This is a preprint; the final reference is here:

Spitler, J. D., J. C. Cook and X. Liu (2020). "Recent Experiences Calculating g-functions for Use in Simulation of Ground Heat Exchangers." Geothermal Resources Council Transactions **44**: 296-315.

Recent Experiences Calculating g-functions for Use in Simulation of Ground Heat Exchangers

Jeffrey Spitler[a], Jack Cook[a], Xiaobing Liu[b,1]

[a]Oklahoma State University, [b]Oak Ridge National Laboratory

Keywords

ground source heat pump, vertical bore ground heat exchanger, modeling, g-function

ABSTRACT

Temperature response functions, known as g-functions, are a computationally efficient method for simulating ground heat exchangers (GHEs), used with ground-source heat pump (GSHP) systems, either as part of a whole-building energy simulation or as part of a dedicated ground heat exchanger design tool. In fact, at present, they are the only feasible way to simulate a ground-source heat pump system in a whole-building energy simulation. This paper summarizes recent developments in the field and recent experience using a new open-source g-function calculation tool known as pygfunction. This experience includes accuracy, computation time, memory requirements and sensitivity to boundary conditions. With larger ground heat exchangers, e.g. in excess of 100 boreholes, the computational time and memory requirements can create challenges.

1. Introduction and Background

Temperature response functions, known as g-functions, are a computationally efficient method for simulating ground heat exchangers (GHEs) used with ground-source heat pump (GSHP) systems, either as part of a whole-building energy simulation or as part of a dedicated ground heat exchanger design tool. In fact, at present, they are the only feasible way to simulate a ground-source heat pump system in a whole-building energy simulation. This speed is possible because the most

---

computationally intensive portion of the analysis is done before the simulation – that is, the g-function is calculated first. At present, both design tools and energy simulation tools rely mainly on libraries of g-functions for specific borehole configurations. By "configurations", we mean geometric arrangements of boreholes with specific ratios of spacing to depth. For example, 100 boreholes in a 10x10 grid with a horizontal spacing to depth ratio of 0.1. The boreholes could be vertical or angled, regularly-placed or irregularly-placed, uniform depth or non-uniform depth.

Libraries of g-functions are quite useful, but not sufficient for many designs. Therefore, it is highly desirable to have methods that do not rely solely on libraries. For purposes of automating design and automatic optimization of design, where hundreds of configurations might need to be evaluated quickly, faster procedures are really needed.

One complicating but subtle feature of g-functions is that the thermal response has a significant dependence on how the heat rejected or extracted is distributed through the borefield. G-functions are computed with a fixed heat rejection rate, but how the heat is distributed through the borefield will vary with time. One can think of the situation for a large rectangular borefield with continuous heat rejection over many years. In this case, the inner boreholes get "saturated' with heat and hence reject less heat over time. There are three approximations that have been used:

1. Uniform inlet fluid temperature (UIFT). Here, all of the boreholes receive fluid at the same temperature. The actual distribution is then calculated as part of the calculation of the g-function.
2. Uniform borehole wall temperature (UBHWT). With this approximation, the borehole wall temperatures have a time-varying but uniform temperature (i.e., same borehole wall temperature for all boreholes at any given time).
3. Uniform heat flux (UHF). With this approximation, the heat input is uniformly distributed and all boreholes have the same heat flux (i.e., the total heat input used to calculate the g-function is divided by the total borehole length).

Arguably, the UIFT approximation is the closest match to reality. The heat transfer fluid is generally returned to the GHE in a single pipe, which is then delivered to each borehole in parallel. Other than differences in the length of horizontal piping having a small effect on the delivery temperature to each borehole, the inlet fluid temperature should be uniform. As will be discussed below, the UBHWT approximation gives similar results to the UIFT approximation. Malayappan and Spitler (2013) investigated the UHF approximation and found that, while it works well for small numbers of boreholes (i.e., less than 30 boreholes), it can give significant sizing errors for large borefields where significant thermal interference between boreholes is present.

The concept of using thermal response functions, known as g-functions, was introduced by Claesson and Eskilson (1985). More recently, several promising approaches have been developed. These include:

- Use of an analytical method (finite line-source) to compute g-functions. Lazzarotto (2015) and Cimmino (2018a) have made a number of improvements that speed the calculations. Cimmino's approach is available in an open-source Python library, pygfunction.
- Use of a block matrix formulation (Dusseault et al., 2018) to calculate g-functions quickly. The block matrix formulation is similar in some respects to Cimmino (2018), but the

numerical integration of the error function is replaced with Chebyshev polynomials. With this approach, the authors have been able to compute g-functions for 50-borehole GHEs over a 40-year period in less than a half-second on a notebook computer.

- Use of artificial neural networks (ANN) to calculate g-functions. (Dusseault and Pasquier, 2018; Pasquier et al., 2018; Dusseault and Pasquier, 2019) In this approach, a library of g-functions is used to provide training data for the ANN. Preliminary testing for small borehole fields (with less than 10 boreholes) had been very promising.

G-functions can cover a range of times. At very short times (minutes to hours), factors such as fluid transit time, heat transfer within the borehole, and heat transfer in the ground immediately surrounding the borehole affect the response. At longer times, borehole-to-borehole interference and end effects (i.e., heat transfer to/from the ground surface and the ground formation beneath the boreholes) become more important. This paper focuses on "long-time-step" g-functions, applying, at times, greater than about two days. For more information on short-time-step g-functions, the reader is referred to work by Mitchell (2019), Brussieux and Bernier (2019), Pasquier et al. (2018), Beier et al. (2018), Beier and Spitler (2016), Xu and Spitler (2006) and Yavuzturk and Spitler (1999).

2. Literature Review

Methods for computing long time step g-functions were introduced by Prof. Johan Claesson of the University of Lund and his graduate student, Per Eskilson, in the 1980s. For many years, these methods were the only ones available. However, in the last five years, there has been a renewed interest in methods for calculating g-functions.

2.1 Superposition Borehole Model

The concept of using thermal response functions, known as g-functions was introduced by Claesson and Eskilson (1985). The methodology for computing g-functions was described by Eskilson and Claesson (1988). A 3600-line FORTRAN 77 implementation is described in the paper. A more detailed explanation is given in a 95-page unpublished report by Eskilson (1986).

The solution procedure makes use of a detailed radial-axial numerical simulation of the ground surrounding a single borehole coupled to a detailed analytical model of the heat transfer inside the borehole. This individual solution is then superimposed spatially to give the response of an entire borefield. The detailed analytical model of the heat transfer in the borehole allows the model to cover all the boundary conditions (UFIT, UBHWT, UHF) discussed above.

It should be noted here that this code is intended for a range of simulations, but that it could be, and has been, adapted to calculate g-functions. This only requires a constant heat input; from the resulting outlet temperature the g-function can be calculated. However, as a practical matter, the code is not publicly available.

