BoreSimSU

A fast C++ simulation program for single U-tube Borehole Heat Exchangers

Zanchini F, Zanchini E, Naldi C

BoreSimSU is a C++ program that can simulate the thermal response of any single U-tube Borehole Heat Exchanger (BHE) in the short and medium term. It is based on the model illustrated in Ref. [1], which combines the speed of Thermal Resistance Capacity Models (TRCMs) and the accuracy of finite-element simulations. The program yields, in a couple of seconds, the same results as those obtained by an accurate finite-element simulation requiring several hours of computation time. It provides the time evolution of the inlet, outlet and mean fluid temperature, of the mean BHE surface temperature, of the 3D and the effective borehole thermal resistance, and it can be easily connected to long-term simulation tools to obtain the full-time-scale thermal response of a bore field.

Description of the program

In the short and medium term, either 3D finite-element simulations or TRCMs can be used to evaluate the thermal response of BHEs. The former can yield very accurate results but require long computation times. The latter are much faster but cannot be fully precise, because they require simplifying assumptions. BoreSimSU combines the best of both: it employs a TRCM to quickly estimate the thermal response of the BHE, then corrects the results by interpolation with a dataset, which has been produced by running 54 finite-element simulations in COMSOL Multiphysics, in various configurations.

The TRCM implemented in the program is described in Section 3 of Ref. [1]. The BHE and the ground are divided into horizontal layers of the same height. The ground is also divided into concentric annuli, of increasing thickness. Each horizontal layer is modeled by a thermal circuit, based on the standard triangular network of thermal resistances [2], to which additional nodes are added. In particular, two nodes are added to model the pipes, and two intermediate grout nodes are introduced between the pipe nodes and the borehole node, to better describe the interior of the BHE. The thermal resistances R_1^{Δ} and R_{12}^{Δ} of the triangular network are determined from the borehole thermal resistance, R_b, and the total thermal resistance between the flows, R_a. The thermal resistances R_a and R_b are computed through the first-order multipole expressions given by Javes and Spitler [3]. Since heat conduction in the vertical direction is negligible, energy transfer in the vertical direction is assumed to take place only through the fluid flow. The energy balance equations for the fluid nodes, the pipe nodes, the grout nodes, the borehole node and the ground nodes form a linear system, which is solved by Gaussian elimination at each time instant.

The TRCM slightly underestimates the inlet and outlet fluid temperatures, T_in and T_out, the mean fluid temperature, T_fm, the 3D borehole thermal resistance, R_b3D, and the effective borehole thermal resistance, R_beff, in the medium term. On the other hand, it yields an accurate time evolution of the mean surface temperature of the BHE, T_b, and of the difference $T_\mathrm{fm} - T_\mathrm{fave}$, where T_fave is the arithmetic mean of the inlet and outlet temperatures.

The inaccuracies of the TRCM are corrected a posteriori through a time-dependent correction coefficient, c_coeff, defined as the ratio $R_\mathrm{b3D,COMSOL}/R_\mathrm{b3D,TRCM}$. The coefficient allows to correct not only the values of R_b3D, but also the values of T_fm, T_fave, T_in, T_out and R_beff, by simple mathematical relations. Only four parameters have a relevant effect on the correction coefficient: the borehole radius, r_b, the shank spacing, s, the thermal conductivity of the grout, k_gt, and the outer radius of the pipe, r_pe. The values of c_coeff have been determined by running pairs of simulations with the TRCM and with COMSOL, in 54 different configurations: for two values of r_pe, namely 20 and 16 mm, three values of r_b, namely 68, 72 and 76 mm, three values of s, namely 27, 37 and 47 mm, and three

values of k_{gt}, namely 1.0, 1.6 and 2.2 $\mathrm{W/(m\,K)}$. In all other configurations, the values of c_{coeff} are estimated by the program through polynomial interpolations.
If the user wishes, the correction of the TRCM can be disabled from the input file.

Instructions for users

BoreSimSU can be run on any Linux PC, as it only requires the standard GCC compiler for C++ (no external libraries). It was written for C++17, compiled with GCC 13.3.0, ant tested on a PC with Ubuntu 24.04 LTS (Noble Numbat).
To use BoreSimSU, download the homonymous program folder, unzip it, and move it to your Desktop. Inside the folder, you will find the main file of the program, BoreSimSU.cpp, together with three files:

- input_data.txt
- correction_factors.txt
- Rb3D_and_correction_coeff.xlsx

The first file contains the parameters that need to be set to run the simulation. The second one contains the 54 time-dependent correction factors used by the program. The third is an excel file which contains the same 54 time-dependent correction factors in a more readable form, together with the time-dependent values of R_{b3D} determined by COMSOL in the 54 configurations mentioned above.

