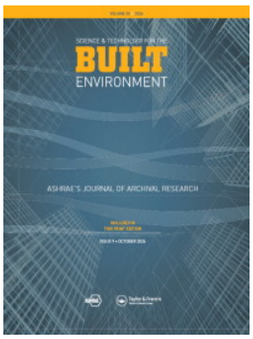


Taylor & Francis
Taylor & Francis Group

Science and Technology for the Built Environment

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/uhvc21

Ground heat exchanger design tool with RowWise placement of boreholes

Timothy N. West & Jeffrey D. Spitler

To cite this article: Timothy N. West & Jeffrey D. Spitler (2024) Ground heat exchanger design tool with RowWise placement of boreholes, Science and Technology for the Built Environment, 30:9, 1148-1166, DOI: 10.1080/23744731.2024.2386882

To link to this article: https://doi.org/10.1080/23744731.2024.2386882

Published online: 21 Aug 2024.

Science and Technology for the Built Environment, (2024) **30**, 1148–1166

ISSN: 2374-4731 print / 2374-474X online
DOI: 10.1080/23744731.2024.2386882

Taylor & Francis
Taylor & Francis Group

Ground heat exchanger design tool with RowWise placement of boreholes

TIMOTHY N. WEST AND JEFFREY D. SPITLER*

MAE, OK State University, Stillwater, OK, USA

Simulation-based design tools have been used since the late 1980s for designing vertical borehole ground heat exchangers (GHE) used with ground source heat pump (GSHP) systems. The ground heat exchanger simulations used in these tools rely on thermal response functions known as g-functions. Because of the significant computational burden in computing g-functions for even a single configuration, the design tools have relied on libraries of pre-computed g-functions. These g-functions were available for standard configuration shapes, such as lines, rectangles, open rectangles, L-shapes, and U-shapes. Standard shapes are often sub-optimal. For any building on a site, the available land may preclude use of a standard shape. For large GSHP systems with significantly imbalanced annual heat rejection and extraction loads, large rectangular fields may experience significant heat build-up (or heat draw-down) in the interior of the field. This paper describes a new ground heat exchanger design tool capable of automatically selecting and sizing both standard and irregular configurations. The focus of this paper is a method for creating, selecting, and sizing irregular configurations where the available land area and "no-go" zones are described as irregular polygons.

Introduction

In the Energy Information Administration's (EIA) April 2023 Monthly Energy Review, the EIA reported that the U.S. commercial and residential sectors accounted for 13% and 16% of all U.S. energy consumption (Energy Information Administration 2023a). The EIA's 2020 Residential Energy Consumption Survey reported that 70% of residential energy consumption went to space heating, water heating, and air conditioning (Energy Information Administration 2023b). The 2018 Commercial Buildings Energy Consumption Survey reported that 51% of commercial building energy consumption went to space heating, cooling, ventilation, and water heating (EIA 2022). Together, these reports can be interpreted to imply that up to 18% of U.S. energy consumption depends on the heating and cooling of commercial and residential buildings. More efficient alternatives to conventional heating, ventilation, and air conditioning (HVAC) technologies could significantly reduce U.S. energy costs and resulting greenhouse gas emissions. One efficient alternative is the ground-source heat pump (GSHP); the GSHP differs from an air-source heat pump by using the ground as a heat sink/source rather than the outdoor air. Under conditions where significant heating or cooling is required, the subsurface ground temperatures are often closer to the desired system temperature than the outdoor air, allowing the system to operate with higher efficiency. Liu et al. (2019) concluded that retrofitting existing HVAC systems with GSHPs across the U.S. could reduce annual energy costs by $49.8 billion. Adoption of GSHP systems across the U.S. is still quite low; this is often attributed to the high installation costs. Therefore, there is significant interest in reducing the initial costs. The ground heat exchanger (GHE) which exchanges heat between the GSHP and the ground with an intermediary coolant often accounts for up to 30% of GSHP installation costs (NYSERDA 2017). When the available land area is limited or soil conditions preclude installation of a horizontal GHE, a vertical borehole ground exchanger (VBGHE) is installed. While the VBGHE can be constructed in a much smaller surface area than horizontal GHE, the drilling can further increase GHE installation costs (Liu et al. 2018). The combination of large potential energy savings with prohibitive installation costs has created a strong incentive for the development of methods to optimize VBGHE's.

In situations where the annual heat rejection and extraction are significantly imbalanced, long-term temperature build-up or draw-down occur due to thermal interference between the boreholes. This effect can be mitigated by adjusting the loads on the GHE with supplemental heat sources or sinks, increasing the number of boreholes, or optimizing the placement of boreholes. Optimizing the placement of

Received November 1, 2023; accepted June 26, 2024
Timothy N. West, BE, Student Member ASHRAE, is a Graduate Research Assistant. **Jeffrey D. Spitler, PhD, Fellow ASHRAE,** is a Regents Professor.
*Corresponding author e-mail: spitler@okstate.edu

boreholes in a VBGHE can help to reduce the total drilling requirements. The subject of this paper is a newly developed tool that can optimize the placement of boreholes. Previous work on optimizing placement of boreholes is summarized below.

A significant milestone in optimizing GHEs was the development of thermal response functions by Prof. Claesson of Lund University and his graduate students (Claesson and Eskilson 1985, 1988). Known as g-functions, these response functions allowed for the simulation of ground heat exchangers with multiple vertical boreholes, accounting for borehole-to-borehole thermal interference. The effects of thermal interference are particularly important for larger ground heat exchangers used with GSHP systems serving commercial and institutional buildings. By pre-computing the thermal response, g-functions present a faster alternative for thermal simulations (once calculated) over other methods.

Existing commercially-available design tools – GLHEPRO (Spitler 2000; OSU 2016) and EED (BLOCON 2022) – use libraries of pre-calculated g-functions for standard shapes (lines, rectangles, etc.). EED has a feature that allows irregularly shaped borefield layouts to be mapped to similar library borefield layouts but does not support automated layout of irregular borefields. These tools size the ground heat exchangers to meet user-specified design temperature constraints, relying on multi-year simulations to estimate the peak temperatures. The depth and configuration are iteratively adjusted to choose a design that just meets the design temperature constraints. In the case of GLHEPRO, configurations are chosen manually by the designer; EED has a feature to automatically iterate over configurations and depth.

Cimmino and Bernier (2014) presented two studies investigating the effects of adjusting borehole spacing to concentrate boreholes closer to the interior or exterior of the field and adding or removing boreholes. The studies were conducted on a 3×7 and a 5×10 rectangular field with a borehole spacing of 7 m. The first study adjusted the spacing of boreholes in each row (the longer dimension). The spacings were increased progressively from the edge of the row to the center. This concentrated the boreholes on the left and right edges of their fields. The study also included two examples that adjusted the spacing in a similar way to concentrate boreholes in the center of each row. The study showed that adjusting the spacing to concentrate the boreholes in the center resulted in the g-function for each field increasing and a resulting increase of 0.8% and 0.9% to the required drilling for each field. It also showed a small decrease in the g-function for both fields and a negligible change for both fields when adjusting the spacing to concentrate boreholes toward the edges. It should be noted that the small gains/losses due to this spacing adjustment likely undervalues the significance of concentrating boreholes near the exterior of the borefield as the fields considered are quite small (<51 boreholes) and the difference in borehole placements between the designs is also quite small compared to possible changes (such as moving a larger number of boreholes to the perimeter of the field). The second study both added and removed a column (the shorter length) from each field to produce four new fields (covering the same land area). The study revealed that adding a column resulted in a higher g-function for both original fields and an increase of 2% and 2.2% respectively in the required drilling. Conversely, removing a column resulted in fields with lower g-functions and a 2.2% and 2% reduction in the required drilling. This study suggests that removing boreholes from a borefield can produce better performing designs but does require each individual borehole to be longer. In summary, Cimmino and Bernier (2014) found that the removal and careful placement of boreholes could minorly improve the performance of a borefield; however, their results likely underpredict the possible performance gains from this type of optimization due to the small field sizes, apparently well-balanced annual heat rejection/extraction loads and limited variation in the borefield configurations.

Guo et al. (2017) presented a case study comparing the performance between four rectangular fields with 36 boreholes across a 30 m×30m area. The first field was an evenly spaced rectangular field. The next three fields moved interior boreholes toward the perimeter with the last borefield containing no interior boreholes. The thermal simulation used simple fixed borehole loadings during the summer and winter and found that the increase in ground temperature over a 20-year simulation was decreased when boreholes were moved from the interior to the exterior. The biggest reduction in temperature was the field with all boreholes on the perimeter which reduced the maximum from 20.0 °C to 15.2 °C. Although this study did not investigate the effects of moving boreholes to the perimeter of the borefield on the required total drilling, the results presented do suggest that there is the potential to reduce the required drilling.

Gultekin, Aydin, and Sisman (2019) characterized the behavior of line/rectangular shaped fields with a thermal interaction coefficient. Empirical equations are derived to predict the thermal interaction coefficient as a function of number of boreholes and aspect ratio; the effect of borehole spacing is considered when determining the field heat transfer rate. The parameterization work revealed that the thermal interaction coefficient (and presumably the total drilling length) decreases with aspect ratio (the lowest being a line of boreholes). This also seems to suggest that minimizing the boreholes in the interior of a field would lead to better performing GHEs as higher aspect ratios lead to fields with more boreholes along the perimeter.

Spitler, Cook, and Liu (2020) compared a uniformly spaced rectangular configuration to a custom configuration where the boreholes were wrapped around the building. For the specific case of an office building in Atlanta, with a high imbalance between annual heat rejection and extraction, wrap-around configurations could achieve drilling savings of 34–43% depending on the depth constraint. This highlights the importance of effectively using all available space on a property. The design of the wrap-around configuration took many engineer-hours to locate the boreholes, iteratively adjusting the number of boreholes and borehole positions,

and calculating g-functions (taking many computer hours) for each configuration. The large time investment makes it impractical for real-world application.

The work presented so far involves methods to improve the placement of boreholes but requires manual adjustment and iteration to finalize a design. Recent work on automated placement of boreholes is discussed below.

Beck et al. (2013) developed a method to design both the borefield layout and individual borehole heat exchanger (BHE) loading. The loading for each BHE was adjusted using linear programming to minimize the maximum field temperature, though it is not clear in practice how the loading to individual boreholes would be controlled. An evolutionary algorithm was used to optimize the borehole positions. The thermal simulations rely on a modified superposition of the finite line source model. Eight reference points were placed around each borehole which were used to track field temperatures. Several implementations utilizing the optimization methods were presented. The most effective implementation was the simultaneous optimization of both the borehole locations and individual loadings which reduced the maximum ground temperature change from 10.5K to 9.4K. This was the most time-consuming method as the loadings had to be re-optimized for each borefield layout tried in the evolutionary algorithm. The implementation that only optimized the borehole positions with an equal loading also had a significant reduction from the base case (10.5K to 9.5K). This method is limited to positioning boreholes and cannot add or remove them.

Bayer, de Paly, and Beck (2014) described a method for designing GHEs by starting with a pre-defined configuration with excess boreholes, then systematically removing boreholes based on their effectiveness (while also exploring seasonal workload optimization). The borehole(s) with the largest corresponding drop in ground temperature is removed at each step. The iteration process is only stopped once the loading per borehole reaches a user-given limit. The field performance is measured as the maximum temperature difference in the field after a 15-year simulation. Removing boreholes showed the largest reduction in temperature change of 2 K when the heating and cooling loads were the most imbalanced.

Robert and Gosselin (2014) described a GSHP system optimization that selects uniformly spaced rectangular borefields utilizing a cost function. Borehole height, borehole spacing, and percentage of total heat load are three design variables that are adjusted with an optimization algorithm to minimize the installation cost for a set field configuration. An iterative procedure that adds either a row or column at each step is used to determine the rectangular borefield with the minimum installation cost. This system successfully incorporates borefield optimization and could easily be adapted to optimize a borefield for a set land area (by adjusting borehole spacing along with the number of boreholes). However, this procedure is limited to optimizing rectangular fields.