2.1.1 Cimmino

Cimmino (2019b) gives the latest development in a series (Cimmino and Bernier, 2013; Cimmino, 2014; Cimmino and Bernier, 2014; Cimmino, 2015; Cimmino and Eslami-Nejad, 2017; Cimmino, 2018a) of papers on calculation of g-functions. Cimmino's methodology is based on a finite-line-

source simulation. By dividing each borehole into multiple segments, the UBHWT and UFIT boundary conditions can be approximated. This requires a rather complex method for determining the required time-varying heat flux on each segment. Cimmino's method identifies the symmetries within the borehole field and computes segment-to-segment response factors for each unique pair of segments. The results for the overall borehole field can then be calculated at the same time the time-varying heat flux on each segment is calculated.

The approach of dividing the borehole into multiple segments, each of which forms a finite line source, is known as the "stacked finite line source method" (Cimmino and Bernier, 2014) The use of "stacked" is helpful to distinguish it from the finite line source (FLS) with the UHF approximation, where all boreholes have the same heat flux, so each borehole is treated as a single finite line source.

A major advantage to this method is that Cimmino has released an open-source version (Cimmino, 2018b and 2019a), known as pygfunction. We expect that coding (from scratch) any g-function calculation method that can calculate UFIT and UBHWT boundary conditions would likely take several months of graduate student time, at best. The availability of open-source code is highly beneficial.

2.2 Pasquier – SBMF

Prof. Philippe Pasquier, his graduate student Bernard Dusseault, and colleague Prof. Denis Marcotte, have made several contributions to the calculation of GHE response. Dusseault et al. (2018) introduced a structured block matrix formulation (SBMF) for the FLS method. The method is capable of calculating g-functions for UBHWT boundary conditions, but they have implemented it for individual boreholes rather than multiple segments per borehole. (That is, it uses FLS, not stacked FLS.) It may be possible to adapt it to use multiple segments per borehole. Or, it may be that this approximation is sufficiently accurate for fast automated sizing.

The method has several features of interest. For cases with regular borehole spacing and therefore high degrees of symmetry, the borehole-to-borehole response can be represented with Chebyshev polynomials, which speeds the calculation. The resulting computation time for a 50-borehole, regularly-spaced borefield is quite fast – less than a half second on a desktop computer. However, it is difficult to make a direct timing comparison with Cimmino's method. The SBMF method is only working with 50 segments. Cimmino's method, with 66 segments, takes about 9 seconds on our cluster. However, the use of one segment per borehole will not give adequate accuracy over the entire range of configurations of interest. It may be possible to extend the SBMF method to handle multiple segments/borehole, but we have not attempted to do so.

2.3 Lazorotto

Lazzarotto (2016) and Lazzarotto and Björk (2016) describe some aspects of a methodology for computing g-functions with the finite line source method. The emphasis is on improvements to the analytical solution and application to cases where the boreholes are inclined from vertical. Only small borehole fields (<10 boreholes) are considered in the paper, so it is difficult to determine its applicability to larger borehole fields. Timing results are reported for several permutations of implementing the code in Julia, Python, and Python with Cython. However, the borehole configuration is not reported, so it is impossible to compare with other methods. The availability of the source code is unknown.

2.4 Monzó

Monzó et al. (2015) reported on use of a numerical model to calculate g-functions. Although there are some interesting aspects to the work, the proposed method cannot quickly calculate g-functions.

2.5 Cullin – Equation fit

Cullin (2008) described an equation fit for large rectangular borefields. A six-term polynomial gives g-function values as a function of two ratios: the number of corner to interior boreholes and the number of corner to perimeter boreholes. Six coefficients are stored for each value of $\ln\left(\frac{t}{t_s}\right)$ and for multiple *B/H* ratios. Computational time is not reported, but it is very fast even at typical personal computers. In order to pursue such an approach, data sets containing g-functions for a variety of configurations would be needed.

2.6 Pasquier – ANN

Dusseault and Pasquier (2019) present an interesting possibility for fast calculation of g-function – the use of artificial neural networks (ANN). The use of ANN requires a large training set. The ANN developed by Dusseault and Pasquier utilized a training set of 500,000 g-functions computed for random arrangements of 1-10 boreholes within a 30mx30m rectangular area. Pasquier (2019) suggested that a training data set of a million g-functions would be necessary to create an ANN that could calculate g-functions for a wide variety of designs. Once the ANN is created, g-functions can be calculated in a fraction of a second. The first challenge would be to develop a database of a million g-functions that covered the range of configurations of interest.

2.7 Discussion of long time step g-function calculations

Of the methods discussed above, the first five calculate g-functions based on fundamental physics. This is necessarily time-consuming. The last two methods rely on data sets generated from fundamental physics. But once the data sets are produced using one of the first five methods, the last two methods can give very fast answers.

3. Software Development

The pygfunction code, as provided by Cimmino, was set up so that the borefield specifications were set within the code itself. While this is likely satisfactory for a single user generating only a

few g-functions, we have been creating large numbers of long time step (LTS) g-functions and have, therefore, recast the code.

The code to generate large numbers of LTS g-functions works in two steps. The first step is to define calculation configurations (i.e., positions of each borehole in the field) in a spreadsheet. To create, for example, all configurations between 2x2 and 32x32, a short Excel VBA code was used to calculate the borehole positions and, for each configuration, write those to a single worksheet in the Excel workbook. Scripts read the borehole position information on each worksheet and use that information to organize the g-function calculations on one of the clusters at Oklahoma State University.

The second step of the g-function generator takes four arguments. The first three arguments are the path locations of the bore field geometry, input parameters (grout, ground, g-function calculation time, etc.), and the annual profile of the thermal loads (if sizing will also be performed). The last argument selects a boundary condition (UBHWT, UIFT, or UHF).

The initial work was done on a desktop computer. For several reasons, including high memory requirements and the desire to calculate large numbers of g-functions, we transitioned to working on two computing clusters at Oklahoma State University, named "Cowboy" and "Pete." Nodes that are utilized for calculating g-functions are:

- Cowboy has 252 nodes with dual Intel Xeon E5-2620 "Sandy Bridge" hex core 2.0 GHz CPUs and 32 GB RAM.
- The two big memory nodes on Cowboy, each with dual Intel Xeon E5-2620 "Sandy Bridge" CPUs and 256 GB RAM.
- Pete has 164 nodes with dual Intel "Skylake" 6130 CPUs and 96 GB RAM.
- The twelve large memory nodes on Pete, each with dual Intel "Skylake" 6130 CPUs and 768GB RAM.

Further details of the clusters are available at the OSU High Performance Computing Center website (OSU HPCC, 2019). The g-function calculations for large borefields conducted in this study often require the high-memory (256GB or 768GB RAM) system nodes..

4. Results – Computational Requirements and Verification

Based on the discussion above, our testing reported here has been of the Cimmino method, with an eye towards generating simultaneously a large library of long-time-step g-functions and, possibly, a training set for an artificial neural network.

4.1 Timing and memory requirements

Important aspects of the computation of g-functions include the time and memory required for computation of a single g-function. The computational time requirements are important, whether the method is going to be used to compute a custom g-function for a one-off application or to be used to compute a large library of g-functions. Memory requirements are important because:

- Many desktop computers do not have sufficient RAM to compute g-functions for larger borehole configurations.

- Even high-power desktop computers ("workstations") may encounter limits that preclude calculating g-functions with sufficient accuracy.
- The computing clusters have far more nodes with smaller amounts of RAM; knowing the required memory lets us more effectively allocate the available resources.

