0.000217	0.000223	0.000229
34	**28800**	0.000273	0.000237	0.000214	0.0002	0.000193	0.000191	0.000193	0.000198	0.000204	0.00021	0.000216	0.000222	0.000227
35	**29700**	0.000294	0.000256	0.000228	0.000211	0.000201	0.000197	0.000197	0.0002	0.000205	0.00021	0.000216	0.000221	0.000226
36	**30600**	0.000315	0.000273	0.000243	0.000223	0.00021	0.000204	0.000202	0.000203	0.000206	0.000211	0.000216	0.000221	0.000225
37	**31500**	0.000337	0.000292	0.000258	0.000235	0.00022	0.000211	0.000207	0.000206	0.000208	0.000212	0.000216	0.00022	0.000225
38	**32400**	0.000358	0.00031	0.000274	0.000247	0.00023	0.000218	0.000212	0.00021	0.000211	0.000213	0.000216	0.00022	0.000224
39	**33300**	0.000378	0.000328	0.000289	0.00026	0.00024	0.000227	0.000218	0.000215	0.000214	0.000215	0.000217	0.000221	0.000224
40	**34200**	0.000397	0.000345	0.000304	0.000273	0.000251	0.000235	0.000225	0.000219	0.000217	0.000217	0.000219	0.000221	0.000224
41	**35100**	0.000417	0.000362	0.000319	0.000286	0.000261	0.000244	0.000232	0.000225	0.000221	0.00022	0.00022	0.000222	0.000225
42	**36000**	0.000436	0.000379	0.000334	0.000299	0.000272	0.000253	0.000239	0.00023	0.000225	0.000223	0.000222	0.000223	0.000225
43	**36900**	0.000453	0.000396	0.000349	0.000312	0.000283	0.000262	0.000246	0.000236	0.000229	0.000226	0.000224	0.000225	0.000226
44	**37800**	0.000469	0.000411	0.000364	0.000325	0.000294	0.000271	0.000254	0.000242	0.000234	0.000229	0.000227	0.000226	0.000227
45	**38700**	0.000485	0.000426	0.000377	0.000337	0.000305	0.00028	0.000262	0.000248	0.000239	0.000233	0.00023	0.000229	0.000229
46	**39600**	0.0005	0.000441	0.000391	0.00035	0.000316	0.00029	0.00027	0.000255	0.000244	0.000237	0.000233	0.000231	0.00023
47	**40500**	0.000513	0.000454	0.000404	0.000362	0.000327	0.000299	0.000278	0.000262	0.00025	0.000242	0.000236	0.000233	0.000232
48	**41400**	0.000524	0.000466	0.000416	0.000373	0.000337	0.000309	0.000286	0.000268	0.000255	0.000246	0.00024	0.000236	0.000234
49	**42300**	0.000534	0.000477	0.000427	0.000384	0.000348	0.000318	0.000294	0.000275	0.000261	0.000251	0.000244	0.000239	0.000237

Figure 30. Stress Profile Table (GPa) vs. Time

The table can be found in the “Results” directory, under the name "calculated_stress_GPa.xlsx".

- Depth vs. Temperature at a given time plot

The "Next" tab allows observing the stress profile of the pavement at a given time-step. To do so, the user is invited to select the desired time to plot the stress profile through a drop-down menu. Once a time is selected, the plot automatically is actualized, and the user is given the option to "Export" the data. The user also can choose another time to visualize and export. The results are here shown only for the 1st layer, since the stress calculation is realized solely for this part of the pavement (refer Figure 31). The depth is thus adjusted accordingly.

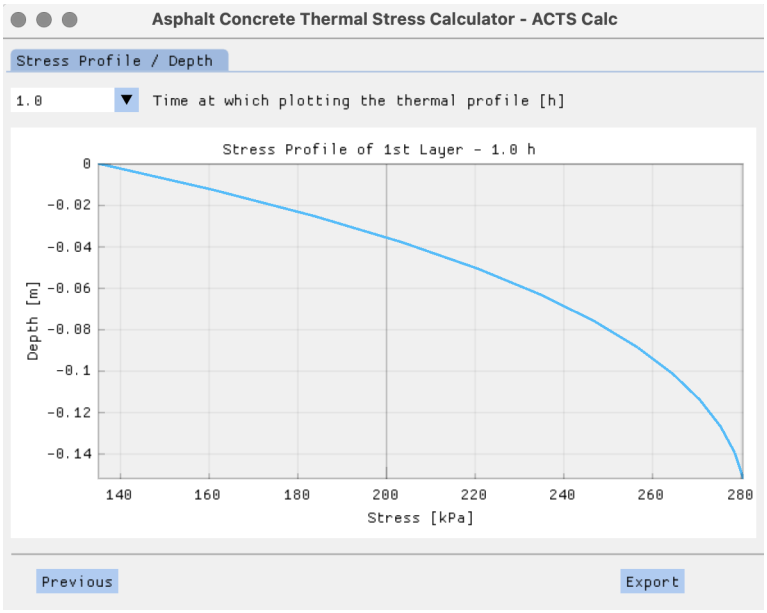


Figure 31. Stress Profile Plot at a specific time.

