



QSA Q-Bat Explorer — Theory Guide

QuickerSim Automotive Ltd.

Contents

1. Software Philosophy	3
2. Mathematical Models of Battery Components	5
2.1. Cell	7
2.2. Cooling Plate	10
2.3. Passive Heat Component	13
2.4. Thermal Pads & Other Contacts	14
2.5. Internal Air & Battery Housing	15
2.6. Aggregating components	17
3. Discretization	19
3.1. Mesh Generation & FEM	19
3.2. Model Reduction	21
3.3. Cooling fluid channel discretization	23

1. Software Philosophy

QSA Q-Bat Explorer is an innovative software designed for heat transfer calculations in battery packs by the means of Finite Element Method (FEM). The software uses Reduced Order Modelling (ROM) to reduce the computation complexity of the problems and provide the user with accurate thermal estimations. Q-Bat Explorer can be used to calculate both individual cells, as well as batteries or whole battery packs coupled hydraulically.

This Theory Guide provides basic theoretical information about the models used in Q-Bat Explorer and describes the software convention, in order to help the User understand the Framework workflow.

The software's UI is developed in a prototype-base convention. At first the User creates a prototype of each of the unique components in the assembly. Component prototypes are only virtual and therefore are not automatically added to the simulation domain. At this stage all of the physical properties of the element are specified and boundary conditions are applied.

Elements are added to the physical domain by specifying location matrices of their occurrences. If only one instance is needed in the assembly, specifying the array is still necessary (it will be an 3x1 array containing the location of the body). All elements can be freely rotated and translated in the domain. If a similar component with different physical properties or boundary conditions is needed, a second base component (prototype) is required to be created.

EXAMPLE

The User wants to simulate heat transfer in a battery containing 6 cells. All of the cells share the same geometry and material properties, but 3 of them are discharged by 200 A, while the other 3 - by twice as much (400 A). In order to create a model of the battery the User has to create two different cell bases and replicate both of them, specifying two matrices which describe the location of the cells.

To depict the steps needed to create the cells of the simulated battery a hierarchical tree is displayed (Figure 1).

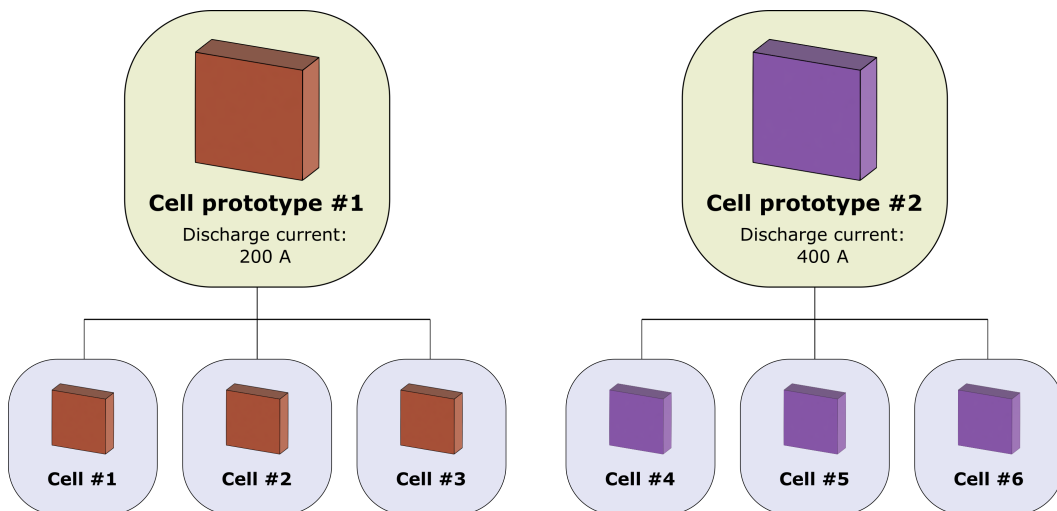


Figure 1: Example: hierarchical tree of cells and cell bases

This way the User creates a model consisting of 6 cells, shown in Figure 2. Omission of cell base replication (creating only the cell bases) would result in an empty model, as cells would not be added to the simulation domain.

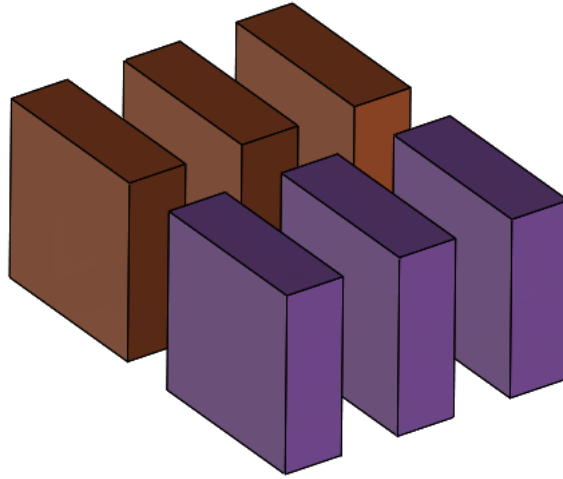


Figure 2: Example: cell arrangement

After specifying all of the components of the assembly (their bases and locations), the collective battery model is created by assigning the component arrays to a higher-level aggregating component (described in detail in Subsection 2.6).

Thermal contacts (Subsection 2.4) and the initial temperature is specified and applied to specific components or the whole assembly. The model is reduced and global mass and diffusion matrices are assembled.

The User defines the parameters of the simulation (length of time step and number of time steps) and calculations are started. Solutions from each time step can be displayed and exported afterwards.

To perform a steady-state calculation the User should specify a very large time step (order of thousands of seconds).

All of the parameters and calculations are in SI units, except for temperature, which is given in degrees Celsius.

2. Mathematical Models of Battery Components

This section focuses on specific battery components and how they are modeled inside the software. Equations and methods applied to describe the properties and state of the components are presented in each of the subsections.

Three main types of elements that can be composed in Q-Bat Explorer include:

- Cell
- Cooling plate
- Passive heat component

By using different arrangements of these components the User is able to model any battery.

Methods for creating different elements, which do not match the ones listed above (such as a thermal pad or battery housing), are described in the following subsections.

A sample geometry used for visualizing specific component models is presented below (Fig. 3).