The stacked finite line source method used in pygfunction allows the discretization of each borehole into segments. We've found that the computing time and memory requirements scale roughly to the total number of segments. Some variations are due to the amount of symmetry in the borehole field. The configurations used here vary in the amount of symmetry. Most of the configurations are uniformly-spaced rectangular configurations.

Required computation times for a range of configurations are shown in Figure 1, along with a 2nd order polynomial curve fit. While actual computation times will depend on the computer and processor, this gives an estimate for a high performance CPU, and can be used to yield an estimate of how long it might take to calculate a library of g-functions.

This also gives some idea of the problems associated with calculating custom g-functions 'on the fly' as part of a design process. Once the configuration of boreholes has been set, it may take on the order of 10 iterations to determine the required depth. In cases with large numbers of boreholes and segments, it is not practical to compute 10 g-functions while the user waits. An alternative approach is to compute a set of g-functions for different pre-defined depths, then calculate g-functions for specific depths using interpolation. One scheme is described in Section 4.5.

Memory requirements may also vary with the installation. We have not done a detailed analysis of the memory usage, but offer two data points:

- When calculating g-functions for a 1024 borehole case with 6 segments/borehole (6144 segments) on a desktop workstation, memory requirements for the process maxed out at 87 GB of RAM or 14 GB/1000 segments. In this cases, other processes on the computer were also consuming RAM.
- When computing g-functions for a range of rectangular configurations on the big memory nodes of the Pete cluster, with 256 GB, the maximum number of segments that can be run without failing is 24,192 segments. A case with 24,800 segments cannot be run, so the absolute limit should fall in between. Using 256 GB to calculate 24,192 segments corresponds to 10.6 GB/1000 segments. This number is presumably lower because other processes on the computer also consume RAM.

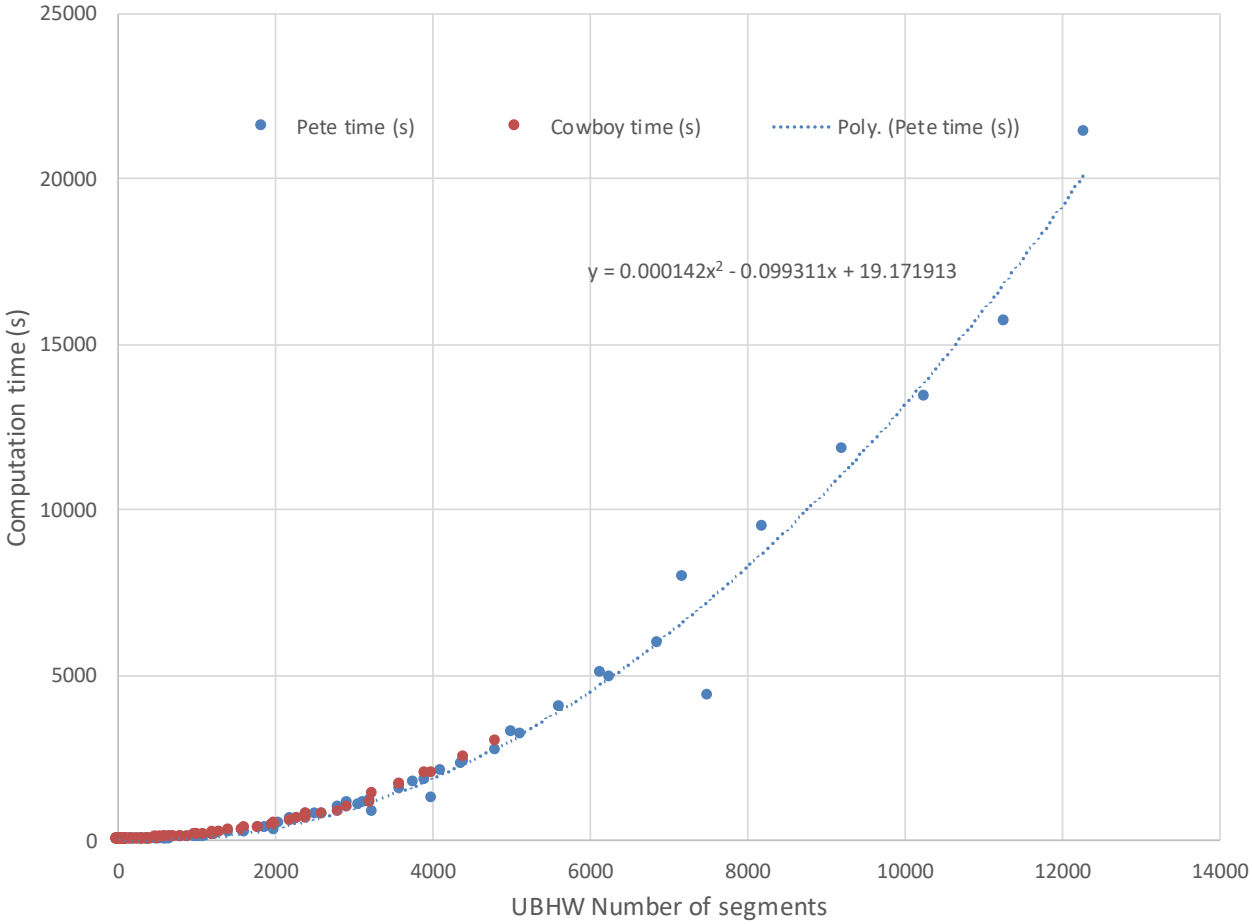

Figure 1: Computation time on the Pete cluster versus the number of segments in a borehole field.

4.2 Sensitivity to Boundary Conditions

As discussed earlier, there are three commonly used boundary conditions: UBHWT, UIFT, and UHF. For a 100-borehole rectangular configuration with 5 m borehole spacing and 100 m borehole depth, g-functions have been computed for all three boundary conditions. Figure 2 shows a comparison of the g-functions. It took 176, 197, and 4.4 seconds, respectively, to compute the g-functions under boundary conditions of UBHWT, UIFT, and UHF. The GLHEPRO values were computed with the superposition borehole method using the UBHWT boundary conditions. As discussed earlier, UIFT is arguably the most accurate, and may be considered as the "reference g-function." At $\ln\left(\frac{t}{t_s}\right) = 0.0913$, corresponding to 30 years with dense rock and 43 years with saturated heavy soil, the g-function value resulting from UBHWT is 4.2% lower than that from UIFT and the g-function value resulting from UHF is 24% higher than that from the UIFT. Actual errors in sizing will be quite a bit lower, since the load on the GHE is not constant. Nevertheless, the UHF has the possibility to significantly oversize the GHE and so cannot be recommended. The UBHWT g-function is likely to give no more than a few percent undersizing, well within the uncertainties of ground properties and building loads.