Hénault, Pasquier, and Kummert (2016) presented a method to optimize the cost (with an estimate of the net present value of the system) of a hybrid ground-coupled heat pump system by parameterizing the GHE with six design variables. Three of these describe the borehole layout – x and y distances between the boreholes and the total number of boreholes. Three other factors describe the GSHP side of the system–minimum and maximum EFT and the number of heat pumps used. A nonlinear optimization algorithm is used to minimize the cost using these six factors. As described in the paper, the optimization is limited to rectangular and L-shaped ground heat exchangers.

Egidi, Giacomini, and Maponi (2021) utilized the steepest descent optimization algorithm to optimize the placement of a set number of boreholes. The objective function in this method consists of the variation in ground temperature for a series of evaluation points. The steepest descent optimization algorithm adjusts the borehole locations to lower the variance while constrained to a user-defined boundary. The magnitude of performance increase is somewhat unclear, but an improvement is shown with the optimized field. The optimal solutions all simply increase the uniform spacing between 16 boreholes rather than shifting borehole positions relative to one-another. The reason for this is unclear and it seems likely that a more efficient field could be found by shifting one or more boreholes to the perimeter.

Noël and Cimmino (2022) presented a method to optimize borehole fields with an irregular layout by iteratively discretizing the property and optimizing the borehole placement and number on that discretization. Each discretization is refined until the borehole field layout converges which provided a reduction in the total drilling length of 9.8% in the case study presented. The method shows promise but requires more development and testing.

This paper[1] describes a recently developed design tool, GHEDesigner, which can automatically design and size borehole configurations. That is, it automatically places the boreholes and determines the drilling depth that meets the design temperature constraints. GHEDesigner can design 'regular' configurations such as uniformly spaced rectangles, bi-uniformly spaced rectangles, zoned rectangles (Spitler et al. 2022a) and others. However, the focus of this paper is the capability to design and size borehole layouts for irregularly-shaped properties for which both the property boundary and internal "no-go zones" are specified as irregular concave or convex polygons.

GHEDesigner

GHEDesigner is an open-source ground heat exchanger design tool released in 2023. (Spitler 2023) GHEDesigner has six borefield design algorithms which all produce a borefield design that minimizes the total drilling length for a GHE while meeting maximum drilling depth and temperature constraints. These algorithms fall into three categories:

[1]This paper is a significantly expanded version of Spitler, West, and Liu (2022b) presented at the 2022 IGSHPA Conference.

a constant-spacing search, variable-spacing searches, and irregular searches. An earlier version of GHEDesigner, GHEDT (Ground Heat Exchanger Design Tool) was developed by Cook and described in his MS thesis (2021).

Thermal simulation and GHE sizing

Like GLHEPRO and EED described earlier, GHEDesigner uses multi-year simulations in an iterative fashion, adjusting the borehole configuration and depth, followed by a multi-year simulation to determine the peak. The search algorithms in GHEDesigner are only feasible due to fast multi-year simulations of candidate ground heat exchangers. The simulations have the following features:

- Long time step (LTS) g-functions are computed with Pygfunction (Cimmino 2018). The fastest LTS g-function computation times rely on the equivalent borehole method which significantly reduces computation time for large borefields while losing very little accuracy (Prieto and Cimmino 2021). The speed is further increased by using eight nonuniform segments for each borehole in the borehole discretization (Cook 2021). The accuracy of this discretization scheme was investigated in section 4.3.1 of Cook (2021) which found that the accuracy of the 8-segment discretization (with the equivalent borehole method) has a small oversizing effect (<2%) on the sized borehole height. Due to the time savings provided by the 8 nonuniform segment discretization, this small oversizing effect is deemed acceptable. Cimmino, Cook, and Isiordia Farrera (2024) have made further improvements to the accuracy by investigating the optimal end segment length, but the examples developed in this paper still use the scheme presented by Cook. These updates reduce the computation time of LTS g-functions to less than a second for most borefields with less than 144 boreholes (Prieto and Cimmino 2021). The increased speed allows "on the fly" computation of the g-functions, eliminating the need for pre-computed libraries.
- The short time step (STS) g-functions are computed with the Xu and Spitler (2006) method.
- The borehole thermal resistances are computed using the multipole method (Claesson and Hellström 2011). Both the STS g-functions and borehole resistances are compatible with single-U, double-U, and coaxial BHEs.
- The simulation is done with a hybrid time-step scheme proposed by Cullin and Spitler (2011). It is also possible to use hourly simulations, but they are generally much slower than desirable for simulation-based design.
- The results of a simulation are a list of GHE exiting fluid temperatures (ExFT) also known as the GSHP entering fluid temperatures (*EFT*) for each hybrid time-step, with the peak values over the multi-year design period being of the most interest. The list of temperatures is referred to as EFT_{sim}.

The design process has two steps with the first being a search for feasible configurations, based on a user-specified maximum depth. The second step (sizing) determines the final required depth. The search options are described in the following sections. The sizing process starts by computing three LTS g-functions for the borefield selected in the first step. These are computed at a maximum and minimum borehole height (user-given) and the average between the two. Further g-functions are approximated with quadratic spline interpolation between these three initial g-functions. The accuracy of the interpolation between three g-functions and the effect on the resulting borehole height is discussed in sections 2.4.3 and 4.3.2 of Cook (2021). The inaccuracy produced by the interpolation was found to have less than a 0.1% effect on the borehole height. This error inaccuracy is deemed acceptable as g-functions for large borefields (several hundred boreholes) can require a few seconds to calculate (see Prieto and Cimmino 2021). The sizing algorithm uses SciPy's (Virtanen et al. 2020) implementation of Brent's method (Brent 1973) to determine the borehole height where the excess temperature (*ET*) is zero. This excess temperature serves as the only constraint on the borehole sizing algorithm and monotonically decreases with respect to increasing borehole height, so utilizing Brent's method on *ET* (with respect to borehole height) minimizes the total drilling required for a GHE subject to $ET > 0$. Excess temperature is defined in Equation 1.

$$ET := \max\left(\max\left(EFT_{sim}\right) - EFT_{\max}, EFT_{\min} - \min\left(EFT_{sim}\right)\right) \tag{1}$$

ET represents how under or oversized a GHE is for a user specified EFT_{max} and EFT_{min}. A positive value means that the GHE is undersized and cannot meet the user-specified temperature constraints. A negative value suggests that the GHE is oversized and exceeds the requirements. A value of zero indicates that the GHE perfectly meets the given temperature constraints; the EFT_{sim} (at some point during the simulation) reaches either the maximum or minimum user-specified allowable *EFT*.

Constant-spacing search

GHEDesigner has one constant-spacing search – the "square/near-square" search – that uses a user-specified fixed spacing (B_s) between boreholes. The upper bound on the NBH is set by a user given side length (L_s) which defines the largest square of land available. The algorithm begins by determining the maximum number of boreholes that can fit along one side of the user given square (N_{max}) which is defined in Equation 2 below:

$$N_{\max} := \left\lfloor \frac{L_s}{B_s} \right\rfloor + 1 \tag{2}$$

Equation 3 defines the function *R* which returns a set of ordered pairs representing *x*-*y* coordinates for a given: number of columns (N_x), number of rows (N_y), spacing between

[2]The equations regarding sets presented in this paper are meant to adhere to the International Organization for Standardization's (ISO's) standards on mathematics (2019).

columns (B_x), and spacing between rows (B_y).

$$R(N_x, N_y, B_x, B_y) := \{(x_i B_x, y_i B_y) | x_i, y_i \in \mathbb{Z}, 0 \le x_i < N_x, \; 0 \le y_i < N_y\} \quad (3)[2]$$

The function (C) defines a set of ordered pairs representing a borefield corresponding to a given value of i. A near-square is returned when i is divisible by 2 and a square is returned otherwise.

$$C(i) := \begin{cases} R\left(\left\lceil \frac{i}{2} \right\rceil, \left\lceil \frac{i}{2} \right\rceil, B_s, B_s\right), & i\%2 = 1 \\ R\left(\left\lceil \frac{i}{2} \right\rceil, \left\lceil \frac{i}{2} \right\rceil + 1, B_s, B_s\right), & i\%2 = 0 \end{cases} \quad (4)$$

The function in Equation 4 is used to help define the domain D_{sns} (see Equation 5) that the square/near-square search will search through. The domain is a sequence (indirectly ordered by the NBH/cardinality of each set), so the search part of the algorithm can make use of the fact that the total drilling is monotonic with respect to the domain indices.

$$D_{sns} := (C(d_i))_{d_i=1}^{2N_{\max}-1} \quad (5)[3]$$

The square/near-square search uses an integer bisection search on D_{sns}. Since D_{sns} is ordered by cardinality, NBH and therefore ET over the domain form a monotonic function with one zero/root. The search will select the field with the maximum nonpositive excess temperature (the smallest NBH which has a nonpositive excess temperature).

Variable-spacing searches

Three types of searches in GHEDesigner use regular configurations with rectangular constrained surface areas (and variable spacings). These are the uniform/bi-uniform rectangular searches and the bi-uniform zoned rectangular search, all of which accept length (L_x) and width (L_y) as surface constraints.

In addition to the length and width, the uniform search requires two spacing inputs: B_{min} and B_{max}. The domain generation part of this search works slightly differently depending on whether L_x or L_y is larger, so, for the purposes of a clear explanation, it will be assumed that $L_x \ge L_y$ (although the algorithm would work similarly if the opposite were true). These are used to calculate a $N_{x,max}$ and $N_{x,min}$ (Equations 6 and 7 below).

$$N_{x,\max} := \left\lfloor \frac{L_x}{B_{\min}} \right\rfloor + 1 \quad (6)$$

$$N_{x,\min} := \left\lceil \frac{L_y}{B_{\max}} \right\rceil + 1 \quad (7)$$

Since the spacing in the x and y direction is constant for the uniform search, the borehole spacing (B) and the maximum number of boreholes in the y-direction (N_y) is computed based on a given value of N_{xi} (see Equations 8 and 9).

$$B(N_{xi}) := \frac{L_x}{N_{xi} - 1} \quad (8)$$

$$N_{y,RU}(N_{xi}) := \left\lfloor \frac{L_y}{B(N_{xi})} \right\rfloor + 1 \quad (9)$$

The domain for the uniform rectangle spacing is formed from three parts. The first part ($D_{RU,1}$) consists of a series of line borefields with the maximum borehole spacing shown in Equation 10. The second part ($D_{RU,2}$) consists of partial rectangles with the maximum borehole spacings shown in Equation 11. The last part of the domain ($D_{RU,3}$) contains full rectangles of decreasing borehole spacing (and increasing NBH) and is defined in Equation 12. Concatenating these three sequences forms the sequence D_{RU} which represents the domain of the uniform spacing rectangular search (shown in Equation 13). The uniform rectangular search has a single monotonic domain, so the uniform search can utilize the same bisection search as the square/near-square search algorithm.

$$D_{RU,1} := \left(R(d_i, 1, B(d_i), 1)\right)_{d_i=1}^{N_{x,\min}} \quad (10)$$

$$D_{RU,2} := \left(R(N_{x,\min}, d_i, B(N_{x,\min}), B(N_{x,\min}))\right)_{d_i=1}^{N_{y,RU}(N_{x,\min})} \quad (11)$$

$$D_{RU,3} := \left(R(N_{xi}, N_{y,RU}(N_{xi}), B(N_{xi}), B(N_{xi}))\right)_{N_{xi}=N_{x,\min}}^{N_{x,\max}} \; where: \; N_{y,RU}(N_{xi}) \ne N_{y,RU}(N_{xi} - 1) \quad (12)$$

$$D_{RU} := D_{RU,1} \oplus D_{RU,2} \oplus D_{RU,3} \quad (13)[4]$$

The bi-uniform search requires three spacing parameters: B_{min}, $B_{max,y}$, and $B_{max,x}$. The bi-uniform rectangular search begins like the uniform search but instead determines the number of boreholes that can be placed on the shorter side, $N_{y,\max}$ and $N_{y,\min}$, calculated like Equations 6 and 7 (understanding that $L_x \ge L_y$ like before). The bi-uniform search also calculates the y-direction spacing with a function (B_y) like B in Equation 8 but utilizes L_y. The bi-uniform search domain is two-dimensional. The search domain is constructed as a set of smaller inner domains. The inner domains ($D_{RB,i}$), are constructed like D_{RU} with only one major difference. The y-direction spacing is independent from the x-direction spacing and the y-direction spacing is set to a constant value provided as a parameter. The full rectangular search (D_{RB}) is defined in Equation 14 below (keep in mind that $D_{RB,i}$ is a function used to define each inner domain). Each inner domain is also monotonic, so the domains are searched in two successive steps. The first step performs a bisection search on each inner domain and keeps track of the selected borefields. Second, the selected fields from each inner domain are compared with each other using total drilling as a metric. The borefield with the lowest total drilling is returned.