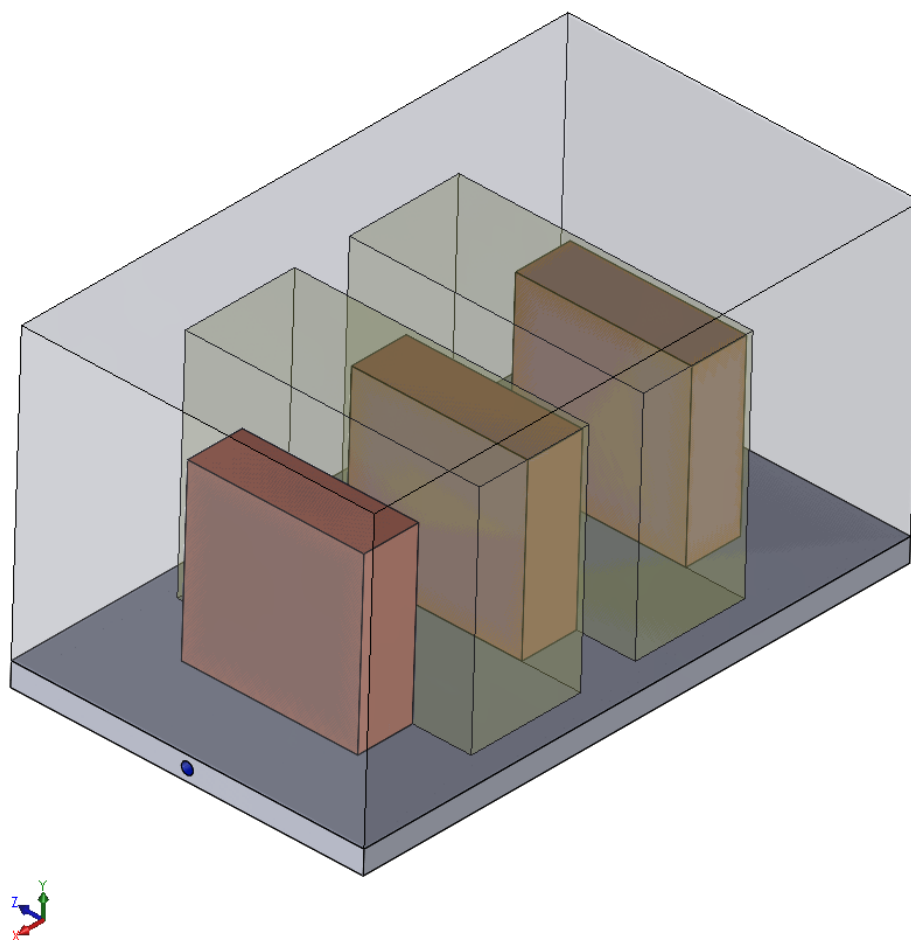


Figure 3: Sample geometry

The geometry comprises three cells (colored red) separated by passive heat components (yellow), standing on a cooling plate (grey) with a cylindrical pipe. Between the cells and cooling plate a thermal pad is applied.

The battery housing (shown in transparent grey) surrounds all of the other elements.

2.1 Cell

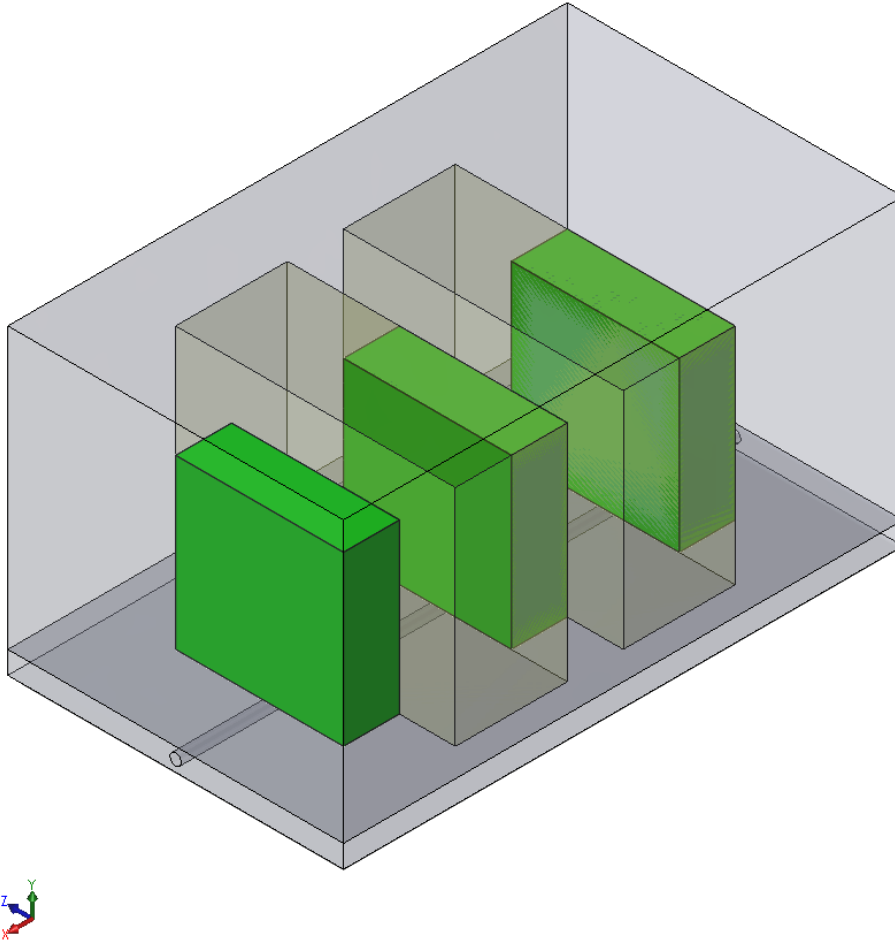


Figure 4: Sample geometry - cells

Cells are modeled as prismatic or cylindrical solid bodies.

Heat generation can be set by either specifying the volumetric heat generation (in $[\frac{W}{m^3}]$) or the electrical properties of the cell and applied current load (in [A]), that can vary in time. To capture the instantaneous capacitive effects, cells are modelled as RC circuits. A model of the circuit is presented below (Fig. 5).