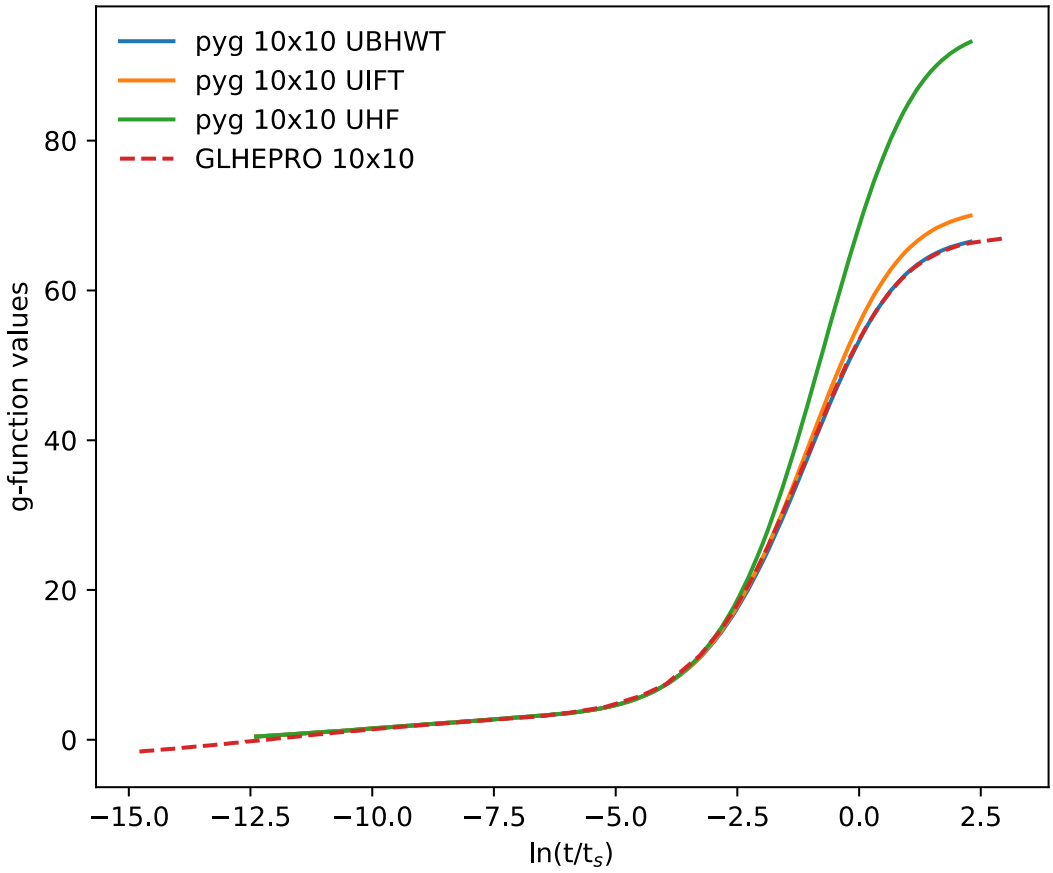

Figure 2: Comparison of g-functions for a 10x10 borefield computed with pygfunction under three different boundary conditions (UBWT, UIFT, and UHF).

It is important to note that the UIFT g-function depends on the system flow rate and borehole resistance. Parameters for an example comparison are given in Table 1. The highlighted rows indicate the parameters that were changed – the four cases represent combinations of low and high system flow rate, and low and high borehole thermal resistance (R_b). The different shank spacings are illustrated in Figure 3. The left-hand image with zero shank spacing corresponds to a high borehole thermal resistance. The right-hand image with 45 mm of shank spacing corresponds to a low borehole thermal resistance.

Table 1: Parameters for UIFT Sensitivity analysis.

Comment	High R_b, High flow	High R_b, Low flow	Low R_b, High flow	Low R_b, Low flow
Burial Depth (m)	4.5	4.5	4.5	4.5
borehole radius (m)	0.075	0.075	0.075	0.075
pipe outer radius (m)	0.0167	0.0167	0.0167	0.0167
pipe inner radius (m)	0.013665	0.013665	0.013665	0.013665
shank spacing (m)	0.017	0.017	0.045	0.045
pipe roughness (m)	2.00E-05	2.00E-05	2.00E-05	2.00E-05
U-tube	Single	Single	Single	Single
Ground thermal diffusivity (m2/s)	1.29E-06	1.29E-06	1.29E-06	1.29E-06
Ground thermal conductivity (W/m.K)	3.3	3.3	3.3	3.3
Undisturbed ground temperature (degC)	20.30	20.30	20.30	20.30
Grout thermal conductivity (W/m.K)	0.7	0.7	2.1	2.1
Pipe thermal conductivity (W/m.K)	0.42	0.42	0.42	0.42
Fluid mass flow rate per borehole(kg/s)	0.75	0.25	0.75	0.25
Fluid specific isobaric heat capacity (J/kg.K)	4173.80	4173.80	4173.80	4173.80
Fluid density (kg/m^3)	998.2	998.2	998.2	998.2
Fluid dynamic viscosity (kg/m.s)	0.001	0.001	0.001	0.001
Fluid thermal conductivity (W/m.K)	0.593	0.593	0.593	0.593
G function calculation years	250	250	250	250
Number of segments per borehole	12	12	12	12
Height (m)	100	100	100	100

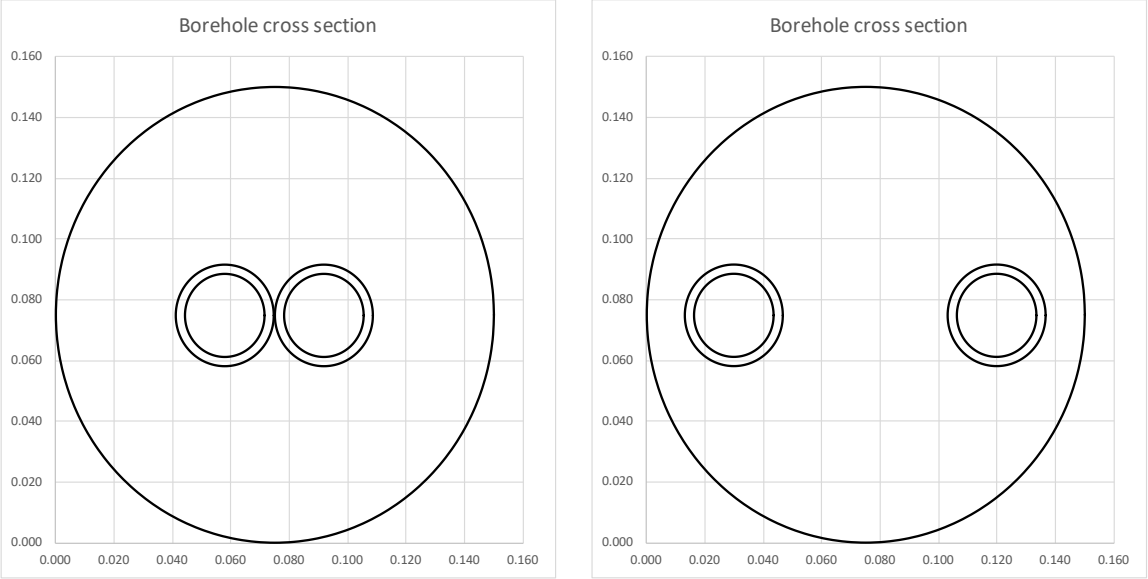

Figure 3: U-tube positions used for UIFT sensitivity analysis.