If the "Export" button is clicked, a new Excel file is generated by the program. In this document, the rows represent the discrete depth-step of the slab, whereas the single column is related to the time of the stress profile. All stress values are expressed in GPa. The table can be found in the "Results" directory, under the name "depthVSstress_GPa_*xx*h.xlsx".

References

Adamczak, S. & Bochnia, J., 2016. Estimating the Approximation Uncertainty for Digital Materials Subjected to stress Relaxation Tests. *Metrology and Measurement Systems,* 23(4), pp. 545-553.

Alavi, M., Hajj, E. Y. & Sebaaly, P. E., 2017. A comprehensive model for predicting thermal cracking events in asphalt pavements. *International Journal of Pavement Engineering,* September, 18(9), pp. 871-885.

Alavi, S. M. Z., 2014. *Comprehensive methodologies for analysis of thermal cracking in asphalt concrete pavements.* Nevada, Reno: University of Nevada.

Gui, J., P.E., P., K.E., K. & J.S., G., 2007. Impact of Pavement Thermophysical Properties on Surface Temperatures. *J.Mater. Civ. Eng.,* 19(8), pp. 683-690.

Hinterhoelzl, R. M. & Schapery, R. A., 2004. FEM Implementation of a Three-Dimensional Viscoelastic Constitutive Model for Particulate Composites with Damage Growth. *Mechanics of Time-Dependent Materials,* March, 8(1), pp. 65-94.

Lytton, R. et al., 1993. *An integrated model of the climatic effects on pavements.* McLean, Virginia United States : United States. Federal Highway Administration. Office of Engineering and Highway Operations R&D.

NCAT, 2009. *A Review of Aggregate and Asphalt Mixture Specifi Gravity Measurments and Their Impacts on Asphalt Mix Design Properties and Mix Acceptance,* s.l.: s.n.

NCAT, 2012. *Effect of Rejuventation on Performance Properties of HMA Mixtures with High RAP and RAS Contents,* s.l.: s.n.

Olsen, N. B., Christensen, T. & Dyre, J. C., 2001. Time-Temperature Superposition. *Phys. Rev. Lett,* Volume 86, p. 1271.

Ozer, H., 2014. *3-D Generalization and Pseudo Variables.* s.l.:s.n.

Ozer, H., 2020. *Part 1: Viscoelasticity L02: Integral Operators and Standard Excitations.* s.l.:s.n.

Ozer, H., 2020. *Viscoelastic Material Model (Based on Abaqus Viscoelasticity Documentation and Hinterhoelzl and Schapery).* s.l.:s.n.

Park, S. & Kim, Y., 1999. Interconversion between Relaxation Modulus and Creep Compliance for Viscoelastic Solids. *Journal of Materials,* February, 11(1), pp. 76-82.

Park, S. & Kim, Y., 2001. Fitting Prony-Series Viscoelastic Models with Power-Law Presmoothing. *J. Mater. Civ. Eng.,* February, 13(1), pp. 26-32.

Roylance, D., 2001. *Engineering Viscoelasticity.* s.l.:Massachusetts Institute of Technology.

Saboo, N. & Kumar, P., 2018. Equivalent Slope Method for Construction of Master Curve. *Indian Highways,* 46(2), pp. 19-28.

Schapery, R. A., 1975. A theory of crack initiation and growth in viscoelastic media. *Int J Fract,* February, 11(1), pp. 141-159.

Schapery, R. A., 1975. A theory of crack initiation and growth in viscoelastic media II. Approximate methods of analysis. *Int J Fract,* June, 11(3), pp. 369-388.

Schapery, R. A., 1984. 'Correspondence principles and a generalizedJ integral for large deformation and fracture analysis of viscoelastic. *Int J Fract,* July, 25(3), pp. 195-223.

Schapery, R. A., 1999. Nonlinear viscoelastic and viscoplastic constitutive equations with growing damage. *International Journal of Fracture,* April, 97(1), pp. 33-66.

SciPy, 2021. *Scientific computing tools for Python.* [Online]
Available at: https://www.scipy.org/about.html
[Accessed 15 September 2021].

Witczak, M. & Bari, J., 2004. *Development of a Master Curve (E*) Database for Lime Modified Asphaltic Mixtures,* Tempe: Arizona State University.