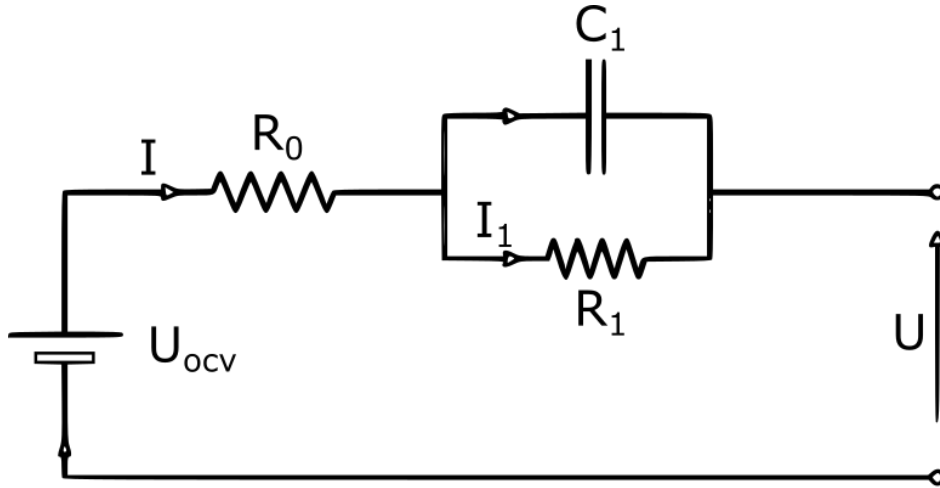


Figure 5: RC circuit

In the presented diagram U_{ocv} is the open circuit voltage and R_0 , R_1 and C_1 are the parameters of the model (resistance and capacitance).

Two governing equations describing the cell's instantaneous state are derived from Kirchhoff's circuit laws:

$$U = U_{ocv} - I * R_0 - U_1 [V] \quad (1)$$

$$I = U_1 + C_1 * dU_1/dt [A] \quad (2)$$

Where U_1 is the voltage drop on the RC pair [V], U is the total cell voltage [V] and I is the cell current [A].

Heat generation is calculated using the Joule heating equation:

$$P = I^2 * R_0 + I_1^2 * R_1 [W] \quad (3)$$

Both heat source and electric current (inputs to the model) can be given as a constant scalar or linear or non-linear matrix defined in an Excel spreadsheet.

By default the heat is generated as uniform distribution in the cell. This can be modified by the User by specifying the distribution matrix.

Within the cell heat is transferred by the means of diffusion, following Fourier's law of thermal conduction, described in a differential form by:

$$\mathbf{q} = -\lambda * \nabla T \left[\frac{W}{m^2} \right] \quad (4)$$

Where q $\left[\frac{W}{m^2} \right]$ is the local heat flux density, λ $\left[\frac{W}{m^2 * K} \right]$ denotes thermal conductivity and ∇T $\left[\frac{K}{m} \right]$ is the local temperature gradient.

Material properties of the cell are defined in an Excel spreadsheet. Properties can be constant scalars or can be given as a table of variables depending on temperature. Electrical properties can also depend on the cell's State of Charge (SOC).

If any of the given properties are temperature-dependent or SOC-dependent variables, Q-Bat Explorer approximates them linearly between the given points and fixes their values after exceeding the marginal values of the independent variables.

Q-Bat Explorer supports anisotropic heat conduction - values of λ can be given in an Excel spreadsheet or entered by the User directly. If the value of λ is given as a 1x1 scalar, the cell is by default treated as isotropic.

2.2 Cooling Plate

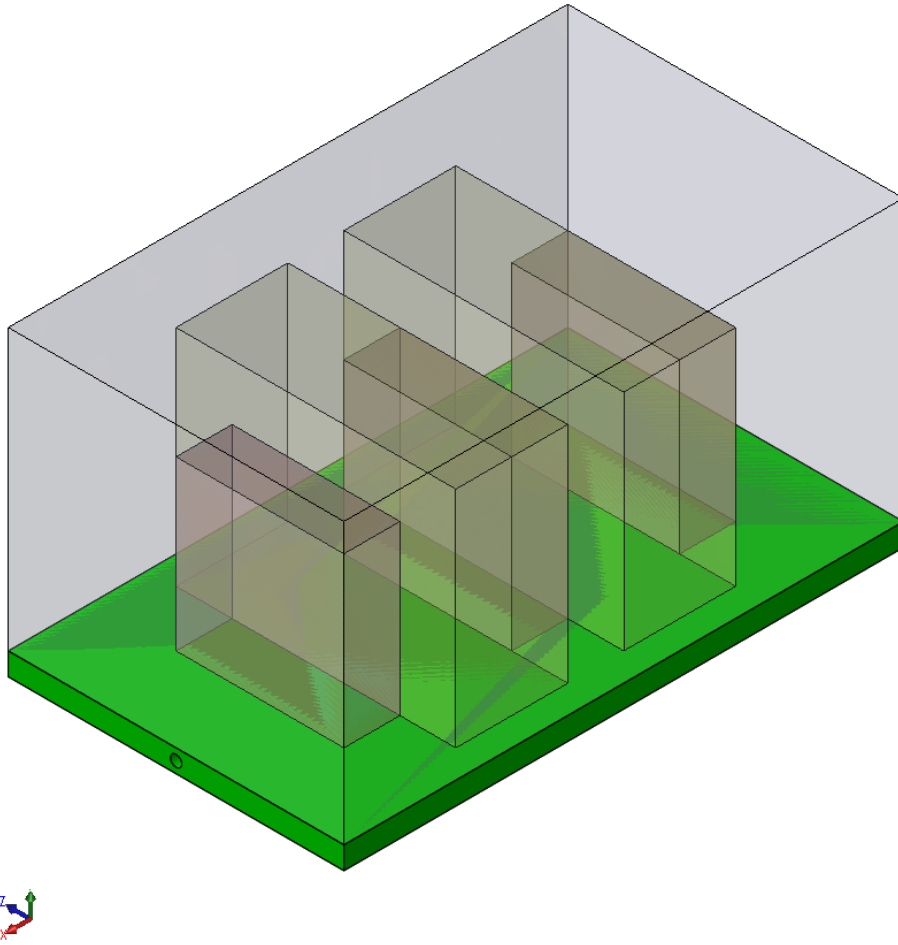


Figure 6: Sample geometry - cooling plate

A cooling plate is modeled as a solid body containing a solid body pipe. The volume of the cooling fluid is modeled as a void in the geometry.