[3]To the authors' knowledge, there is not a consistent set of standards regarding sequences. This equation is meant to represent a sequence of sets of ordered pairs:
$(C(d_i))_{d_i=1}^{2N_{\max}-1} = (\mathrm{C}(1), \mathrm{C}(2), \mathrm{C}(3), \ldots, \mathrm{C}(2\mathrm{N}_{\max} - 1))$.

[4]The character "$\oplus$" is meant to represent the operation of sequence concatenation. That is:
$(\mathrm{a}_1, \mathrm{a}_2, \ldots, \mathrm{a}_n) \oplus (\mathrm{b}_1, \mathrm{b}_2, \ldots, \mathrm{b}_n) = (\mathrm{a}_1, \mathrm{a}_2, \ldots, \mathrm{a}_n, \mathrm{b}_1, \mathrm{b}_2, \ldots, \mathrm{b}_n)$

$$D_{RB} := \left(D_{RB,i}(L_x, L_y, B_{\min}, B_{\max,x}, B_y(N_{yi}))\right)_{N_{yi}=N_{y,\min}}^{N_{y,\max}} \quad (14)$$

The bi-uniform constrained zoned rectangular search differs from the other variable spacing searches by generating fields with different interior and perimeter spacing (Cook 2021). The zoned search utilizes the same input parameters as the bi-uniform rectangular search. The domain for this search is more complex (although the structure is the same as the bi-uniform rectangular search) than the previous three search methods and is not as closely related to the irregular searches (the focus of this paper), so the reader is encouraged to read section 4.4.3 of Cook (2021) or Appendix 2 of Spitler et al. (2021) for more details on the formation of the domain. This domain is a set of monotonic inner domains, so the method to search through the domain is the same as the bi-uniform rectangular search.

Irregular searches

There are two searches in GHEDesigner that fit this category. This category is unique in that the borefields can have surface constraints described as irregular polygons, with "no-go zones" that are also described as irregular polygons.

BUPCRS

The first type is the bi-uniform polygonal constrained rectangular search (BUPCRS), described by Cook (2021). This search is based on the bi-uniform rectangular search which means that the domain (D_{BUPCRS}) is formed like D_{RB}. The major difference is that each set of coordinates is trimmed to fit a given property boundary. To this end, the BUPCRS algorithm requires two more parameters than the bi-uniform rectangular search: an outer polygon and a set of inner polygons. Let PB be the set of all ordered pairs of points within the outer polygon and let NG be the set of all ordered pairs of points outside of every inner polygon. Then L_x and L_y are determined by putting a bounding box around the outer polygon. The function R is replaced with R_{BUPCRS}, as shown in Equation 15, to create the BUPCRS domain. One note about D_{BUPCRS}, is that the resulting inner domains are monotonic with respect to NBH but not always ET. This is because the trimming of boreholes can sometimes create multiple fields with the same NBH. These fields do not have the exact same coordinates, so care must be taken when searching the inner domains to compare the total drilling between fields of the same NBH (if they exist) to the selected borefield; otherwise, the domain search part of the BUPCRS algorithm works like the bi-uniform rectangle search.

$$R_{BUPCRS}(N_x, N_y, B_x, B_y) = R(N_x, N_y, B_x, B_y) \cap PB \cap NG \quad (15)$$

RowWise methodology

The RowWise methodology consists of two algorithms that work to optimize the placement of boreholes across an irregularly shaped property with irregularly shaped no-go zones. The first algorithm, the RowWise Placement Algorithm, parameterizes borehole placements and the second algorithm, the RowWise Search/Optimization Algorithm, selects fields generated by the Row Wise Placement Algorithm to meet design requirements. The methodology is named so because it places boreholes in rows. The orientation of the rows varies depending on the situation. The original idea behind the usage of this parameterization was to promote convenient borehole placement (for installation) when optimizing fields. In practice, the degree to which the boreholes appear in rows varies with the geometry. Nevertheless, the parameterization is still quite useful as it can represent quite complex fields with three or four parameters. There are three versions of the RowWise Search/Optimization Algorithm. The first two versions utilize one-dimensional searches as described in the "RowWise Search Algorithm" section; the third version utilizes a multi-dimensional optimization algorithm as described in the "RowWise Optimization Algorithm" subsection. The "Proposed Design Algorithms and RowWise Testing Algorithms" subsection describes the specific implementations of the search/optimization algorithms developed for the purpose of testing the RowWise algorithm. These algorithms are all either based on the existing optimization/search algorithms with small changes to the input parameters or were specifically developed to compare the search algorithms to a more exhaustive search.

RowWise placement algorithm

This algorithm parameterizes borefields for a given property described by the boundary and no-go zones. The property boundary is defined by a list of points representing a convex or concave irregular polygon; the points can be ordered clockwise or counterclockwise. Each no-go zone polygon is likewise defined by a list of points. Boreholes are placed in parallel rows inside the property and outside the no-go zones. These rows can be rotated relative to the x-axis. Additionally, boreholes can be placed along the perimeter of the property and no-go zones (rather than in standard rows) when the independent perimeter placement feature is enabled. For a given scenario, a RowWise field is defined with four parameters: intra-row spacing (S_{intra}), inter-row spacing (S_{inter}), perimeter spacing (S_p), if independent perimeter placement is enabled, and field orientation (R_f). Figure 1 below visualizes the effect of these inputs on the resulting field.

The first part to understand about the RowWise algorithm is how boreholes are distributed. Equations 16–19 below detail this process. Equation 16 defines N_{row}: the number of boreholes that can be placed along a row with the given spacing. Next, Equation 17 illustrates the calculation of B_{adj} which will be the actual distance between boreholes in a row. Third, the normalized vector (d_{row}) pointing from p_2 to p_1 is determined as in Equation 18. Lastly, Equation 19 shows the definition of the set of ordered pairs representing the x-y locations of the points placed along a particular

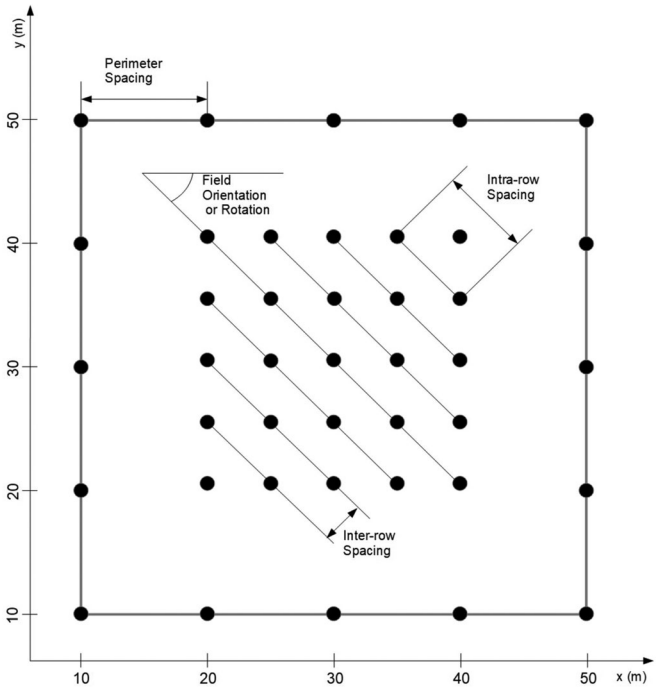


Fig. 1. Visualization of RowWise parameters.

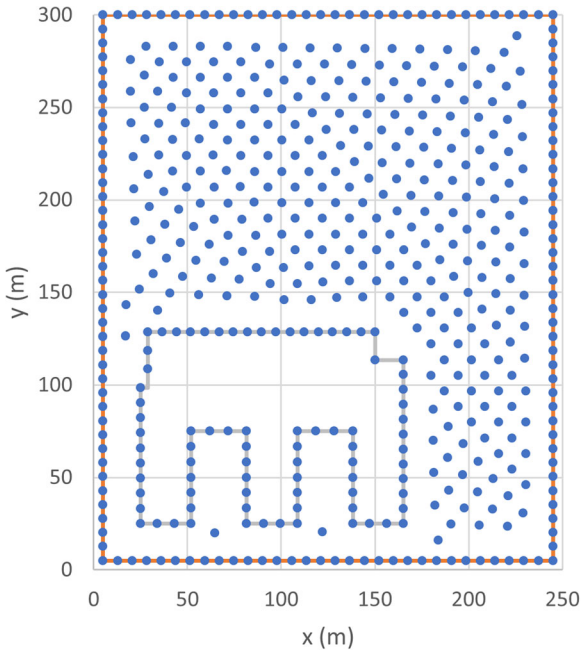


Fig. 2. Example RowWise field (row orientation: –45°).

row.

$$N_{row}(p_1,p_2) = \left\lfloor \frac{dist(p_1,p_2)}{S_{intra}} \right\rfloor \tag{16}$$

$$B_{adj}(p_1,p_2) := \frac{dist(p_1,p_2)}{N_{row}(p_1,p_2)} \tag{17}$$

$$d_{row}(p_1,p_2) := \frac{(p_2(0)-p_1(0), p_2(1)-p_1(1))}{dist(p_1,p_2)} \tag{18}$$

$$Row(p_1,p_2) := \left\{ p_1 + (iB_{adj}(p_1,p_2))d_r(p_1,p_2) | i \in \mathbb{Z}, 0 \le i < N_{row}(p_1,p_2) \right\} \tag{19}$$

The other key part of the RowWise Placement Algorithm is the distribution of rows. Rows are distributed in three steps. The first step is to distribute a set of parallel lines across the property boundary representing where rows could be placed. The second step is to determine where these lines intersect with the property boundary and no-go zones. The third step is to then parse each line's intersections and create rows along each section of line that is both inside the property boundary and outside every no-go zone Since the orientation of the rows is likely not parallel to either the x or y axis, (defined by R_f) the "highest" and "lowest" vertices of the property boundary relative to a rotated x-axis (rotated to be parallel to the field orientation) are determined. If v represents a vertex of the outer polygon, then its "height" relative to this rotated axis would be defined as (assuming the vertex is in the first Cartesian Quadrant):

$$H(v) := dist((0,0),v) * \sin\left(\arctan\left(\frac{v(1)}{v(0)}\right) - R_f\right) \tag{20}$$

Lines are distributed across the property (parallel to the rotated x-axis) between the "highest" and "lowest" vertices. The inter-row spacing is used in a similar way to the intra-row spacing (in Equations 16 and 17). This means that the orthogonal distance between each parallel line is at least S_{inter} but is likely slightly larger. For each of these lines, rows are created from the intersections with the line and the property boundary/no-go zones.