Setting the input data

To set the input data, open the file input_data.txt. The file requires 26 parameters.
The program considers equally spaced values of $\log_{10}(t_{\mathrm{hours}})$, where t_{hours} is time in hours. The first three parameters describe the time mesh used by the program for the simulation:

- the base ten logarithm of the initial instant
- the base ten logarithm of the final instant
- the logarithmic time step, namely $\Delta\log_{10}(t_{\mathrm{hours}})$

For the first parameter, the minimum allowed value is -2.5, which corresponds, approximately, to 11.4 seconds. For the second parameter, the maximum allowed value is 3, which corresponds to 1000 hours. For the third parameter, the default value is 0.1. This value can be changed with no restrictions, although inaccurate results may be obtained if the chosen value is too large, while the program may struggle to run if the chosen value is too small.
The next six parameters are some general characteristics of the BHE:

- the length of the BHE
- the radius of the BHE
- the half shank spacing
- the total heat rate supplied to the BHE
- the volume flow rate
- the number of horizontal layers in which the BHE should be divided

The latter has a default value of 100, which is large enough in most cases, although higher values may be needed if a short time step is selected.
The next four parameters are the thermal properties of the fluid, namely:

- the thermal conductivity
- the density
- the specific heat capacity at constant pressure
- the dynamic viscosity

The default values are the thermal properties of water at 20° C, taken from the NIST Chemistry WebBook [4].
The next four parameters are the properties of the pipes, namely:

- the thermal conductivity
- the volumetric heat capacity
- the inner radius
- the outer radius

The next two parameters are the properties of the grout, namely:

- the thermal conductivity
- the volumetric heat capacity

The next five parameters concern the ground, and they are:

- the thermal conductivity
- the volumetric heat capacity
- the thickness of the first annulus
- the thickness increase factor
- the end of ground

The thickness of the first ground annulus is set, by default, to 1 cm. As for the thickness increase factor, its default value is 1.25. This means that each subsequent annulus is 25% thicker than the previous one. The program keeps adding ground annuli until the distance from the BHE axis becomes greater or equal to the value set as the "end of ground".

Finally, the last two parameters are:

- the implementation of the correction factor
- the number of decimal digits that will be printed in the output file

The implementation of the correction factor is a Boolean parameter. Its default value is 1, which means that the results of the TRCM are corrected a posteriori, as previously explained. If the value is set to 0, no correction is performed. The number of decimal digits to print in the output file can be chosen arbitrarily.

Compiling and running

If you have already set the parameters for the simulation (and you have saved the changes), you are ready to compile and run the program.
Open a terminal and type the following command line to access the program folder:

```
cd Desktop/BoreSimSU
```

Next, type the following command line to compile:

```
g++ -O4 -o BoreSimSU.out BoreSimSU.cpp
```

(you will need to compile the program only once).
Finally, type the following command line to run the program:

```
./BoreSimSU.out
```

The program will run on its own, and will print some info on the terminal. The computation time depends on many factors, but, normally, it should not exceed a few seconds.

Checking the results

At the end of the execution, the program creates a file named output_data.txt with the results of the simulation, containing the time evolution of T_{in}, T_{out}, T_{fave}, T_{fm}, T_{b}, R_{beff} and R_{b3D}.
The output file is overwritten every time the program is executed. If the file was already open before the program execution, you may need to refresh it to view the latest results.

References

1. E. Zanchini, F. Zanchini, C. Naldi, A combined thermal-resistance-capacity and finite-element model for very fast and accurate short- and medium-term simulations of single U-tube borehole heat exchangers, arXiv:2510.14421 [physics.app-ph]. https://doi.org/10.48550/arXiv.2510.14421
2. G. Hellström, Ground heat storage: Thermal analyses of duct storage systems—Theory. Doctoral Thesis, University of Lund, Lund, Sweden, 1991.
3. S. Javed, J. Spitler, Accuracy of borehole thermal resistance calculation methods for grouted single U-tube ground heat exchangers. Applied Energy 187 (2017) 790–806. http://dx.doi.org/10.1016/j.apenergy.2016.11.079
4. NIST Chemistry WebBook. 2025. Online: https://webbook.nist.gov/chemistry/fluid/

License