Within the cooling plate heat is transferred by the means of diffusion, following Fourier's law of thermal conduction, described in a differential form by:

$$\mathbf{q} = -\lambda * \nabla T \left[\frac{W}{m^2} \right] \quad (5)$$

Where $q \left[\frac{W}{m^2} \right]$ is the local heat flux density, $\lambda \left[\frac{W}{m^2 * K} \right]$ denotes thermal conductivity of the cooling plate material and $\nabla T \left[\frac{K}{m} \right]$ is the local temperature gradient.

Material properties of the cooling plate are defined in an Excel spreadsheet. Properties can be constant scalars or can be given as a table variables depending on temperature. When defining the material properties during cooling plate creation, two different Excel spreadsheets are needed - one describing the properties of the cooling plate, and the other for the cooling fluid. The cooling pipe is omitted - the cooling plate and pipe are therefore modeled as one body.

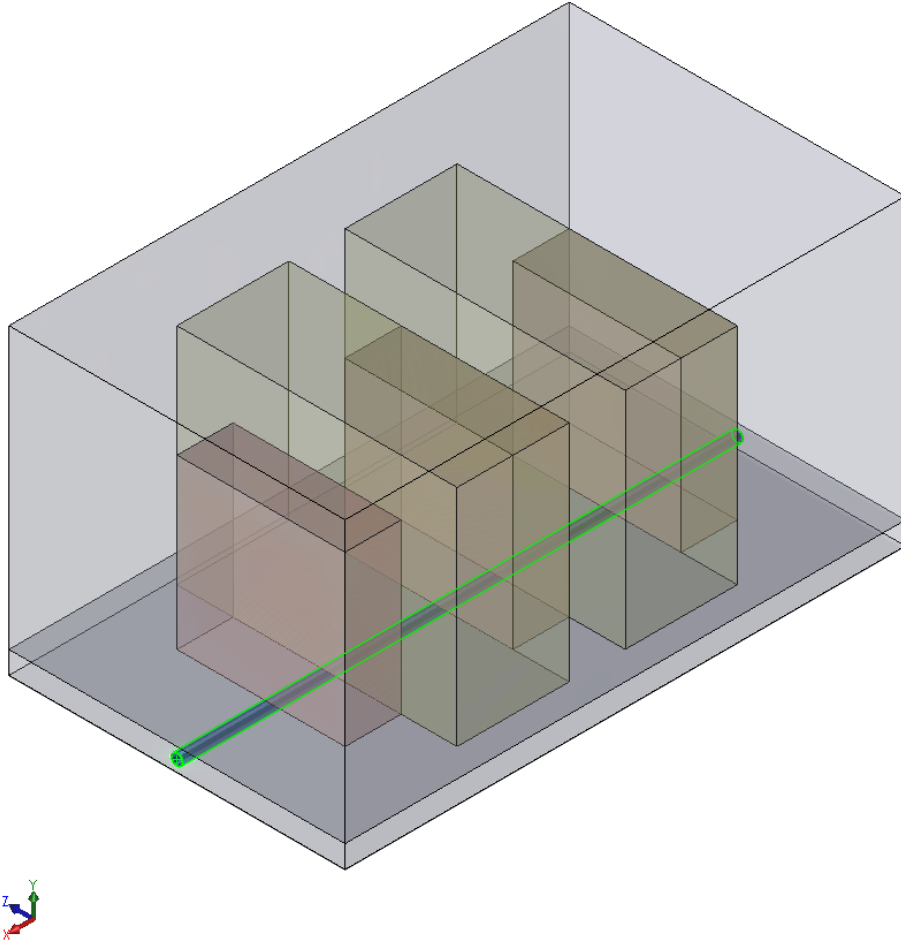


Figure 7: Sample geometry - cooling fluid

The cooling fluid is modeled as a one dimensional fluid flow. Apart from setting physical properties of the fluid, it is necessary to specify the hydraulic diameter of the pipe (d [m]), mass flow rate (in [$\frac{kg}{s}$]) and the fluid inlet temperature(in [$^{\circ}C$]).

Heat between the fluid and cooling pipe is transferred by the means of forced convection, quantified by Newton's Law of cooling:

$$\mathbf{q} = \alpha * \Delta T \left[\frac{W}{m^2} \right] \quad (6)$$

Where $\alpha \left[\frac{W}{m^2 * K} \right]$ is the heat transfer coefficient and ΔT [K] is the difference in the cooling fluid's temperature between the wall temperature and its surrounding fluid.

The value of α can be given by the User or calculated by Q-Bat Explorer from the definition of the Nusselt number, where d is the hydraulic diameter of the pipe [m].

$$\alpha = Nu * \frac{\lambda}{d} \quad (7)$$

The Nusselt number (Nu) depends on the character of the flow and can be obtained from [2]:

-
- For Reynold's number $Re < 2000$ the Nusselt number is given by Michiejew's equation for laminar flows. Grashof number has been omitted, as heat transfer from forced convection is much greater than that from natural convection, and would require additional User input:

$$Nu = 0.15 * Re^{0.33} * Pr_f^{0.43} * \left(\frac{Pr_f}{Pr_w}\right)^{0.25} * E_L [-] \quad (8)$$

- For $2000 < Re < 10000$ the Nusselt number is given by Michiejew's equation for transitional flows:

$$Nu = K_0 * Pr_f^{0.43} * \left(\frac{Pr_f}{Pr_w}\right)^{0.25} [-] \quad (9)$$

- For $10000 < Re < 5000000$ (fully turbulent flow) Sieder and Tate's equation is used:

$$Nu = 0.023 * Re^{0.8} * Pr_f^{0.33} * \left(\frac{\mu_f}{\mu_w}\right)^{0.14} [-] \quad (10)$$

In equations (8)-(10) Pr_f and Pr_w denote the Prandtl number and μ_f and μ_w [$Pa * s$] are the values of dynamic viscosity in the fluid and on the wall surface respectively. E_L is a correction factor based on the L/d ratio (where L - length of pipe [m]). The set of E_L values for different L/d ratios are given in Table 1.