The results of the sensitivity analysis are shown in Figure 4. As can be seen there, within the range of flow rates shown, and for a 100 m deep borehole, the effect of the flow rate is fairly small. The effect of the borehole thermal resistance is about 5% at times corresponding to 45 years for dense rock or 65 years for heavily saturated soil.

For the work described here, we use the UIFT with a 'typical' flow and R_b value, in between the cases shown here, as a reference g-function. To serve as a reference g-function, it also needs to have sufficient segments. The required number of segments is discussed in Section 4.4. Different conditions may lead to ±2.5% variation in the long-term g-functions. As GHEs generally have lower design lifetimes than the 45 and 65 years shown in Figure 4, and as the annual variation in building heating/cooling loads leads to the response at shorter times being more important, we expect that the error associated with this approximation is somewhat less than ±2.5%.

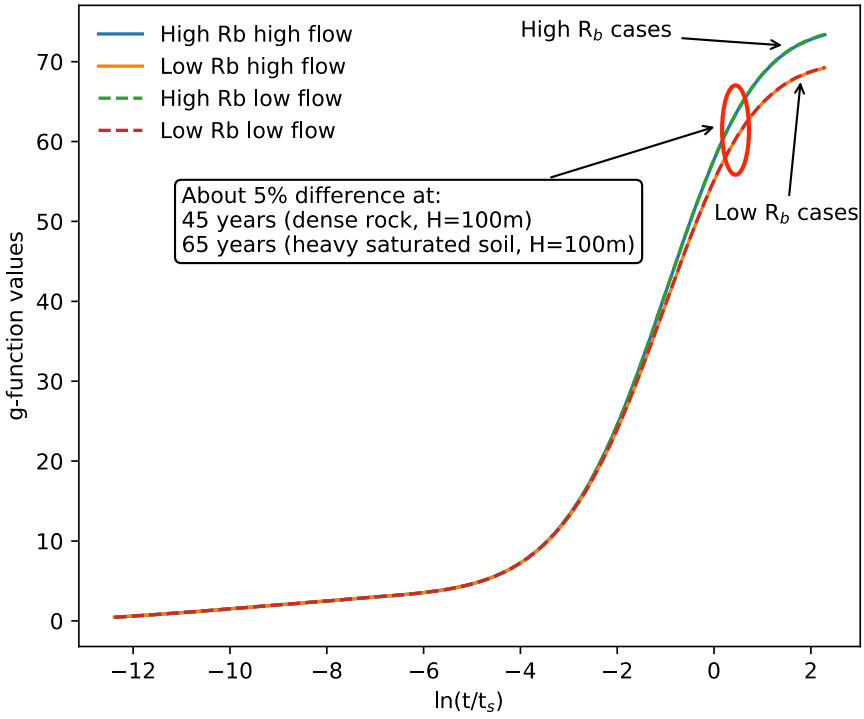

Figure 4: Effect of borehole parameters on g-functions calculated with UIFT boundary condition for a 100-borehole field.

4.3 Accuracy of Long-Time-Step g-functions

Fully validating the accuracy of g-functions with full-scale experimental measurements is difficult or, more likely, impossible. Difficulties include the time-scale, maintaining adequately accurate experimental measurements and fixed heat inputs over time, and locating the ground heat exchanger in a sufficiently homogeneous underground.

Some laboratory-scale measurements have been made, and validations of design methods using g-functions have been reported (Cullin et al., 2015), though these cases have at most only a few years of data and relatively balanced annual loads. When the loads are balanced, the effects of borehole-to-borehole interference are diminished, and any errors in the late-time g-functions are irrelevant.

Therefore, verifying the g-functions with one of two approaches is probably the best that can be done at present:

- The accuracy of the g-functions for borefields with regular spacing can be verified by comparison to the g-functions used by GLHEPRO or by g-functions provided by Eskilson (1987). This allows comparisons for rectangular fields of up to 20 x 20 boreholes.
- The number of segments required to reach a converged solution can be verified by using the equivalent to a grid-independency test used with finite difference and finite volume conduction heat transfer analyses. That is, the number of segments is sequentially increased until the change in the calculated g-function is negligible.

The accuracy of the g-functions for borefields with regular spacing can be verified by comparison to the g-functions used by GLHEPRO or by g-functions provided by Eskilson (1987). With the exception of the 200 borehole field, the long time-step portions of the g-functions [where $\ln\left(\frac{t}{t_s}\right) \geq -8.5$] come from the GLHEPRO library and were calculated with the superposition borehole model (Eskilson, 1986). The g-function for the 200 borehole field was computed using the equation fit developed by Cullin (2008) based on g-functions computed with the superposition borehole model provided by Hellström (2006).

Figures 5 and 6 show comparisons of the g-functions generated with pygfunction for cases with nine or fewer vertical boreholes, and 30-200 boreholes, respectively. The parameters used for these boreholes are shown in Table 2. The dashed lines represent the g-functions used by GLHEPRO. As discussed in Section 3, we are using only the long time-step (LTS) portion of the g-function generated by pygfunction. So, the deviation below $\ln\left(\frac{t}{t_s}\right) = -8.5$ may be ignored. As can be seen, the g-functions match quite closely. To quantify the deviation, $\ln\left(\frac{t}{t_s}\right) = 1$ corresponds to 74 years for a 100 m deep borehole in hard rock conditions and 107 years for saturated heavy soil conditions. At this point, the difference between the g-values for the 0009_BH case is about 0.1%. This implies that in the worst case of a completely imbalanced system with a constant load on the GHE, the prediction of long-term temperature rise or fall could be off by about 0.1% after 75 years. This error is much, much smaller than other errors in the design process and is, therefore, acceptable. Furthermore, slightly higher errors may also be acceptable for the sake of decreasing computation time.

It should be noted that these verifications utilize 12 segments and 100 m deep boreholes, suggesting that 12 segments/100m may be acceptable. This is examined further in the next section.

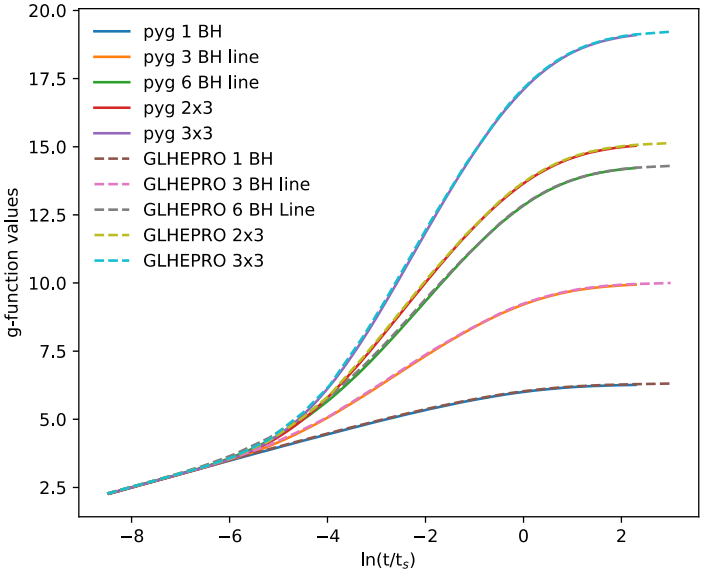

Figure 5: Comparisons between g-functions computed with pygfunction and GLHEPRO (superposition borehole method) for borefields with less than 10 vertical boreholes.