The row boreholes and perimeter boreholes (if independent perimeter placement is enabled) are generated separately. The row boreholes are generated as discussed above. Perimeter boreholes (with the independent perimeter placement enabled) are generated in two steps. First, the row boreholes that are within a user-given distance of the property or a no-go zone are removed as they would overlap with the perimeter boreholes. The user-given distance is more of a minimum as the interaction between the way boreholes are placed and the system of removal can lead to slightly larger spacings between the property or no-go zone boundary and the interior borehole rows. Next, perimeter boreholes are generated along the property and no-go zones with the perimeter spacing parameter. Figure 2 contains an example of a RowWise field generated with independent perimeter placement enabled (the row orientation is −45°). Here, the increased spacing between the property and no-go zone boundaries and the interior boreholes is visible.

RowWise search algorithm

The first two versions of the RowWise search/design algorithm use a one-dimensional bisection search, with and without the independent perimeter placement feature. Both perform a bisection search to find a RowWise field that has the lowest total drilling while still maintaining a nonpositive excess temperature. RowWise fields are not guaranteed to form monotonic domains with one zero like the other field

types in GHEDesigner. Several strategies are used to allow a one-dimensional search.

The first strategy separates field rotation as a parameter from the one-dimensional search. Figure 3 below demonstrates the sensitivity of total drilling with row orientation for an example field. In this case, RowWise searches have been made for row orientations between −90° and 0° at 0.5° intervals. This can have a significant effect on the performance of a generated borefield (≥3% in this example), but the effect of field orientation on a generated field is unpredictable and erratic, and other parameters strongly influence the performance of each row rotation. This creates a hindrance to using standard optimization procedures. An exhaustive search, where a RowWise search or optimization is performed for every rotation interval, could be done, but this significantly increases the required computation time. Instead, for a fixed interval between 0° and −90°, an exhaustive search is performed to determine the row rotation corresponding to the borefield with the minimum number of boreholes and maximum number of boreholes with the RowWise placement algorithm. These values are used to select a single row orientation for sizing; this selection method is referred to as the maximum or minimum NBH heuristic based on whether the rotation corresponding to the maximum or minimum NBH is selected. The performance of choosing either orientation will be investigated in the "Rotation Domain Investigation" section.

The second strategy for creating a one-dimensional, monotonic domain with a single root involves reducing the spacing parameters into a single parameter. This is done by choosing a single intra-row and inter-row target spacing. Next, if independent perimeter placement is used, the perimeter spacing is set by a user-defined ratio of this target spacing. If, for example, the user-given ratio is 0.7 and the current target spacing is 10 m, the perimeter target spacing would be set to 7 m. Figure 4 demonstrates a typical relationship between total drilling and target spacing. The relationship is not strictly monotonic though it often is. There is generally a strong trend – the total drilling reduces with increasing target spacing. To eliminate the possibility of converging to a local optimum, the initial search is followed by an exhaustive search near the converged optimum. The additional search does not usually produce a better solution but may in some cases. This search can be disabled if a user does not wish for the tradeoff.

While these strategies successfully reduce the number of parameters in the search to one, it leaves some concerns. The perimeter spacing is defined by a user-given ratio which will either require user-intuition or a second outer search (which is investigated in a later section). Additionally, the linking of the intra-row and inter-row spacing can cause limitations for some types of properties. There is room for improvement, but this search method still performs well (as will be discussed later). Figure 5 illustrates the search process for an example problem.

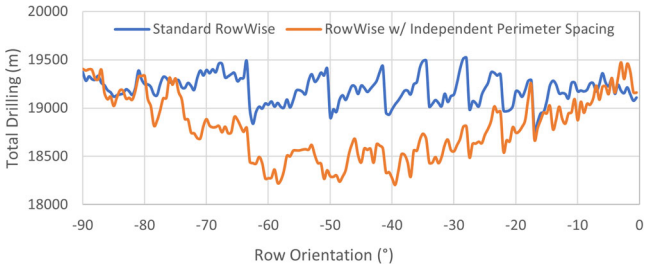


Fig. 3. Sensitivity of total drilling to row orientation for a typical RowWise problem.

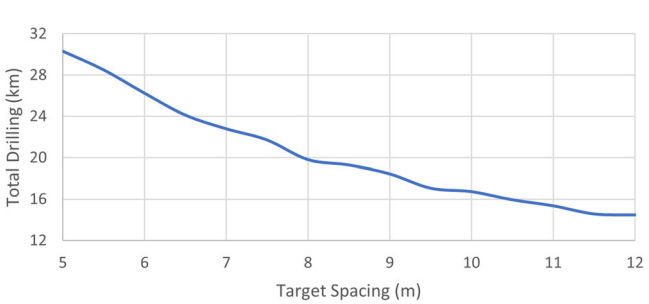


Fig. 4. Total drilling vs. target spacing for a typical RowWise problem.

RowWise optimization algorithm

The third version of the RowWise search/optimization algorithm uses multi-dimensional optimization, so it does not require reducing the problem to a single independent variable. Instead, intra-row, inter-row, and perimeter spacing are optimized with SciPy's implementation of the Nelder-Mead optimization algorithm (Virtanen et al. 2020). The field rotation is treated the same way as the prior search algorithms as the effect of field rotation on borefield performance is erratic and not readily optimizable without an exhaustive search. Each of the three parameters is constrained by user-specified maximum and minimum values. The initial simplex for the algorithm is defined with user-specified coordinates relative to the user-specified bounds (0 corresponds to a minimum and 1 corresponds to a maximum). The current default is to use the following simplex: (1.0, 0.5, 0.5), (0.5, 1.0, 0.5), (0.5, 0.5, 1.0), and (0.5, 0.5, 0). The first three points all have one of the spacings increased from the center-point (it is faster to search the higher spacings as they have lower corresponding NBH values). The fourth point includes a low perimeter spacing value as the GHE layout is highly sensitive to the perimeter spacing selection. This simplex covers a reasonably large

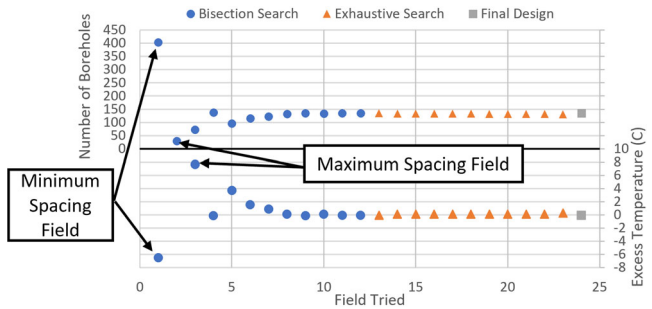


Fig. 5. Typical progression of the RowWise search algorithm for an example problem.

part of the midsection of the search volume while focusing on the section of the domain that contains faster-to-simulate designs. Additionally, an option to both restart the algorithm or divide the domain into smaller sections to help prevent convergence on local minima. By default, no restarts are used (as restarts did not usually improve the resulting GHE design), and the domain is cut into six even parts along planes parallel to the intra-row spacing and inter-row spacing planes. This domain partitioning was selected to split the domain into domains that have a perimeter spacing variation of 2 m (for the examples presented in this paper). The RowWise domain is full of local minima (discussed in the "Domain Investigation" section), and the convergence of the optimization algorithm is sensitive to the given perimeter spacing guesses. One sub-section is likely sufficient if the perimeter spacing range input has been narrowed down to a 2 m range. The resulting minimum from each sub-domain is compared and the best field found is returned.

To ensure that the thermal temperature constraints are met, an objective function utilizing a penalty function is used, as shown in Figure 6 below. For a given set of spacings, fields are generated for each rotation interval. Then, the field with either the maximum or minimum number of boreholes is simulated. If the simulated field (with the maximum allowable borehole height, H_{max}) has a nonpositive excess temperature, the total drilling required to create this field (with a borehole height of H_{max}) is returned as the value of the objective function. Otherwise, a penalty scaled off *ET* defined in Equation 21 is added to the value:

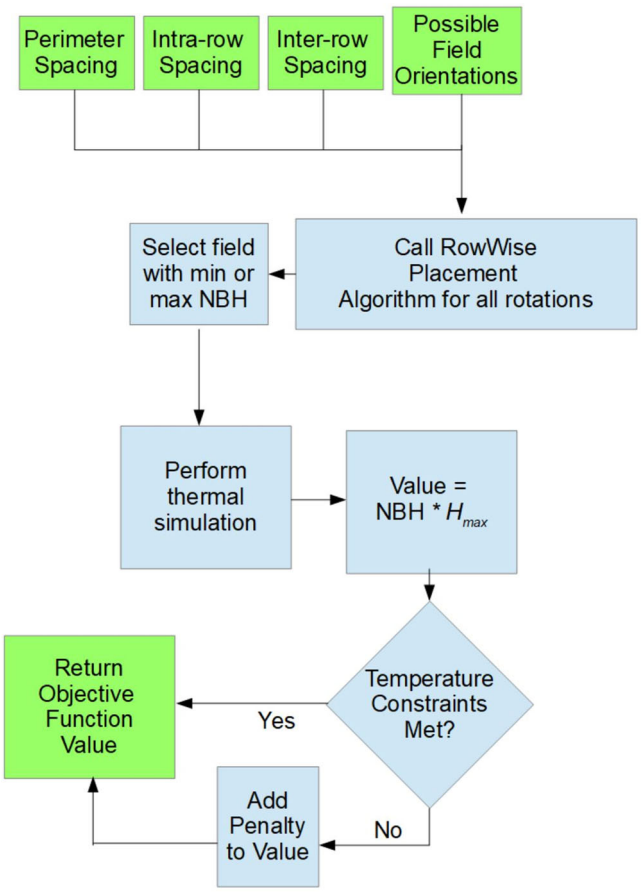


Fig. 6. RowWise optimization algorithm objective function control-flow diagram.

The required total drilling length represents much of a VBGHE's installation cost. As with currently available design tools, GHEDesigner relies on the user to consider other aspects of the design – e.g. decisions on such design elements such as grout conductivity can be accounted for in the tool, and the impact on the total drilling length can be quantified. Still, it would be advantageous to use first cost or life cycle cost as an objective function, but further research is needed (1) to more comprehensively account for costs and (2) automate other aspects of the design such as horizontal piping topology, and (3) to develop optimization procedures capable of optimizing borehole placement and other aspects of the design simultaneously. Future improvements will require refinements to the objective function.

$$penalty = 1e6 * (ET + 10)^5 \tag{21}$$

This penalty has been adjusted to ensure that any field with a nonnegative *ET* is desirable over one with a positive *ET* while trying to limit pushing the solution away from the edge of viable designs (as this is likely where ideal fields will be found). The penalty is very severe, and the constant 10 introduces a large discontinuity in the penalty. The objective function is already discontinuous (due to the nature of the RowWise algorithm) which would require an optimization algorithm that can handle discontinuities, and introducing this discontinuity to the penalty function strongly discourages the optimization algorithm from exploring the solution space in areas with positive excess temperature. The large constant and exponent of 5 were similarly chosen to be large enough to strongly encourage the optimization algorithm back toward the part of the solution space with nonpositive excess temperatures. The penalty function could probably be refined to be less severe (and scale off NBH or TD to ensure that the penalty is still sufficient), but the current definition works reliably with the optimization algorithm presented in this paper for the variety of cases presented and does not impede the optimization algorithm from finding highly optimized designs (see the "Domain Investigation" section). When the optimization algorithm converges onto a field, the field is sized, and the field and corresponding required total drilling are returned.

To both evaluate this design algorithm and inform future work, three studies were conducted with the RowWise search/optimization algorithms. The three studies make use of five illustrative examples with varying property layouts and thermal parameters. The first study compares the borefield designs from all versions of the RowWise design algorithm in addition to the BUPCRS algorithm and a few additional algorithms that were designed for use in the latter two studies. The BUPCRS algorithm will serve as a "base case" for this first study, because to the authors' knowledge, it is the only irregular borefield design algorithm publicly available, other than the RowWise algorithms. The second study investigates the effectiveness of the current rotation selection methods (Min NBH and Max NBH). The third

study investigates the effectiveness of the search/optimization algorithms relative to the RowWise field domain.