L/d	1	2	5	10	15	20	30	40	50
E_L	1.9	1.7	1.44	1.28	1.18	1.13	1.05	1.02	1

Table 1: E_L correction factor values for laminar flows

K_0 is a correction factor based on the flow's Reynold's number. A table of K_0 values is presented below.

Re	2200	2300	2500	3000	3500	4000	5000	6000	7000	8000	9000	10000
K_0	2.2	3.6	4.9	7.5	10	12.2	16.5	20	24	27	30	33

Table 2: K_0 correction factor values

The values of K_0 and E_L are interpolated linearly between the points specified in the tables.

Higher values of Re are not supported, as they are non-physical in this application.

2.3 Passive Heat Component

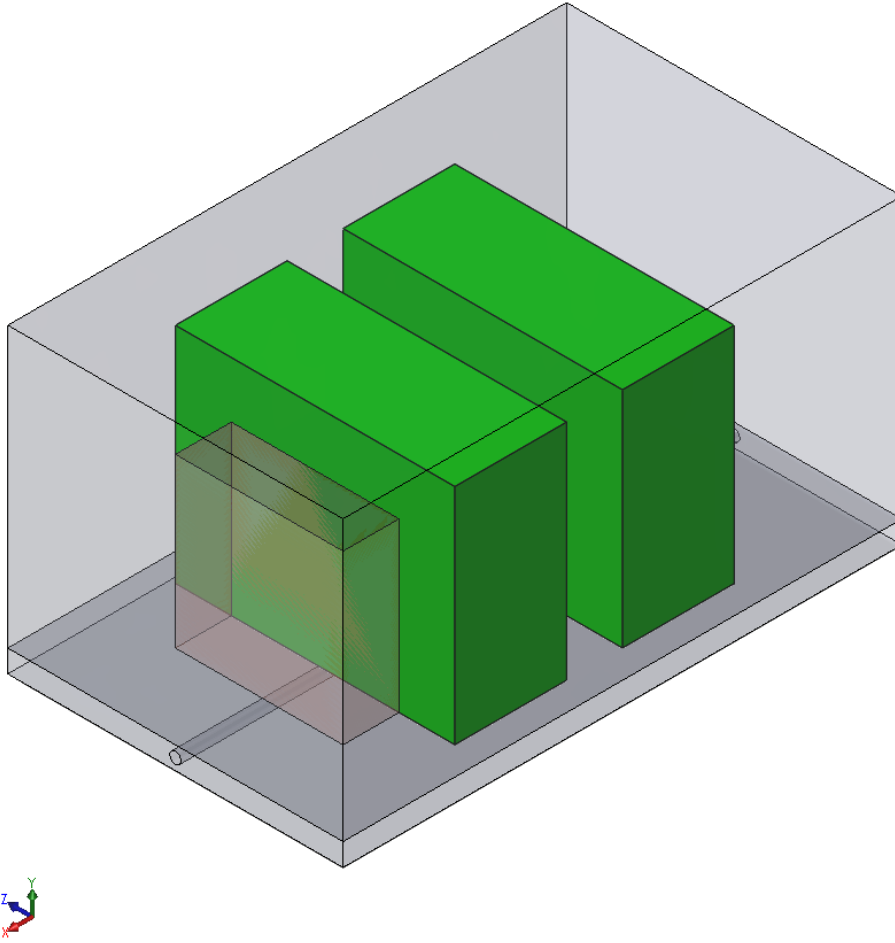


Figure 8: Sample geometry - heat component

Passive heat components are modeled by solid bodies of any geometry.

Within the component heat is transferred by the means of diffusion, following Fourier's law of thermal conduction, described in a differential form by:

$$\mathbf{q} = -\lambda * \nabla T \left[\frac{W}{m^2} \right] \quad (11)$$

Where $q \left[\frac{W}{m^2} \right]$ is the local heat flux density, $\lambda \left[\frac{W}{m^2 * K} \right]$ denotes thermal conductivity of the components and $\nabla T \left[\frac{K}{m} \right]$ is the local temperature gradient.

Material properties of the heat component are defined in an Excel spreadsheet. Properties can be constant scalars or can be given as a table of variables depending on temperature.

Passive heat components are not meant to generate heat. Although it is possible to assign a heat source to a passive heat component this practice is not recommended.

2.4 Thermal Pads & Other Contacts

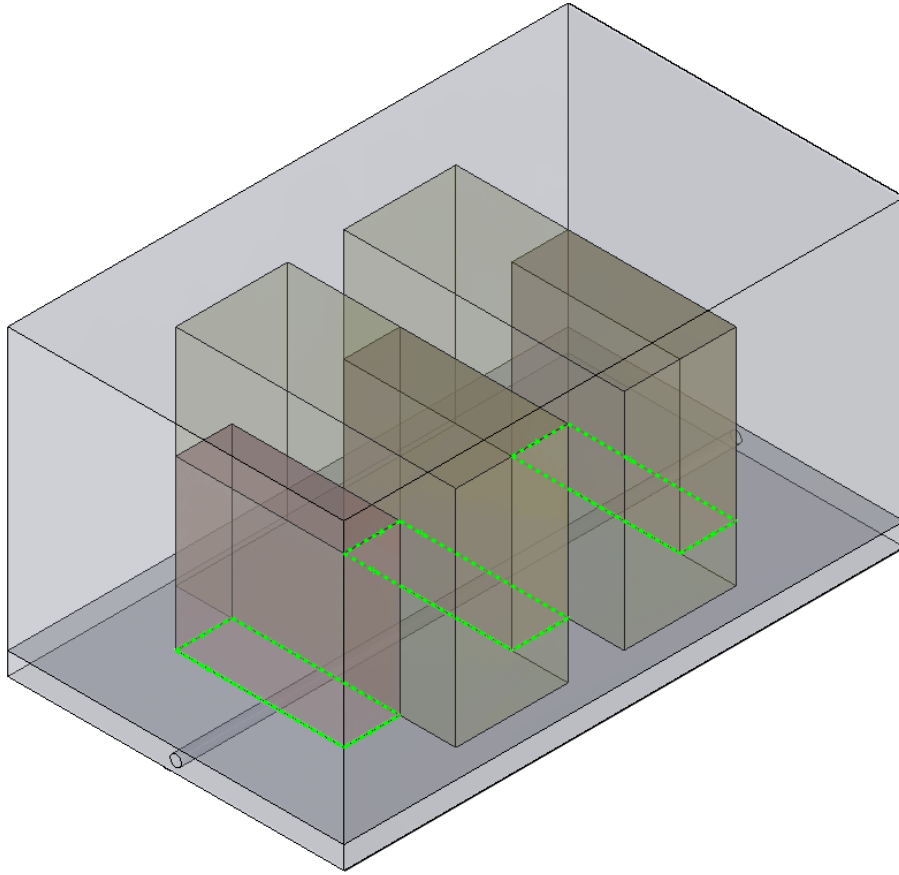


Figure 9: Sample geometry - thermal pad

By default the conductance of adhering components is equal to zero, meaning heat transfer between them does not occur. In order to allow heat transfer between the elements it is necessary to add contacts to the model (and specify their conductance).