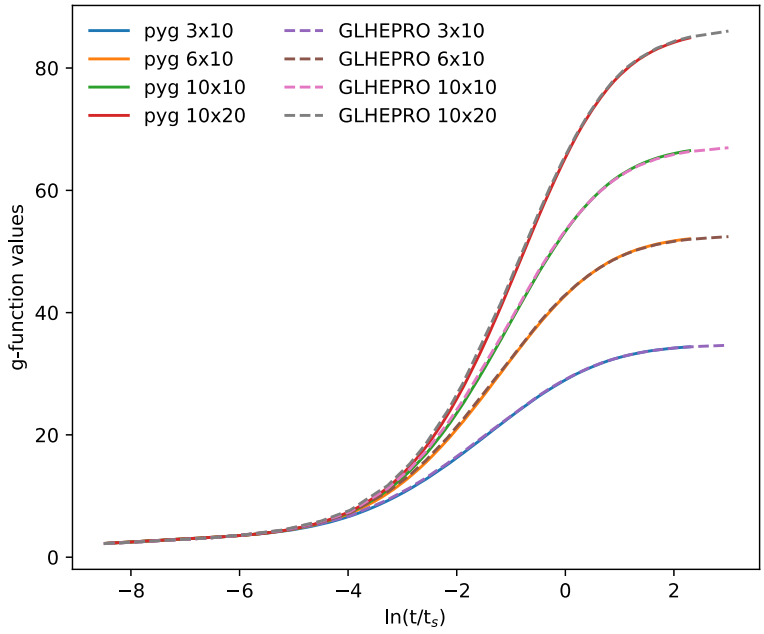

Figure 6: Comparisons between g-functions computed with pygfunction and GLHEPRO (superposition borehole method) for borefields with 30 to 200 vertical boreholes.

Table 2: g-function inputs for verification test cases.

Input	Value
Burial Depth (m)	4.5
Height (m)	100
Horizontal spacing (m)	5
Borehole radius (m)	0.075
Type of U-tube	Single U
Ground thermal diffusivity (m^2/s)	1.29E-06
Ground thermal conductivity (W/m·K)	3.3
Undisturbed ground temperature (°C)	20.30
Fluid mass flow rate per borehole (kg/s)	0.503
Fluid specific isobaric heat capacity (J/kg·K)	4173.8
Fluid density (kg/m^3)	998.2
Number of segments per borehole	12

4.4 Sensitivity to Number of Borehole Segments

Since pygfunction approximates each borehole as a number of segments, and over each segment the heat flux is constant, the solution depends on the number of segments per borehole. Furthermore, the dependence on the number of segments seems to vary with overall depth and number of boreholes. This complicates the sensitivity analysis, which is still underway as we write. Therefore, the results in this section are preliminary. To begin the analysis, we analyzed the values of the g-functions at specific times. For example, Figure 7 shows the variation of the g-value with the number of segments for a 10x10 borehole field with 5m spacing and 275m depth.

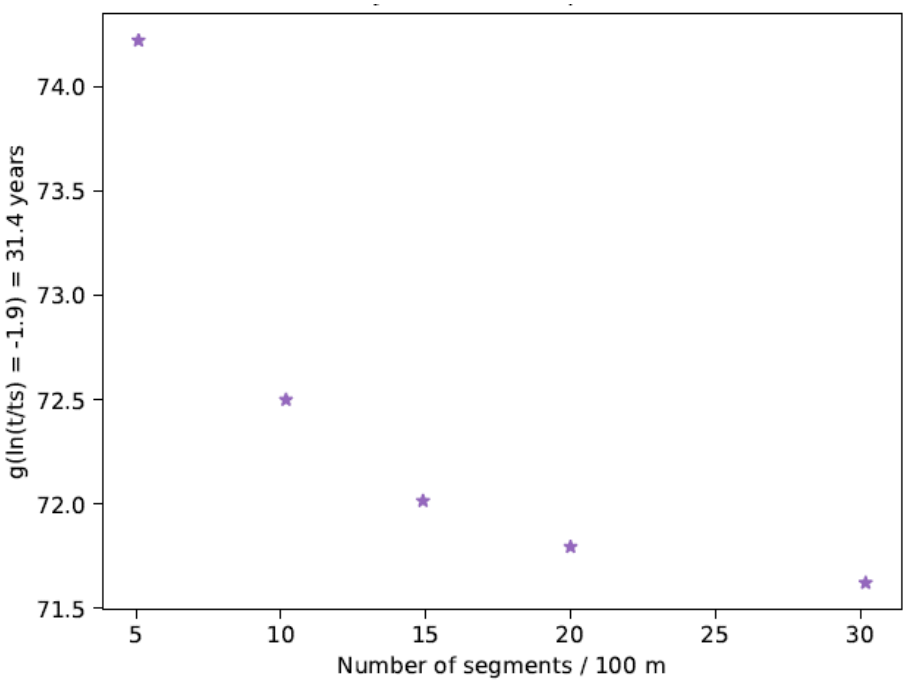

Figure 7: g-values at 31.4 years vs. number of segments for a 10x10 borefield, 275 m deep.

Based on many results like this for fields as large as 10x10, we are preliminarily treating 30 boreholes/100m as a reference calculation. The attentive reader may see the problem. When attempting to calculate the reference g-function for a 32x32 borehole field, 400m deep, we have 122,880 segments. Based on the estimates in Section 4.1, such a simulation might require about 23 days and 1.3 terabytes of RAM. The Pete cluster has one node with 1.5 TB, but it is not usually possible to obtain use of it for more than a week.

Therefore, to date, we have only been able to make rather coarse estimates of the possible error for different numbers of segments. By treating 30 segments/100m as a reference case, the error may be plotted, as shown in Figure 8. Here, errors with 4x4, 6x6, 8x8, and 10x10 boreholes are plotted as points for different numbers of segments per 100m. Cases with 10.2 segments/100m (i.e., 28 segments/275m) and 14.9 segments/100m are extrapolated as shown. The authors concede that these extrapolations may not be sufficiently accurate! The computational work necessary to calculate actual errors with higher numbers of boreholes is underway now. Given the limited data to date suggests that on the order of 10-15 segments per 100 m depth may be sufficient for calculation of g-functions with acceptable error (under 5%).