Proposed design algorithms and RowWise testing algorithms

Several implementations of the design algorithms discussed above were created for two purposes. The first set of procedures is so called "Proposed Design Algorithms" which are considered in this paper for usage as tools to optimize irregular GHE designs. These algorithms include the BUPCRS algorithm and six implementations of the RowWise algorithm including two implementations of the RowWise optimization algorithms and four implementations of the RowWise search algorithm. The second set of procedures, "RowWise Test Algorithms", was created to help evaluate the effectiveness of the proposed RowWise design algorithms.

Figure 7 below contains a tree which contains all the proposed design algorithms. The leaves of the tree (shown in green) each represent a specific proposed design algorithm, and the parents of each branch/leaf describe an aspect of the branch. For instance, the "Search" node in Figure 8 implies that the four leaves of the branch below utilize the RowWise Search Algorithm discussed in the "RowWise Search Algorithm" section. The "Minimum NBH" and "Maximum NBH" nodes define whether the below algorithms use the minimum or maximum NBH heuristic discussed in the "RowWise "RowWise Search Algorithm" section.

Figure 8 below illustrates the RowWise Test Algorithms. These algorithms were developed for two main purposes: to evaluate how well the proposed design algorithms optimize the borefield design relative to an exhaustive search ("Spacing Domain Investigation) and to evaluate the minimum and maximum NBH heuristic ("Rotation Domain Investigation"). The function of the spacing interval algorithms used for the rotation domain investigation is described in the "Rotation Domain Investigation" section. The exhaustive search works by exhaustively simulating every spacing (perimeter, inter-row, and intra-row spacing) with a set step size (defined in Table 2).

Table 2 describes the different algorithms tested and their respective inputs for the study. There are 11 algorithms in total. The RowWise optimization algorithm used the default initial simplex described in the "RowWise Optimization Algorithm" section. The RowWise independent perimeter placement algorithm was run eight times with the different perimeter spacing ratios (S_{Ratio}) shown in Table 2.

Presenting the layouts of all fields generated would be intractable, so Figure 9 below has been included to give a sense of how the fields generated with the different design procedures vary. The fields shown were generated for illustrative Example 1. The BUPCRS procedure produces regular constant spacing fields; the RW-SchMin procedure maintains much of the row structure that BUPCRS has but is better able to utilize the available area. The RW-OptMin and RW-ISchMin produce similarly structured fields that lose much of the row organization but can produce much more versatile and better performing borefields that effectively utilize the available land area.

Illustrative examples

This section describes the five cases that are used for the three studies presented in this paper. All building load profiles are from simulations using the DoE commercial buildings library (USDOE 2022). Example 1 is based on an actual single-story

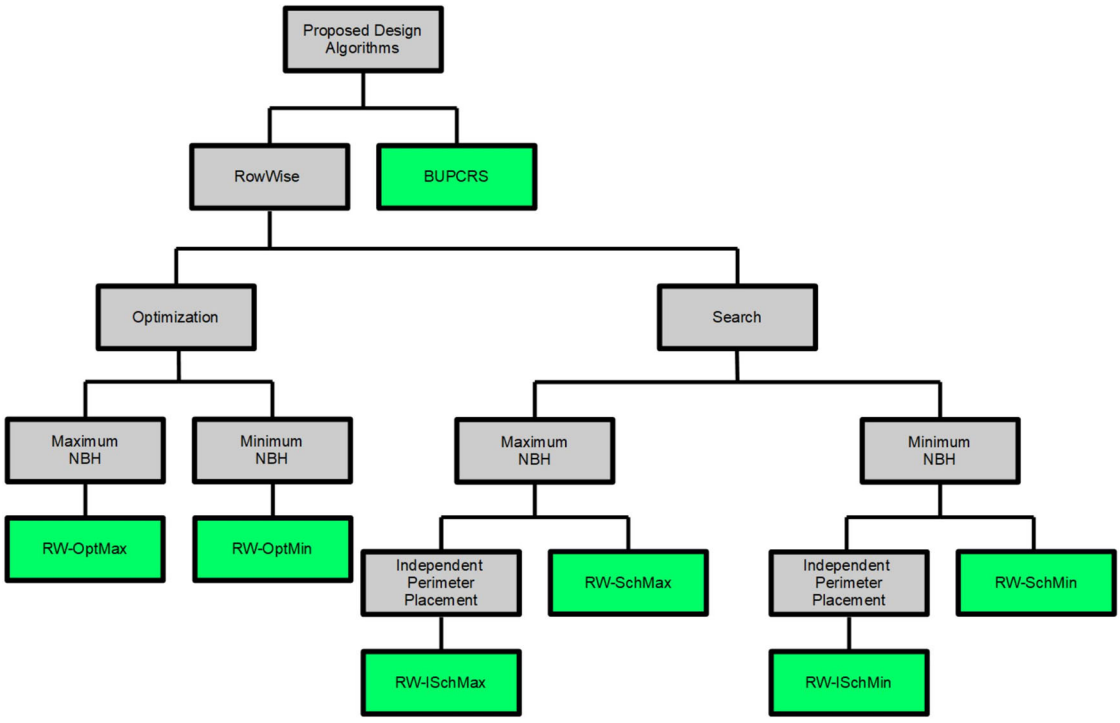


Fig. 7. Proposed design algorithms.

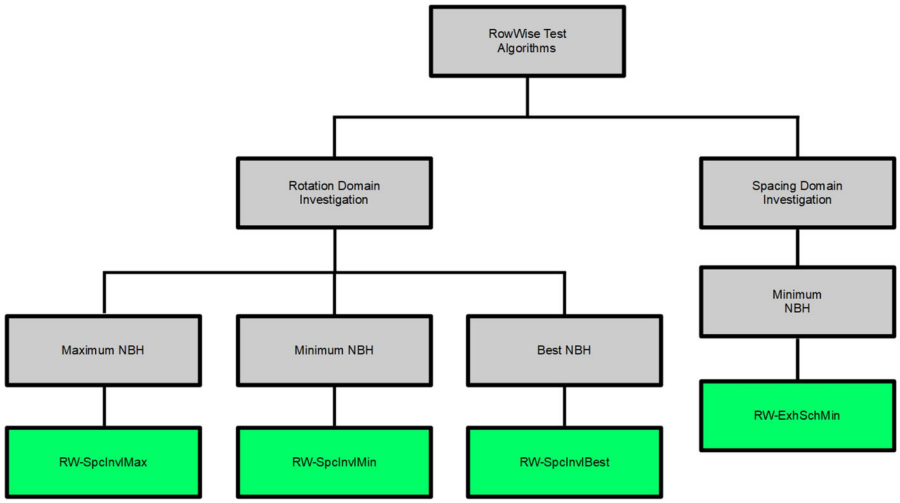


Fig. 8. RowWise test algorithms.

Table 1. Other simulation parameters.

Parameter	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5
Max/min design EFT (°C)	34/9	26/1	35/10	26/1	26/1
UGT (°C)	17.3	9.7	18.3	9.7	9.7
$k_{soil}\left(\frac{W}{m\cdot K}\right)$	2.8	1.7	2.2	1.7	1.7
$c_{rho,\,soil}\left(\frac{kJ}{m^3\cdot K}\right)$	2600	2600	2600	2600	2600
$H_{\max}(m)$	135	100	120	140	120
Loading multiplier (-)	5	1	1	1	1
Ratio annual heat rejection/extraction	24.1	2.5	26.1	1.3	1.1
Total property area (m^2)	9,450	4,900	70,800	39,750	116,530
Total no-go zone area (m^2)	–	1,575	11,220	12,320	25,870
Available land area (m^2)	9,450	3,330	59,600	27,400	90,700

medical office building in Stillwater, Oklahoma, but with the loads scaled up to represent a multi-floor facility on a relatively small property. Example 2 is an apartment building in Minneapolis, Minnesota. Example 3 is a single school building in Atlanta, Georgia. Example 4 is a small school complex in Minneapolis, Minnesota. Example 5 represents a district of buildings in Minneapolis, Minnesota, all served by a common borefield. The five examples share some simulation parameters which are described in Table 3. The borehole completion details (radius, grout properties, U-tube, etc.) and burial depth are constant across the five examples. While these details can significantly affect the design, the focus of this paper is on optimizing the BHE layout and not the borehole completion details or burial depth. The differing simulation parameters necessary to simulate the examples in different locations are shown in Table 1.

The GHE loadings were defined with one-year hourly annual heat extraction rates. The loads are repeated year-to-year for the 20-year simulations. In reality, GHE loads will vary year-to-year in ways that cannot be forecast. Therefore, we use the same approximation as all GHE design tools – that a typical year is repeated. GHEDesigner does support multi-year load inputs, if the user wishes to explore alternative scenarios. The examples differ significantly in the degree to which the annual heat extraction and rejection are imbalanced. Examples 1 and 3 are the most imbalanced with annual heat rejection 24.1X and 26.1X greater than the annual heat extraction. Example 2 has a moderate imbalance with a ratio of 2.5. Examples 4 and 5 have more balanced loads with ratios of 1.3 and 1.1.

The examples also cover a range of property, building and no-go zone shapes and sizes. (For purposes of laying out the borefield, buildings and no-go zones are specified in the same way – as irregular polygons.) Example 1 uses a highly irregular property shape that incorporates the building and a no-go zone into the property polygon. Example 2 consists of a square property with one H-shaped building. Example 3 consists of a rectangular property with a school building in the bottom-left corner of the property. Example 4 is made of a school building with 2 additional square no-go zones placed onto a slightly more complex property. Example 5 is made up of fifteen buildings (12 apartment buildings, a school, hotel, and an office building) distributed across a large rectangular property. Figure 10 shows plan views of each example.

Results and discussion

RowWise procedure comparison

Tables 4–6 below contain the results for the five examples with all design algorithms. Table 4 contains the resulting

Table 2. Summary of algorithms.

Algorithm	Description	Parameter ranges
BUPCRS	Bi-uniform polygon constrained rectangular search	$B_{min} = 3\,\text{m}$, $B_{max,x} = B_{max,y} = 30\,\text{m}$
RW-OptMax	RowWise using optimization; for any combination of spacings, row orientation with max NBH is used.	$S_{p,min} = 3\,\text{m}$, $S_{p,max} = 15\,\text{m}$, $S_{inter,min} = S_{intra,min} = 5\,\text{m}$, $S_{inter,max} = S_{intra,max} = 30\,\text{m}$
RW-OptMin	RowWise using optimization; for any combination of spacings, row orientation with min NBH is used.	$S_{p,min} = 3\,\text{m}$, $S_{p,max} = 15\,\text{m}$, $S_{inter,min} = S_{intra,min} = 5\,\text{m}$, $S_{inter,max} = S_{intra,max} = 30\,\text{m}$
RW-ISchMax	RowWise using search with independent perimeter placement, row orientation with max NBH is used.	$S_{target,min} = 5\,\text{m}$, $S_{target,max} = 30\,\text{m}$, *$S_{Ratio}$ = [0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90, 0.95]
RW-ISchMin	RowWise using search with independent perimeter placement, row orientation with min NBH is used.	$S_{target,min} = 5\,\text{m}$, $S_{target,max} = 30\,\text{m}$, *$S_{Ratio}$ = [0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90, 0.95]
RW-SchMax	RowWise using search, row orientation with max NBH is used.	$S_{target,min} = 5\,\text{m}$, $S_{target,max} = 30\,\text{m}$
RW-SchMin	RowWise using search, row orientation with min NBH is used.	$S_{target,min} = 5\,\text{m}$, $S_{target,max} = 30\,\text{m}$
RW-ExhSchMin	RowWise using an exhaustive search over the target spacings with a set interval with independent perimeter placement, row orientation with min NBH is used.	$S_{p,min} = 4\,\text{m}$, $S_{p,max} = 20\,\text{m}$, $S_{p,step} = 0.5\,\text{m}$, $S_{inter,min} = S_{inra,min} = 4\,\text{m}$, $S_{inter,max} = S_{intra,max} = 30\,\text{m}$, $S_{inter,step} = S_{intra,step} = 2\,\text{m}$
RW-SpcInvlMax	RowWise using an exhaustive search over the single parameter target spacing with set intervals, row orientation with max NBH is used.	$S_{Ratio} = 0.70$, $S_{target,min} = 4\,\text{m}$, $S_{target,max} = 30\,\text{m}$, $S_{target,step} = 0.05\,\text{m}$
RW-SpcInvlMin	RowWise using an exhaustive search over the single parameter target spacing with set intervals, row orientation with min NBH is used.	$S_{Ratio} = 0.70$, $S_{target,min} = 4\,\text{m}$, $S_{target,max} = 30\,\text{m}$, $S_{target,step} = 0.05\,\text{m}$
RW-SpcInvlBest	RowWise using an exhaustive search over the single parameter target spacing with set intervals, an additional exhaustive search is performed over the row orientations for each target spacing.	$S_{Ratio} = 0.70$, $S_{target,min} = 4\,\text{m}$, $S_{target,max} = 30\,\text{m}$, $S_{target,step} = 0.05\,\text{m}$