Contacts are defined by specifying thermal conductivity λ [$\frac{W}{m \cdot K}$] and thickness δ [m] of a contact. Thermal pads and other significantly thinner components are modeled as thermal contacts with a specified conductance.

Heat transfer on the contact is described by the one-dimensional form of Fourier's Law:

$$\mathbf{q} = \frac{-\lambda}{\delta} * \Delta T \left[\frac{W}{m^2} \right] \quad (12)$$

2.5 Internal Air & Battery Housing

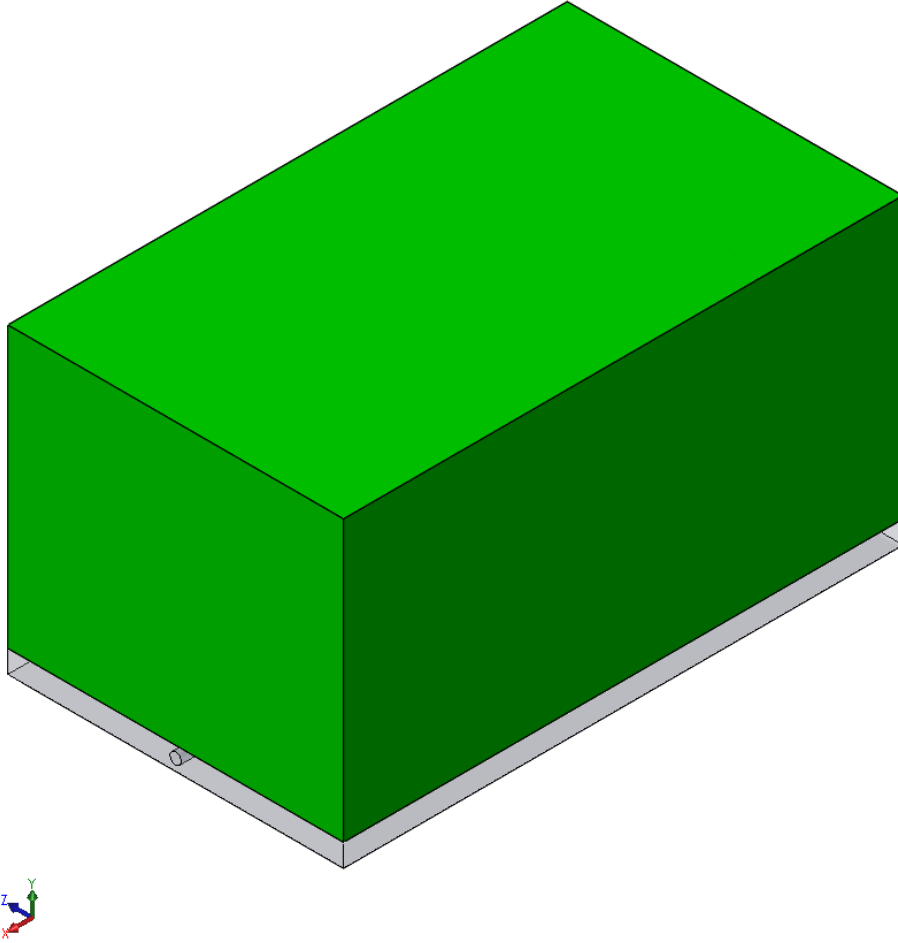


Figure 10: Sample geometry - housing

At this point natural convection is not supported by Q-Bat Explorer. This is to be added in later editions of the software.

A temporary solution involves creating a domain of air (as a passive heat component) with artificial deflation of its thermal conductance.

To simulate a battery housing it is recommended to create a passive heat component with the material properties of air and a corresponding geometry. Thermal contacts between the air and adhering components are required to be added. The value of the heat transfer coefficient α can be broadly approximated by the equations for natural convection near a single vertical wall (described in [3]):

$$Nu_l = 0.6773 * \left(\frac{Pr}{(Pr + 0.952)} * Ra_l \right)^{0.25} \quad [-] \quad (13)$$

$$\alpha = \frac{Nu_l * \lambda}{l} \left[\frac{W}{m^2 * K} \right] \quad (14)$$

Where Nu_l is the average Nusselt number, Pr is the Prandtl number for air and Ra is the Rayleigh number for the assumed air flow. The height of the wall is denoted by l [m], $\lambda \left[\frac{W}{m * K} \right]$ is the thermal coefficient of air.

The obtained values should range between 5-10 $\frac{W}{m^2 * K}$, what is in accordance with typical values of α in natural convection in air.

While specifying the contact between components and the surrounding air, λ should be equal to the value of α obtained from eq. 14 and the value of thickness δ should be equal to 1 (the summary conductance will then be in the range of 5-10 $\frac{W}{m^2 * K}$).

To simulate the heat transfer through the wall of the housing it is recommended to specify a thermal contact on the outer walls of the air domain. An additional Robbin boundary condition can be added to consider forced convection around the housing walls.

Internal air should be treated in a similar manner.

2.6 Aggregating components

Q-Bat Explorer gives the opportunity to aggregate components, in order to form integral battery parts. These assemblies can be easily rotated, transformed or duplicated. The supported aggregating types include cell caskets, battery modules, battery islands and batteries.

These components require the User to append different basic types of components. On this level it is also possible to add thermal couplings between the objects, as described in subsection 2.4. After creating an aggregated component and assigning contacts between its parts, it can be aggregated to a higher form, up to a battery.

The supported aggregating types include:

- A cell casket is an object that consists of cells and passive heat components.
- A battery module is an object that consists of cells caskets, passive heat components and cooling plates.
- A battery island is an object that consists of battery modules, passive heat components and cooling plates.
- A battery is an object that consists of battery islands and passive heat components.