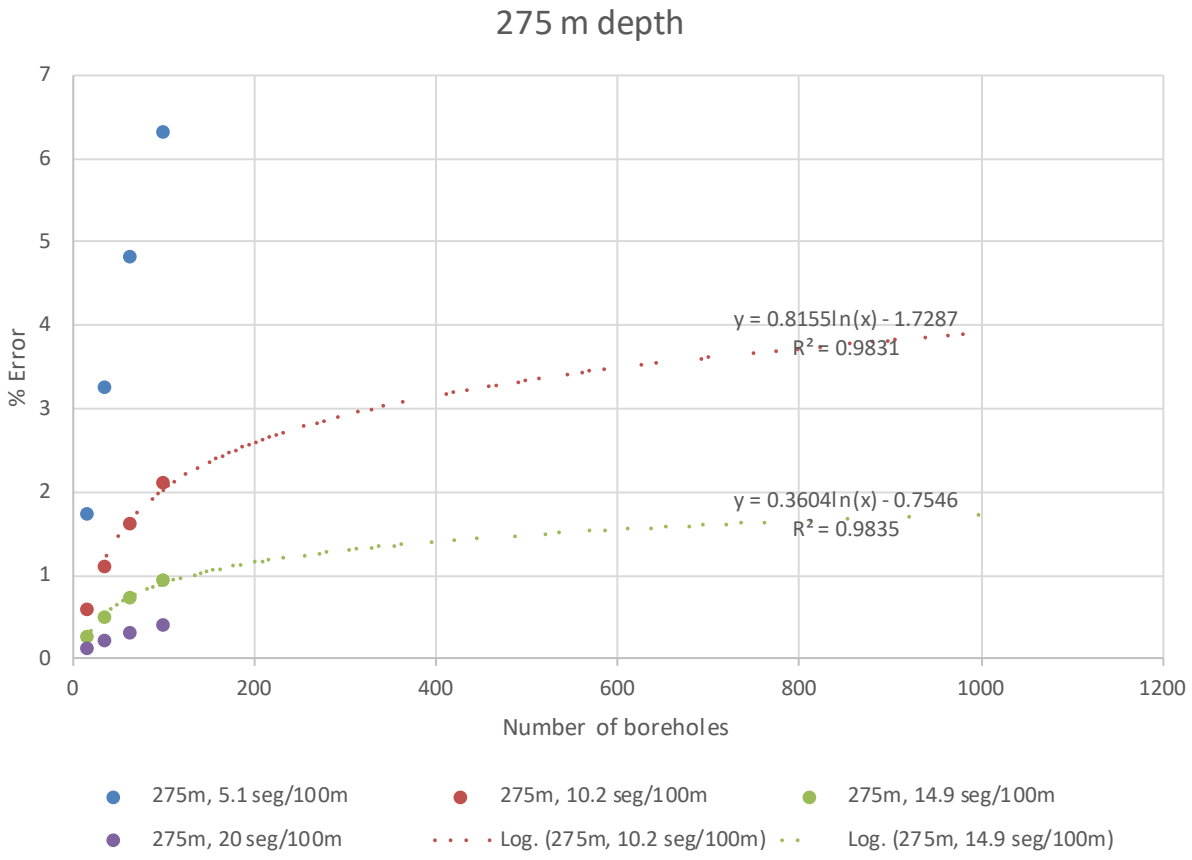

Figure 8: Error in g-value for cases with 16-100 boreholes, extrapolated to 1000 boreholes.

4.5 Verification of Interpolation Used to Compute Depth-Specific g-functions

As discussed in Section 4.1, our approach is based on developing g-functions for several borehole depths, then, when sizing, interpolating between the g-functions associated with the pre-computed borehole depths. We are using Lagrange polynomials to do this. Figure 9 shows an example of g-functions pre-computed for four different borehole depths. The dashed lines represent the Lagrange polynomial interpolations. Finally, Figure 10 shows a comparison between the interpolated g-function and one calculated for a depth of 65 m. The interpolation is nearly perfect, with a difference of 0.01% in the area beneath the curves (g-functions).

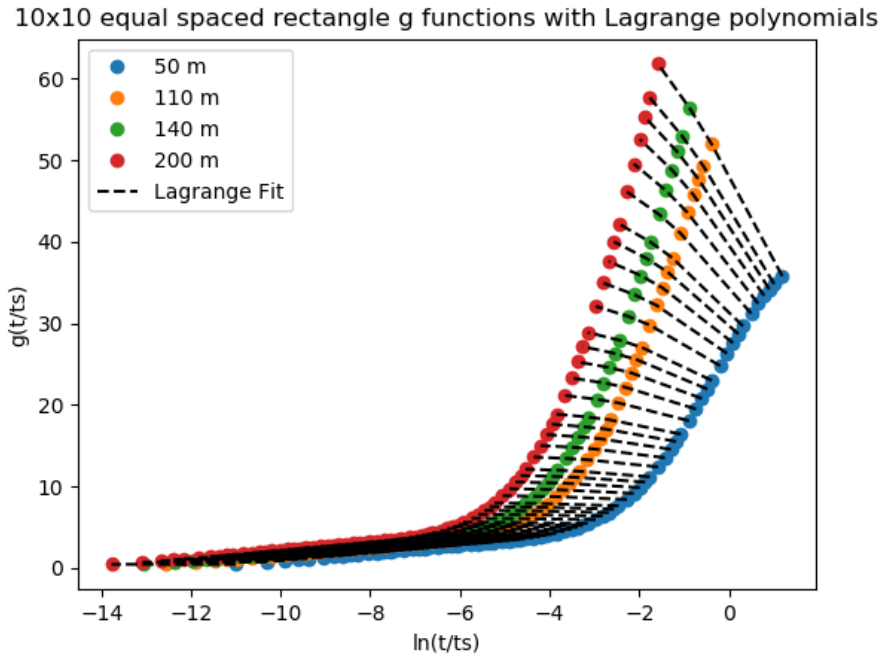

Figure 9: Lagrange polynomial interpolation based on g-functions pre-computed for four different borehole depths.

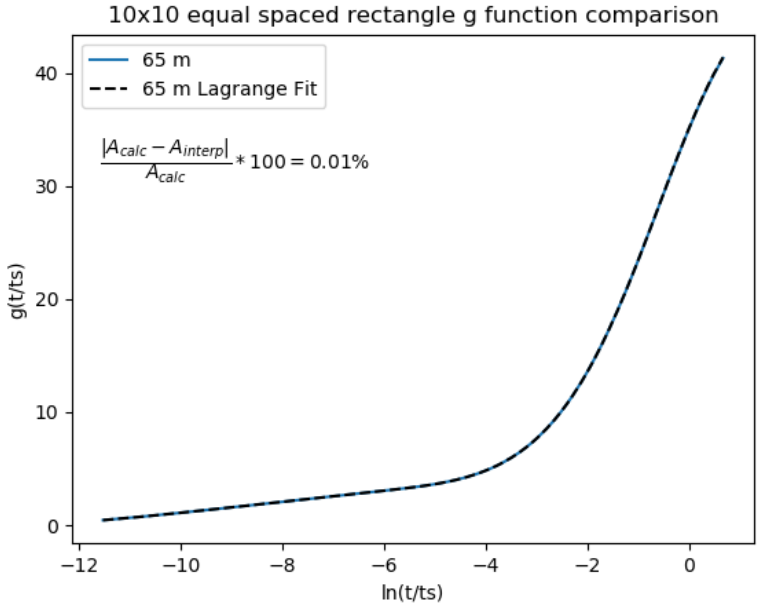

Figure 10: Comparison of interpolated g-function to one calculated with pygfunction for the exact borehole depth.