*The RW-ISch methods only take one S_{Ratio} value, but an outer loop compared the resulting fields from the S_{Ratio}'s and selected the field with the lowest total drilling.

required drilling for each design. Table 4 only contains the minimum required total drilling from all 8 perimeter spacing ratios run for the RW-ISch algorithms. Table 5 contains the computation times, rounded to the nearest 10 s. Computation times vary with CPU loadings, so the times should be thought of as indicative. Table 6 contains the number of objective function calls which is another indication of computation time. The computation times for the RW-SpcInvlMax and RW-SpcInvlMin algorithms were not tracked as they were implemented as part of the RW-SpcInvlBest algorithm. The computation times and number of objective function calls listed for the RW-ISch procedures are the average of a single call across all eight perimeter spacing ratios. This function call includes the borefield generation, simulation, and evaluation. The BUPCRS algorithm generates borefields differently than the RowWise procedures leading to faster computation times even though it has a similar number of objective function calls to the RowWise Search algorithms.

The fields designed with the BUPCRS algorithm had the largest required total drilling for all examples; however, the BUPCRS algorithm was also the fastest of the procedures for all examples. The best performing algorithm in terms of required total drilling is the RW-ExhSchMin procedure. This algorithm's resultant borefields had the lowest (or tied) required drilling for four of the five examples (it was second to the RW-OptMin and RW-ISchMin algorithms in Example 1). However, this method had the second slowest computation times (between a few hours to over a day depending on the example) and second highest number of objective function calls (an order of ten higher than the third highest). This algorithm is not recommended as a GHE design algorithm as the dramatically higher computation times come with minor reductions in total drilling compared to the better

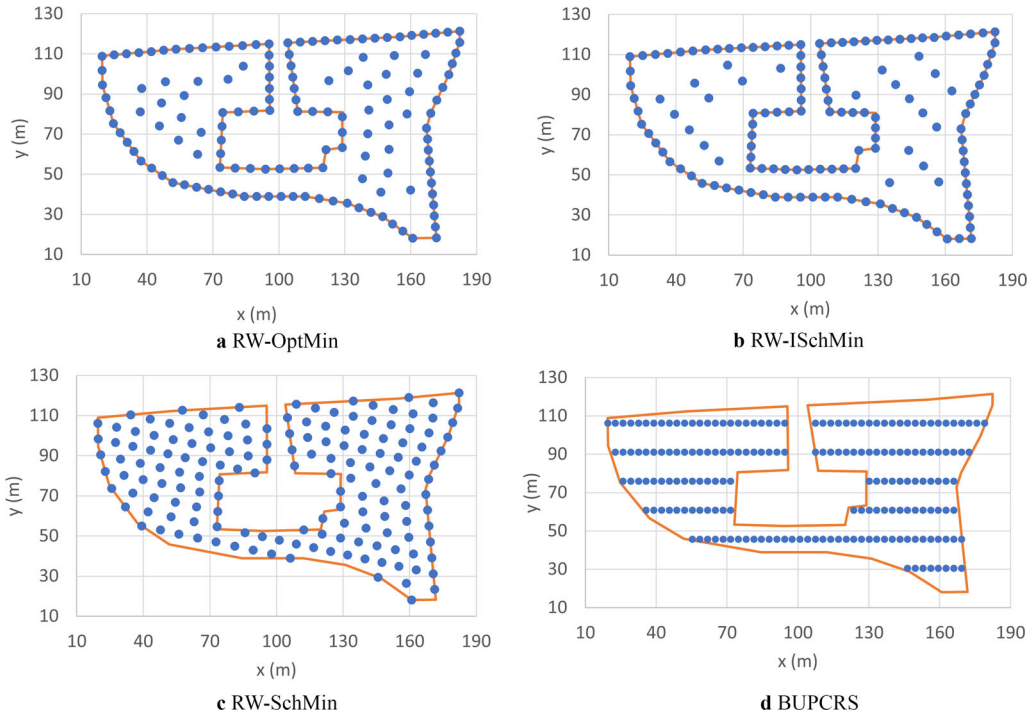


Fig. 9. Four fields for illustrative Example 1. The RowWise fields were generated with the minimum NBH approach.

Table 3. Shared simulation parameters.

Parameter	Value
Simulation time (year)	20
Ground heat exchanger type	Single U-tube
Borehole burial depth (m)	2
Flowrate per borehole (L/s)	0.3
Fluid	Water
Borehole diameter (mm)	150
Outer and inner pipe diameter (mm)	33.4, 37.0
Center-to-center shank spacing (mm)	32.3
Grout thermal conductivity (W/m K)	1.00
Grout volumetric heat capacity (kJ/K m^3)	3901

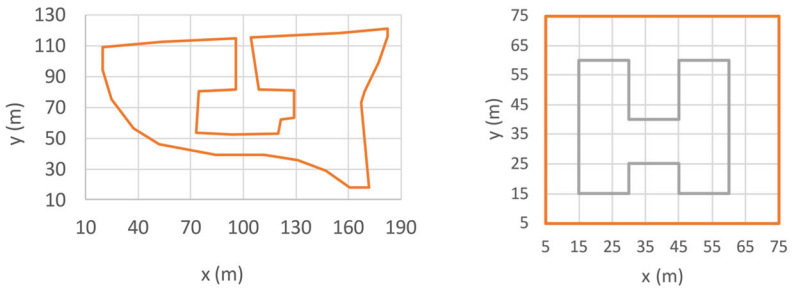
performing RowWise optimization algorithms (the maximum being 1.4% for Example 2).

The overall best performing algorithms (balancing computation requirements and resulting required total drilling) were the RowWise independent perimeter placement search algorithms and the RowWise optimization algorithm. The search algorithm is more consistent at producing fields with low total drilling requirements. The optimization algorithm will sometimes produce "bad" results that are 1–2% worse than the search algorithms. The search algorithms are 1–2 order of magnitudes faster than the optimization algorithms and require an order of magnitude less objective function calls. The search algorithms were run eight times with different perimeter spacing ratios which reduces the computation improvements to a 2/3 decrease from the optimization (which is still a significant reduction). The total drilling for these algorithms is sensitive to this ratio (see Figure 11) where the variation is between the best and worst field is about 5%; a sweep of perimeter spacing ratios is recommended when using this algorithm although usually ratios between 0.6–0.8 provide good results so a larger sweep is likely not necessary in most cases. The independent RowWise perimeter placement search algorithm is likely the best algorithm for the general use case. That said, the optimization algorithm has a valuable usage case. The search algorithms have a limited snapshot of the RowWise field domain. They are limited to keeping the intra-row and inter-row target spacing equal and maintaining a set ratio between the target-spacing and perimeter spacing. This greatly simplifies the RowWise domain making a simple search possible, but it can limit resulting field performance for some geometries. The optimization algorithm does not have these domain limitations but trades them for a much more complex and difficult-to-optimize domain (see "Domain Investigation" section for details).

When comparing the maximum and minimum NBH heuristic methods (i.e., RW-OptMax, RW-ISchMax and RW-SchMax vs. RW-OptMin, RW-ISchMin and RW-SchMin), the better of the two is not clear. Both have similar computation times and objective function calls, and their total drilling performance is similar. They usually produce similar solutions, but one heuristic is not consistently better than the other. The next subsection investigates this further.

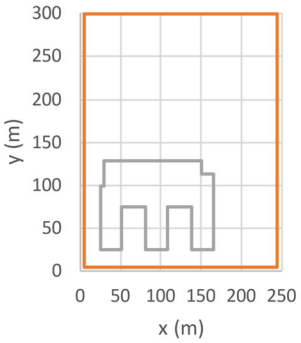
Rotation domain investigation

The minimum and maximum NBH heuristics are compared to a "Best" NBH algorithm which exhaustively compares field orientations. The three methods have been applied to a spacing interval design algorithm. For each target spacing increment, a RowWise borefield is generated with a series

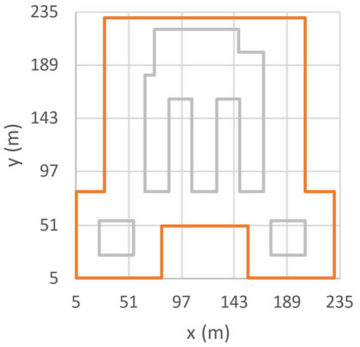


a Example 1

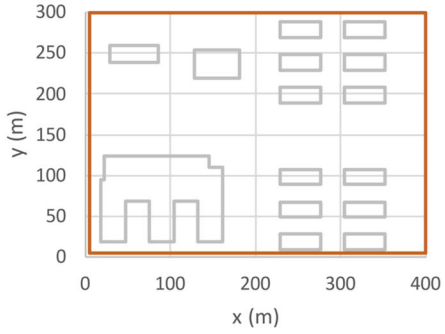
b Example 2



c Example 3



d Example 4



e Example 5

Fig. 10. Example properties from a top-down view. Property boundaries are shown in orange and buildings, or no-go zones are shown in grey.

Table 4. Design total drilling comparison (km).

	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5
BUPCRS	25.3	4.79	70.3	40.0	101.6
RW-OptMax	19.5	4.44	61.5	39.3	100.3
RW-OptMin	19.4	4.44	61.0	39.3	100.4
RW-ISchMax	19.6	4.42	60.1	39.4	100.4
RW-ISchMin.	19.4	4.42	60.2	39.3	100.4
RW-SchMax	21.6	4.44	62.5	39.8	100.4
RW-SchMin	21.4	4.39	64.1	39.9	100.5
RW-ExhSchMin.	19.5	4.36	60.1	39.3	100.4
RW-SpcInvlMax	19.7	4.52	60.7	40.0	100.4
RW-SpcInvlMin	19.8	4.52	60.4	39.6	100.5
RW-SpcInvlBest	19.6	4.42	60.4	39.3	100.4

Table 5. Design investigation computation times (s).

	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5
BUPCRS	20	10	110	50	270
RW-OptMax	1,420	340	4,630	1,810	12,160
RW-OptMin	1,740	420	5,100	2,500	15,990
RW-ISchMax	60	30	300	120	820
RW-ISchMin.	60	30	330	130	880
RW-SchMax	10	30	580	290	840
RW-SchMin	50	30	590	290	850
RW-ExhSchMin.	10,000	3,000	40,000	20,000	100,000
RW-SpcInvlMax	–	–	–	–	–
RW-SpcInvlMin	–	–	–	–	–
RW-SpcInvlBest	16,800	21,300	159,000	60,500	349,000

Table 6. Design investigation objective function calls.