Batteries are the highest form of aggregating objects, although they can be coupled with one another hydraulically.

A simple hierarchy of all the components and elements that can be aggregated within them is shown schematically in the figure below.

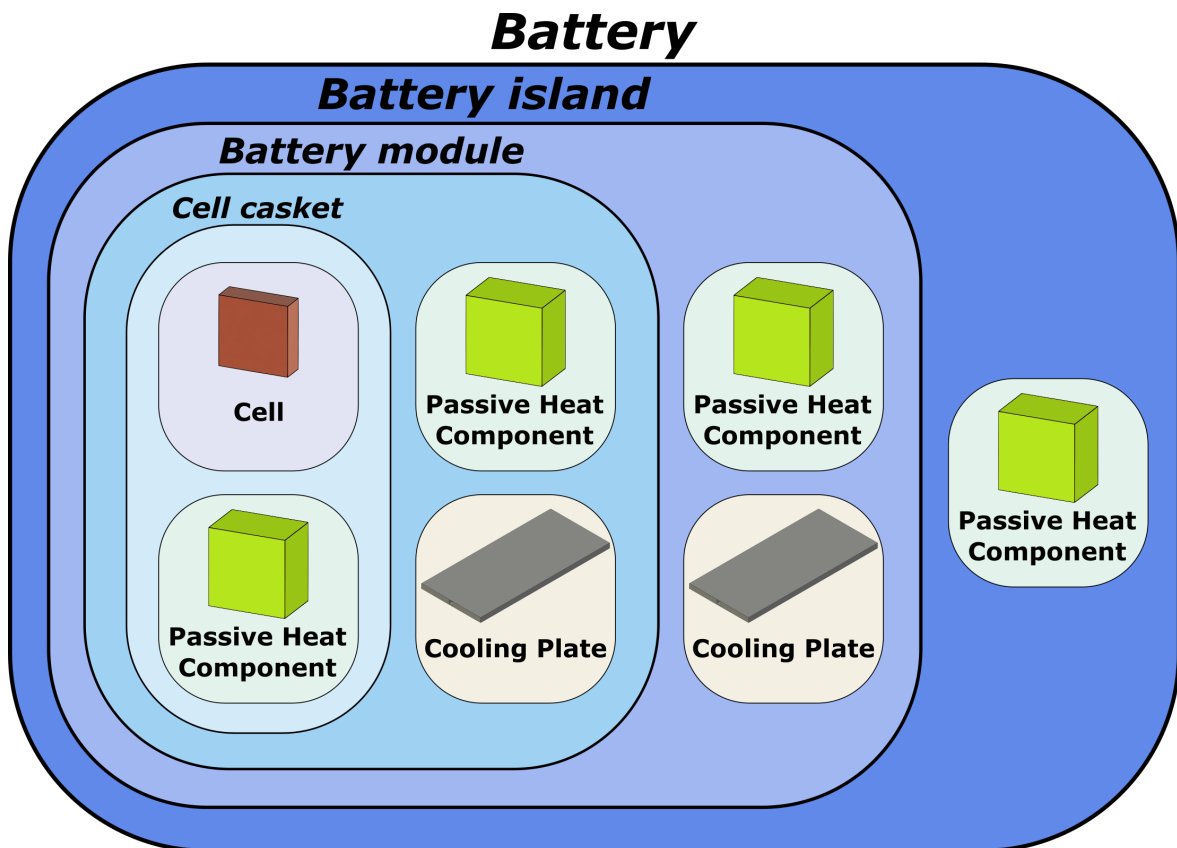


Figure 11: Aggregating components

Comment: Each of the components shown above can but does not have to consist of the shown elements. It can also include multiple objects of the same type. For example a battery module can consist of three different cell caskets, two identical cooling plates and no passive heat components.

3. Discretization

As the software uses FEM, discretization of the computational domain is crucial. Q-Bat Explorer reduces the basis of the solution to simplify the model and therefore substantially shortens the computational time. This section briefly describes the methods of discretization used in the software.

3.1 Mesh Generation & FEM

Q-Bat Explorer uses Gmsh software [1] to generate Finite Element meshes. Mesh creation is done automatically, with just the mesh density as the User input. For more complicated geometries, it is possible to create a custom mesh using the Gmsh software and import it directly into Q-Bat Explorer. This approach can help the user create a dense mesh only near the regions of interest, where most of the heat transfer occurs. When importing a msh file all surfaces must have physical IDs.

The automatic mesh creation takes in a parameter between 0-1, specifying the mesh density. Values close to 1 correspond to a poor quality mesh, while those closer to 0 correspond to a denser mesh (what also significantly increases the reduction time).

EXAMPLE

Meshes created automatically in Q-Bat Explorer for different mesh densities for a simple cuboid geometry.

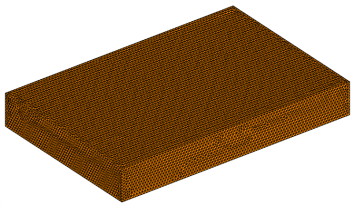


Figure 12: Mesh - 0.1

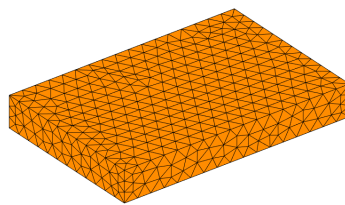


Figure 13: Mesh - 0.5

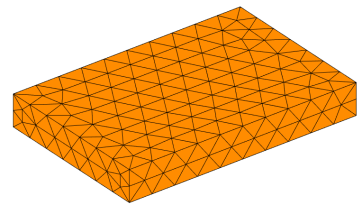


Figure 14: Mesh - 0.9

When creating a cooling plate it is also possible to specify the number of points on curvature. This parameter allows the create a denser mesh only in the proximity of the cooling pipe (if it is circular).

EXAMPLE

Cooling plate meshes created automatically in Q-Bat Explorer for different number of curvature points.