5. Conclusions

This paper describes our experiences using a modified version of Cimmino's pygfunction (2019a and 2019b) to generate g-functions for a wide range of borehole configurations, including some that exceed 1000 boreholes in size. It is possible to calculate g-functions for large borehole configurations, but there are challenges with computational time, memory requirements and adequately establishing the accuracy for large fields. For smaller borehole fields, up to 20x10, it has been shown to give excellent accuracy, with less than 1% difference in the predicted long-term temperature rise or fall compared with that predicted with the original g-functions. Verification of accuracy for larger fields and a wider range of depths is underway.

Acknowledgement

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE), under the Geothermal Technologies Office, Low-Temperature and Co-Produced Resources Program. The authors would like to thank Dr. Massimo Cimmino – without his development of the open-source pygfunction, this work would not have been possible. The authors also appreciate the guidance and inputs of Mrs. Arlene Anderson, the Lead Technology Manager of Low-Temperature Geothermal Program at U.S. Department of Energy.

REFERENCES

Beier, R. A., M. S. Mitchell, J. D. Spitler and S. Javed. 2018. *Validation of borehole heat exchanger models against multi-flow rate thermal response tests*. Geothermics 71(Supplement C): 55-68.

Beier, R. A. and J. D. Spitler. 2016. Weighted average of inlet and outlet temperatures in borehole heat exchangers. Applied Energy 174: 118-129.

Brussieux, Y. and M. Bernier. 2019. *Universal short time g*-functions: generation and application*. Science and Technology for the Built Environment 25(8): 993-1006.

Cimmino, M. 2014. Development and Experimental Validation of Response Factors for Geothermal Well Fields (in French), Polytechnique Montréal, Canada.

Cimmino, M. 2015. The effects of borehole thermal resistances and fluid flow rate on the g-functions of geothermal bore fields. International Journal of Heat and Mass Transfer 91: 1119-1127.

Cimmino, M. 2018a. A finite line source simulation model for geothermal systems with series- and parallel-connected boreholes and independent fluid loops. Journal of Building Performance Simulation 11(4): 414-432.

Cimmino, M. 2018b. pygfunction: an open-source toolbox for the evaluation of thermal. eSim 2018, Montreál, IBPSA Canada. 492-501.

Cimmino, M. 2019a. "pygfunction GitHub Page." Retrieved October 24, 2019, from https://github.com/MassimoCimmino/pygfunction.

Cimmino, M. 2019b. Semi-Analytical Method for g-Function Calculation of bore fields with series- and parallel-connected boreholes. Science and Technology for the Built Environment 25(8): 1007-1022.

Cimmino, M. and M. Bernier. 2013. *Preprocessor for the generation of g-functions used in the simulation of geothermal systems.* . 13th International IBPSA Conference, Chambery, France.: 2675-2682.

Cimmino, M. and M. Bernier. 2014. *A semi-analytical method to generate g-functions for geothermal bore fields*. International Journal of Heat and Mass Transfer 70: 641-650.

Cimmino, M. and P. Eslami-Nejad. 2017. A simulation model for solar assisted shallow ground heat exchangers in series arrangement. Energy and Buildings 157(Supplement C): 227-246.

Claesson, J. and P. Eskilson. 1985. *Thermal analysis of heat extraction boreholes*. Proceedings of 3rd International Conference on Energy Storage for Building Heating and Cooling ENERSTOCK 85, Toronto, Canada, Public Works Canada. 222–227.

Cullin, J. 2008. Improvements in Design Procedures for Ground Source and Hybrid Ground Source Heat Pump Systems. MSME, Oklahoma State University.

Cullin, J. R., J. D. Spitler, C. Montagud, F. Ruiz-Calvo, S. J. Rees, S. S. Naicker, P. Konečný and L. E. Southard. 2015. *Validation of vertical ground heat exchanger design methodologies*. Science and Technology for the Built Environment 21(2): 137-149.

Dusseault, B. and P. Pasquier. 2018. *Near-instant g-function construction with artificial neural networks*. 2018 IGSHPA Research Conference, Stockholm, IGSHPA. 450-458.

Dusseault, B. and P. Pasquier. 2019. Efficient g-function approximation with artificial neural networks for a varying number of boreholes on a regular or irregular layout. Science and Technology for the Built Environment 25(8): 1023-1035.

Dusseault, B., P. Pasquier and D. Marcotte. 2018. *A block matrix formulation for efficient g-function construction*. Renewable Energy 121: 249-260.

Eskilson, P. 1986. Superposition Borehole Model - Manual for Computer Code. University of Lund.

Eskilson, P. 1987. *Thermal Analysis of Heat Extraction Boreholes*. PhD Doctoral Thesis, University of Lund.

Eskilson, P. and J. Claesson. 1988. *Simulation Model for Thermally Interacting Heat Extraction Boreholes*. Numerical Heat Transfer 13(2): 149-165.

Hammock, C. W. and S. Sullens 2017. Final Report. Coupling Geothermal Heat Pumps with Underground Seasonal Thermal Energy Storage. ESTCP Project EW-201135., ESTCP.

Hellström, G. 2006. E-mail. J. Spitler.

Lazzarotto, A. 2015. *Developments in Ground Heat Storage Modeling* Doctoral Thesis, KTH, Sweden.

Lazzarotto, A. 2016. A methodology for the calculation of response functions for geothermal fields with arbitrarily oriented boreholes – Part 1. Renewable Energy 86: 1380-1393.

Lazzarotto, A. and F. Björk. 2016. A methodology for the calculation of response functions for geothermal fields with arbitrarily oriented boreholes – Part 2. Renewable Energy 86: 1353-1361.

Malayappan, V. and J. D. Spitler 2013. Limitations of Using Uniform Heat Flux Assumptions in Sizing Vertical Borehole Heat Exchanger Fields. Clima 2013. Prague.

Mitchell, M. S. 2019. An Enhanced Vertical Ground Heat Exchanger Model for Whole Building Energy Simulation. Doctoral Thesis, Oklahoma State University.

Monzó, P., P. Mogensen, J. Acuña, F. Ruiz-Calvo and C. Montagud. 2015. A novel numerical approach for imposing a temperature boundary condition at the borehole wall in borehole fields. Geothermics 56: 35-44.

OSU HPCC. 2019. "Learn about Cowboy." Retrieved October 25, 2019, from https://hpcc.okstate.edu/learn-about-cowboy#overlay-context=.

Pasquier, P. 2019. E-mail. J. D. Spitler.

Pasquier, P., A. Zarrella and R. Labib. 2018. Application of artificial neural networks to near-instant construction of short-term g-functions. Applied Thermal Engineering 143: 910-921.

Xu, X. and J. D. Spitler. 2006. *Modelling of Vertical Ground Loop Heat Exchangers with Variable Convective Resistance and Thermal Mass of the Fluid*. 10th International Conference on Thermal Energy Storage - Ecostock 2006, Pomona, NJ. 8.

Yavuzturk, C. and J. D. Spitler. 1999. A Short Time Step Response Factor Model for Vertical Ground Loop Heat Exchangers. ASHRAE Transactions 105(2): 475-485.