	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5
BUPCRS	19	18	21	21	22
RW-OptMax	567	573	633	531	557
RW-OptMin	580	573	560	617	656
RW-ISchMax	24	24	24	24	24
RW-ISchMin.	24	24	24	24	24
RW-SchMax	24	24	24	24	24
RW-SchMin	24	24	24	24	24
RW-ExhSchMin.	5,408	5,408	5,408	5,408	5,408
RW-SpcInvlMax	95	206	154	149	147
RW-SpcInvlMin	86	206	140	135	135
RW-SpcInvlBest	5,700	12,360	9,240	8,940	8,820

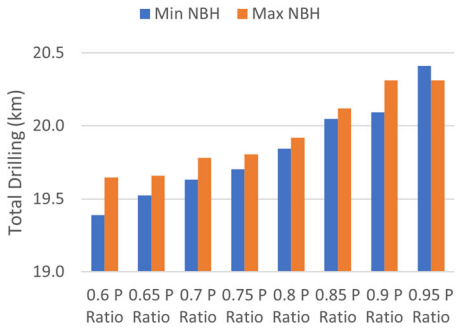


Fig. 11. The total drilling required from the RW-ISch algorithms with different perimeter spacing ratios for Example 1.

of rotations (every 1.5°). Each borefield is simulated with the maximum borehole height. If the resulting excess temperature is nonpositive, the field is then sized and the total drilling for that rotation is recorded; otherwise, the field and its respective rotation are marked as unsatisfactory. Three fields are tracked for this spacing increment: the field with the most boreholes, the field with the least boreholes, and, for all rotations, the field with the smallest drilling length. The maximum target spacing is increased until all rotations produce only unsatisfactory fields. The resulting three algorithms are the RW-SpcInvlMin, RW-SpcInvlMax, and RW-SpcInvlBest algorithms presented in the previous section.

Figures 12 illustrates the domain of Example 4, plotting total drilling vs. target spacing, for the three methods. The total drilling generally decreases with target spacing with all three methods. However, particularly for Example 4, when the rotation with maximum NBH is selected, local minima can be seen. This represents a challenge to finding the global minimum.

Table 7 summarizes the results for all five examples (the total drilling values are in Table 4). Choosing either the rotation with the minimum NBH or the maximum NBH gave results within 2.2% of the exhaustive approach for all cases. In practice, though, both methods will have lower differences when instead of a fixed constant target-spacing step-size, the target-spacing is adjusted with a search method. The minimum NBH method performed better on examples 1 and 5 but the maximum NBH method performed better on

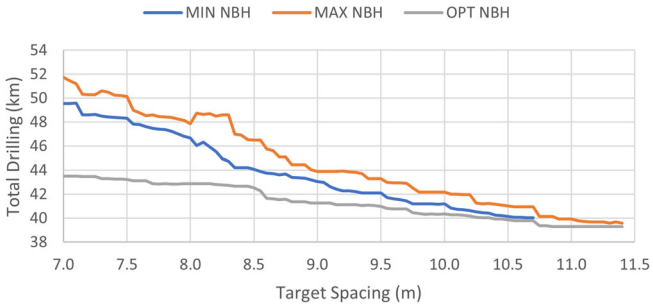


Fig. 12. Total drilling results for the Example 4 rotation domain investigation.

examples 3 and 4 (they performed the same on example 2). Considering the potential computation time savings, either the maximum or minimum NBH seem sufficient as a heuristic. Total drilling could be minimized further by running an optimization procedure with both heuristics and selecting the better of the two. The exhaustive search approach could also be used as a final refinement stage as the high computation time would be less of an issue.

Domain investigation

The RowWise search/optimization methods are compared against an exhaustive search to better evaluate their performance relative to the "global minimum". Table 8 contains the percent difference in required total drilling between the exhaustive search and optimization algorithm (RW-ExhSchMin and RW-OptMin respectively). The maximum difference in the required drilling between the two methods is 1.8%. Since the difference in performance between the optimization and exhaustive search is relatively small even for the optimization algorithm's worst performing examples, the current default values for domain sub-sectioning, restarts, and initial simplexes are likely satisfactory. The additional computation time required for further sub-sectioning or restarts is likely not worth the small gains in field performance (at least by default). With some engineering intuition, the design variable bounds could be adjusted, and the default number of subsections could be reduced to improve computation time substantially.

To better understand the cause of the optimization algorithm's limitations, Figures 13 and 14 show contour plots of the domains for Examples 1 and 4 (created from the results of the exhaustive search). To keep the visualization tractable, the intra-row and inter-row spacings are averaged. When there are two data points with the same perimeter and average spacing, the data point with the smaller total drilling is displayed. This average forms the x-axis while the perimeter spacing forms the y-axis. The plotted surface is the required drilling in km of the resulting borefield, for which we are seeking the minimum. The red line represents the boundary of viable borefields. Spacing combinations above and to the right of this line cannot meet the design temperature constraints with maximum depth boreholes. Near the red boundary lines, the domains are relatively flat (in magnitude) when compared to sections near the axes but are not smooth.

Table 7. Rotation investigation summary.

	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5
% Difference between RW-SpcInvlBest and RW-SpcInvlMin	0.49	2.17	0.57	1.84	0.03
% Difference between RW-SpcInvlBest and RW-SpcInvlMax	1.21	2.17	0.00	0.72	0.15

Table 8. Domain investigation summary..

	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5
% Difference between RW-ExhSchMin and RW-OptMin	−0.51	1.82	1.49	0.0	0.0

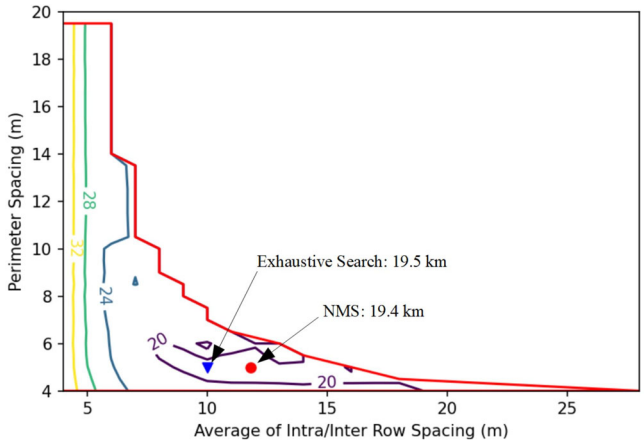


Fig. 13. Km of drilling required for Example 1.

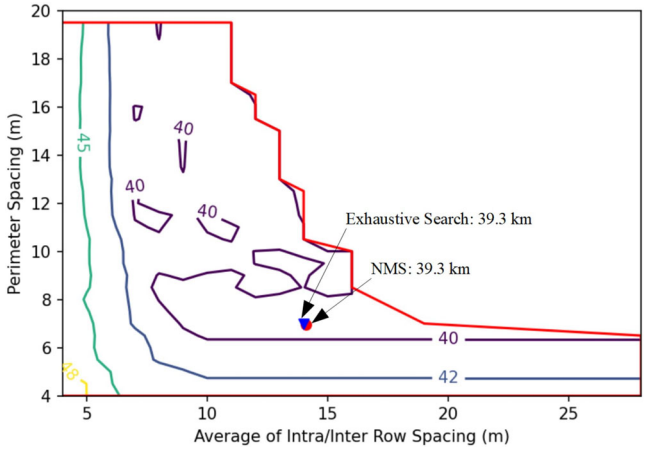


Fig. 14. Km of required drilling contour plot for Example 4.

The fields from the RW-OptMin (NMS in the figures) algorithm and the RW-ExhSchMin algorithm (Exhaustive Search in the figures) are included in Figures 13 and 14.

The solutions are close in both value and location. Both examples show some evidence of local minima but lack the fidelity to fully illustrate the shape of the domain in the flatter regions. Figure 15 contains the results from another exhaustive search on Example 1 but refined to a smaller region with smaller intervals. The perimeter spacing bounds for the exhaustive search were adjusted to 4 m and 7 m with 0.1 m intervals; the intra-row spacing, and inter-row spacing bounds were adjusted between 4 m and 30 m with 1.3 m intervals. The plot was cropped to show the area of most interest and only contains total drilling values between 19.65 and 19.40 km rather than the 32 km through 20 km shown in Figure 13. This contour plot better displays the numerous local minima in this small section of the domain. A more complex optimization procedure would be necessary to search among the local minima. This higher fidelity exhaustive search found a field with only 19.3 km of required total drilling (a 0.5% reduction from previous exhaustive search).

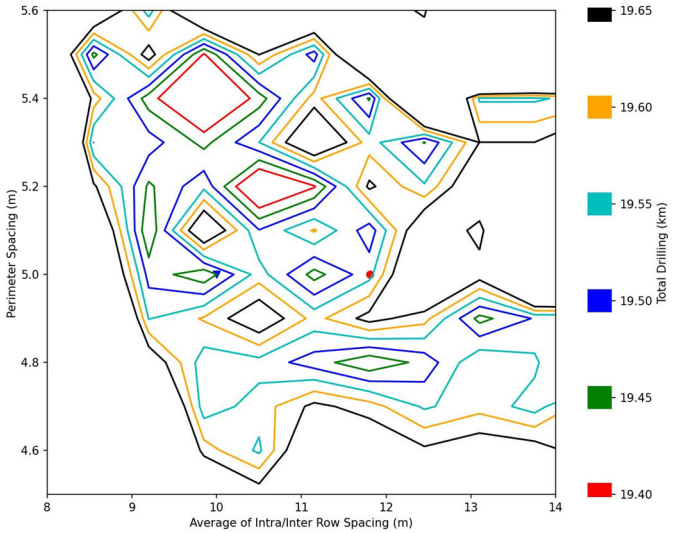


Fig. 15. Km of required drilling contour plot for Example 1 (refined).

Conclusions and recommendations

Heuristic simulation-based methods for automatically configuring and sizing vertical borehole ground heat exchanger fields are presented in this paper. These methods have been implemented in an open-access software tool, GHEDesigner. The methods as implemented in GHEDesigner allow for the design of highly irregular borehole fields that take full advantage of the available property on site.

Use of such a tool is particularly advantageous when there is significant imbalance between the annual heat

rejection and extraction. This includes a 23% reduction in the most extreme case from the BUPCRS algorithm. For the most balanced field (a 1.1X ratio between the heat rejection and extraction), the tool provided a 1% reduction. The RW-Opt and RW-ISch procedures produce reasonable fields with little to no user-adjustment.

The rotation and spacing domain investigations provided insights into the effectiveness of the rotation exhaustive search heuristics and RW-Opt procedures. The rotation investigation revealed that both the RW-SpcInvlMax and RW-SpcInvlMin algorithms consistently produced fields within 2% of the RW-SpcInvlBest algorithm. Considering the large reduction in computation time by not performing thermal simulations for every field corresponding to a rotation, the maximum and minimum NBH heuristics are deemed acceptable. However, an exhaustive search could be applied for further refinement. The spacing investigation showed that the default optimization parameters produced borefields with total drilling amounts within 2% of the best fields found with an exhaustive search. This is also deemed acceptable but could be further improved by adjusting the optimization parameters to fit a specific problem (adjusting the domain boundaries for example).

The RowWise algorithms rely on heuristic methodologies to place the boreholes; there is little doubt that many aspects of the RowWise algorithms could be improved. Optimizing the position of every borehole for large fields with hundreds of boreholes is currently infeasible for design practice. But possible improvements to the existing algorithms include:

- Improved method for configuring the borefield. Other parameterizations of the problem are certainly possible and may lead to better optimized solutions.
- Improvement of the current parameterization. The current borefield generation tool has some quirks that can lead to inconsistent behavior when adjusting inputs for some edge cases. These edge cases could be fixed, and the generation tool could undergo further polishing to make it more consistent (which would have a positive effect on optimization procedures).
- An optimization algorithm more suited to domains with many local minima could also provide better and more consistent optimization performance with the current parameterization.
- Incorporation of heat pump models could improve the accuracy and remove the need to estimate the hourly heat extraction/rejection.
- Improvement to borehole design other than topographical layout (the addition of tilted boreholes, zoning to support different depth of boreholes, etc.) are possibilities that need further investigation.