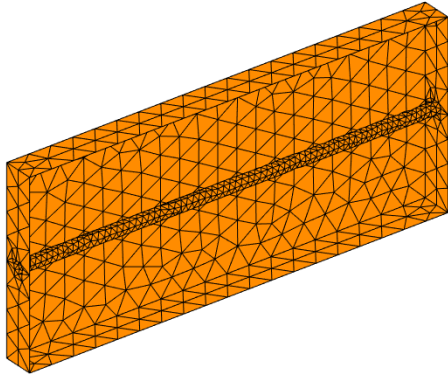


Figure 15: Mesh - 3 points on curvature

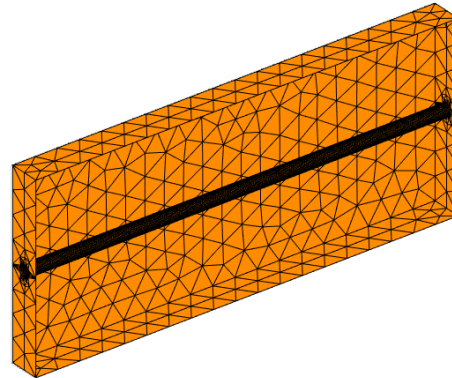


Figure 16: Mesh - 10 points on curvature

On basis of the created meshes, the heat transfer equations are solved by means of Finite Element Method. The Galerkin approach is utilized for calculating the diffusion matrix.

For each component a local mass and diffusion matrix is created. The matrices are coupled together depending on occurring contacts. Global matrices for the whole model are assembled and a linear equation system is solved. The Euler method is used for calculating the solution.

3.2 Model Reduction

For each body, the thermal problem is given by the equation:

$$\mathbf{M} * \frac{d\mathbf{T}}{dt} + \mathbf{D} * \mathbf{T} = 0 \quad (15)$$

Where M is the mass matrix, D is the diffusion matrix and T is the solution vector. Assuming a linear problem, the solution can be approximated as:

$$\mathbf{T}(\vec{x}, t) = e^{-\lambda * t} * \Phi(\vec{x}) \quad (16)$$

Bringing the problem to an eigenvalue equation:

$$-\lambda * \mathbf{M} * e^{-\lambda * t} * \Phi(\vec{x}) + \mathbf{D} * e^{-\lambda * t} * \Phi(\vec{x}) = 0 \quad (17)$$

$$-\lambda + \mathbf{D} * \Phi(\vec{x}) = 0 \quad (18)$$

Where $\Phi(\vec{x})$ is the eigenvector.

The obtained eigenvectors are used to create a basis of the solution. A reduction transformation matrix is formed:

$$\mathbf{L} = [\Phi_1, \Phi_2, \dots, \Phi_n] \quad (19)$$

A low-dimensional representation of the original space is created by forming a reduced mass and diffusion matrix:

$$\mathbf{K}_r = \mathbf{L}' * \mathbf{K} * \mathbf{L} \quad (20)$$

$$\mathbf{M}_r = \mathbf{L}' * \mathbf{M} * \mathbf{L} \quad (21)$$

The software uses the reduced matrices M_r and K_r to find the solution to the problem, which is later transformed to the full model.

The determined values of eigenvectors are regarded as modes, used to represent the basis of the solution. A visualization of a few of the modes generated for each object is depicted in the example below.

EXAMPLE

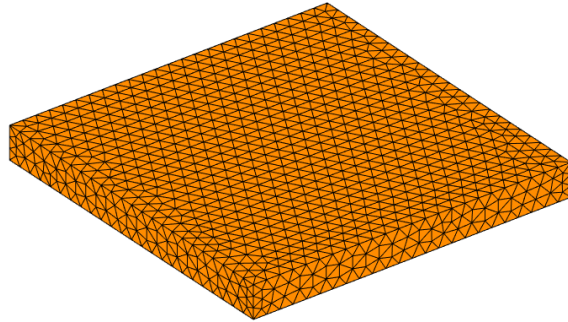


Figure 17: Sample mode generation - geometry

Some of the generated modes are displayed below to illustrate the concept of a reduced solution basis.

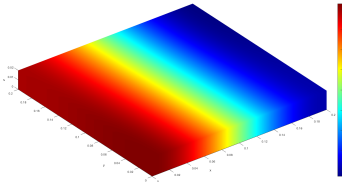


Figure 18: Sample mode generation - mode 2

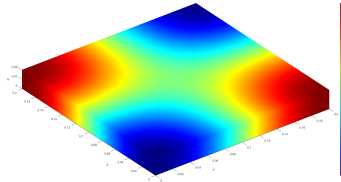


Figure 19: Sample mode generation - mode 3

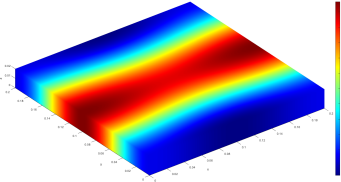


Figure 20: Sample mode generation - mode 5

3.3 Cooling fluid channel discretization

The cooling fluid is discretized by the User by defining the number of divisions (n) during cooling plate creation. Control point inside the fluid are created and the fluid is divided into $n - 1$ volumes.

The flow in the pipe is calculated using a second order Backward Differentiation Formula (BDF) in the discretized volume centroids, described by the following equations. The fluid is coupled with the surrounding cooling pipe in every discretized volume by the wall temperature.

$$\rho_i * \dot{e}_i * V_i = A_i * u_i * e_{i-1} - A_{i+1} * u_{i+1} * e_i + \alpha_i * S_i * (T_{w_i} - T(e_i)) \quad (22)$$

$$\dot{m}_{in} = u_i * \rho_i * A_i = u_{i+1} * \rho_{i+1} * A_{i+1} \quad (23)$$

Where:

ρ_i - density in the i -th volume [$\frac{kg}{m^3}$].

e_i - energy in the i -th volume [J].

V_i - i -th volume [m^3].

A_i - cross-section area before the i -th volume [m^2].

α_i - heat transfer coefficient in the i -th volume [$\frac{W}{m^2 * K}$].

S_i - side area (pipe area) of the i -th volume [m^2].

T_{w_i} - wall temperature in the i -th volume [K].

\dot{m}_{in} - mass flow rate [$\frac{kg}{s}$].

References

- [1] URL: <https://gmsh.info/>.
- [2] Wiesław Gogol. *Wymiana ciepła - Tablice i wykresy*. s.l. : Wydawnictwa Politechniki Warszawskiej, 1972.
- [3] Tomasz S. Wisniewski Stefan Wisniewski. *Wymiana ciepła*.