Nomenclature

B = spacing between boreholes, (m)
D = the domain of a GHEDesigner search algorithm (consists of a sequence of sets of ordered pairs). These domains are usually unimodal (w/respect to NBH)
EFT_{max} = maximum allowable heat pump entering fluid temperature, (°C)
EFT_{min} = minimum allowable heat pump entering fluid temperature, (°C)
EFT_{sim} = list of all heat pump entering fluid temperatures throughout a simulation, (°C)
ET = excess temperature – the temperature difference by which the maximum or minimum EFT exceeds the design temperature constraints, (°C)
H = the "height" of a vertex relative to a reference axis. Used by the RowWise placement algorithm, (m)
L = the length of a piece of geometry (either a square or rectangle), (m)
N = maximum number of boreholes that can be placed along a line segment given its length and the minimum spacing between the boreholes
NG = the set of all ordered pairs outside of a given set of no-go zones
p = an ordered pair representing the Cartesian coordinates: (x, y)
PB = the set of all ordered pairs inside of a property boundary.
$penalty$ = penalty value used in an objective function, (m)
R = set of ordered pairs representing a rectangle of bi-uniformly spaced Cartesian coordinates
Rf = orientation of rows counter-clockwise from horizontal (- or °)
S_{inter} = minimum spacing between rows of boreholes in a RowWise borefield, (m)
S_{intra} = minimum spacing between boreholes in a row, (m)
S_p = minimum spacing between boreholes along the perimeter of a property/no-go zone in a RowWise borefield, (m)
S_{Ratio} = ratio between S_p and S_{target} [-]
S_{target} = minimum spacing that refers to both S_{intra} and S_{inter} when used, (m)
BHE = borehole heat exchanger
$BUPCRS$ = bi-uniform polygonal constrained rectangular search
EFT = heat pump entering fluid temperature, (°C)
$ExFT$ = GHE exiting fluid temperature, (°C)
NBH = the number of boreholes in a borefield
GHE = ground heat exchanger
$GSHP$ = ground source heat pump
$HVAC$ = heating, ventilation, and air conditioning
LTS = long time step
$RW\text{-}ExhSchMin$ = RowWise exhaustive search procedure with the minimum NBH heuristic
$RW\text{-}ISchMax$ = RowWise search procedure with the independent perimeter placement feature enabled and the maximum NBH heuristic

RW-ISchMin = RowWise search procedure with the independent perimeter placement search with the minimum NBH heuristic
RW-OptMax = RowWise optimization procedure with the maximum NBH heuristic
RW-OptMin = RowWise optimization procedure with the minimum NBH heuristic
RW-SchMax = RowWise search procedure with the maximum NBH heuristic
RW-SchMin = RowWise search procedure with the minimum NBH heuristic
RW-SpcInvlBest = RowWise interval search procedure with an exhaustive field orientation search
RW-SpcInvlMax = RowWise interval search procedure with the maximum NBH heuristic
RW-SpcInvlMin = RowWise interval search procedure with the minimum NBH heuristic
VBGHE = vertical borehole ground heat exchanger

Subscripts

adj = short for "adjusted" clarifies that variable has been adjusted relative to its original value
i = signifies that a variable is some type of iterator
max = signifies that a variable represents the upper limit or maximum possible value
min = signifies that a variable represents the lower limit or minimum possible value
RB = signifies that a variable relates to the bi-uniform rectangle algorithm
row = signifies that a variable relates to the RowWise placement algorithm
RU = signifies that a variable relates to the uniform rectangle algorithm
sns = signifies that a variable relates to the square/near-square algorithm
x/y = signifies that a variable pertains to the x or y direction in Euclidean space

Acknowledgments

Most of the original development on the tool was done by OSU research assistant, Mr. Jack Cook and is reported in his MS thesis.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

Development of the original GHE design tool into which the RowWise algorithm was implemented was funded through US Department of Energy contract DE-AC05-00OR22725 *via* a subcontract from Oak Ridge National Laboratory. The RowWise algorithm was developed by Timothy West, who was supported by the Oklahoma State University Center for Integrated Building Systems under project 21-19.

ORCID

Timothy N. West http://orcid.org/0009-0008-2691-2047
Jeffrey D. Spitler http://orcid.org/0000-0003-0826-0512

References

Bayer, P., M. de Paly, and M. Beck. 2014. Strategic optimization of borehole heat exchanger field for seasonal geothermal heating and cooling. *Applied Energy* 136:445–53. doi: 10.1016/j.apenergy.2014.09.029

Beck, M., P. Bayer, M. de Paly, J. Hecht-Méndez, and A. Zell. 2013. Geometric arrangement and operation mode adjustment in low-enthalpy geothermal borehole fields for heating. *Energy* 49:434–43. doi: 10.1016/j.energy.2012.10.060

BLOCON 2022. "EED version 4 Earth Energy Designer Update manual" https://buildingphysics.com/manuals/EED4.pdf

Brent, R. P. 1973. *Algorithms for minimization without derivatives*, 3–4. Englewood Cliffs, NJ: Prentice-Hall. Ch.

Cimmino, M. 2018. pygfunction: An open-source toolbox for the evaluation of thermal response factors for geothermal borehole fields. *eSim 2018 – the 10th Conference of IBPSA-Canada*, 492–501. Montréal, Canada.

Cimmino, M., and M. Bernier. 2014. Effects of unequal borehole spacing on the required borehole length. *ASHRAE Transactions* 120:158–73.

Cimmino, M., J. C. Cook, and J. A. Isiordia Farrera. 2024. Optimal discretization of geothermal boreholes for the calculation of g-functions. *Science and Technology for the Built Environment* 30 (3):234–49. doi: 10.1080/23744731.2023.2295823.

Claesson, J., and G. Hellström. 2011. Multipole method to calculate borehole thermal resistances in a borehole heat exchanger. *HVAC&R Research* 17 (6):895–911. doi: 10.1080/10789669.2011.609927

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*, 222–7. Toronto (Canada).

Claesson, J., and P. Eskilson. 1988. Conductive heat extraction to a deep borehole: Thermal analyses and dimensioning rules. *Energy* 13 (6):509–27. doi: 10.1016/0360-5442(88)90005-9

Cullin, J. R., and J. D. Spitler. 2011. A computationally efficient hybrid time step methodology for simulation of ground heat exchangers. *Geothermics* 40 (2):144–56. doi: 10.1016/j.geothermics.2011.01.001.

Cook, J. C. 2021. *Development of computer programs for fast computation of G-functions and automated ground heat exchanger design*. M.S. Thesis., Oklahoma State University, Stillwater, OK.

Egidi, N., J. Giacomini, and P. Maponi. 2021. Inverse heat conduction to model and optimise a geothermal field. xarXiv Preprint.

Energy Information Administration 2022. Commercial buildings energy consumption survey (CBECS). EIA. https://www.eia.gov/consumption/commercial/data/2018/

Energy Information Administration 2023a. April 2023 monthly energy review. EIA. https://www.eia.gov/totalenergy/data/monthly/archive/00352304.pdf

Energy Information Administration 2023b. Residential energy consumption survey (RECS). EIA. https://www.eia.gov/consumption/residential/data/2020/

Gultekin, A., M. Aydin, and A. Sisman. 2019. Effects of arrangement geometry and number of boreholes on thermal interaction

coefficient of multi-borehole heat exchangers. *Applied Energy* 237:163–70. doi: 10.1016/j.apenergy.2019.01.027

Guo, M., N. Diao, K. Zhu, and Z. Fang. 2017. Optimization of ground heat exchangers in area with imbalanced heating and cooling load. *Procedia Engineering* 205:3727–34. doi: 10.1016/j.proeng.2017.10.311

Hénault, B., P. Pasquier, and M. Kummert. 2016. Financial optimization and design of hybrid ground-coupled heat pump systems. *Applied Thermal Engineering* 93:72–82. doi: 10.1016/j.applthermaleng.2015.09.088

International Organization of Standardization (ISO/TC12). ISO Standard 2019. *Quantities and units – Part 2: Mathematics* (No. 80000-2:2019). Accessed April 4, 2024. https://www.iso.org/standard/64973.html.

Liu, X., P. Hughes, K. McCabe, J. Spitler, and L. Southard. 2019. *GeoVision analysis supporting task force report: Thermal applications—geothermal heat pumps*. Oak Ridge, TN: Oak Ridge National Laboratory.

Liu, X., Y. Polsky, D. Qian, and J. Mcdonald. 2018. *Analysis of cost reduction potential of vertical bore ground heat exchanger*. Oak Ridge, TN: Oak Ridge National Laboratory.

New York State Energy Research and Development Authority (NYSERDA). 2017. *Renewable heating and cooling policy framework: Options to advance industry growth and markets in New York*. Accessed February 7, 2017. https://www.nyserda.ny.gov/-/media/Project/Nyserda/Files/Publications/PPSER/NYSERDA/RHC-Framework.pdf.

Noël, A., and M. Cimmino. 2022. Development of a topology optimization method for the design of ground heat exchangers. *Proceedings of the third International Ground Source Heat Pump Association Conference, Las Vegas, Nevada*. doi: 10.22488/okstate.22.000017

OSU (Oklahoma State University). 2016. *GLHEPro 5.0 for windows user's guide*. Accessed August 14, 2014. https://betsrg.org/s/GLHEPRO_50_Manual.pdf.

Prieto, C., and M. Cimmino. 2021. Thermal interactions in large irregular fields of geothermal boreholes: The method of equivalent boreholes. *Journal of Building Performance Simulation* 14 (4): 446–60. doi: 10.1080/19401493.2021.1968953

Robert, F., and L. Gosselin. 2014. New methodology to design ground coupled heat pump systems based on total cost minimization. *Applied Thermal Engineering* 62 (2):481–91. doi: 10.1016/j.applthermaleng.2013.08.003

Spitler, J. D. 2000. GLHEPRO – A design tool for commercial building ground loop heat exchangers. *Proceedings of the Fourth International Heat Pumps in Cold Climates Conference*, Aylmer, Québec.

Spitler, J. D. 2023. *GHEDesigner – A flexible and automatic ground heat exchanger design tool.* Accessed July 18, 2023. https://pypi.org/project/GHEDesigner/.

Spitler, J. D., J. C. Cook, and X. Liu. 2020. A preliminary investigation on the cost reduction potential of optimizing bore fields for commercial ground source heat pump systems. *Proceedings, 45th Workshop on Geothermal Reservoir Engineering*. Stanford, California, Stanford University.

Spitler, J. D., T. N. West, and X. Liu. 2022b. Ground heat exchanger design tool with RowWise placement of boreholes. *IGSHPA Research Conference Proceedings*, 53–60. Las Vegas.

Spitler, J. D., T. West, X. Liu, and I. Borshon. 2022a. Open library of g-functions for 34,321 configurations. *IGSHPA Research Conference Proceedings*, 264–271. Las Vegas, Nevada.

Spitler, J.D., J. Cook, T. West, and X. Liu. 2021. G-function library for modeling vertical bore ground heat exchanger. doi: 10.15121/1811518.

USDOE. 2022. Prototype building models. https://www.energycodes.gov/prototype-building-models.

Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, SciPy 1.0 Contributors, et al. 2020. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods* 17 (3):261–72. doi: 10.1038/s41592-020-0772-5

Xu, X., and J. D. Spitler. 2006. Modeling of vertical ground loop heat exchangers with variable convective resistance and thermal mass of the fluid. *Proceedings of the 10th International Conference on Thermal Energy Storage*. Pomona, NJ.