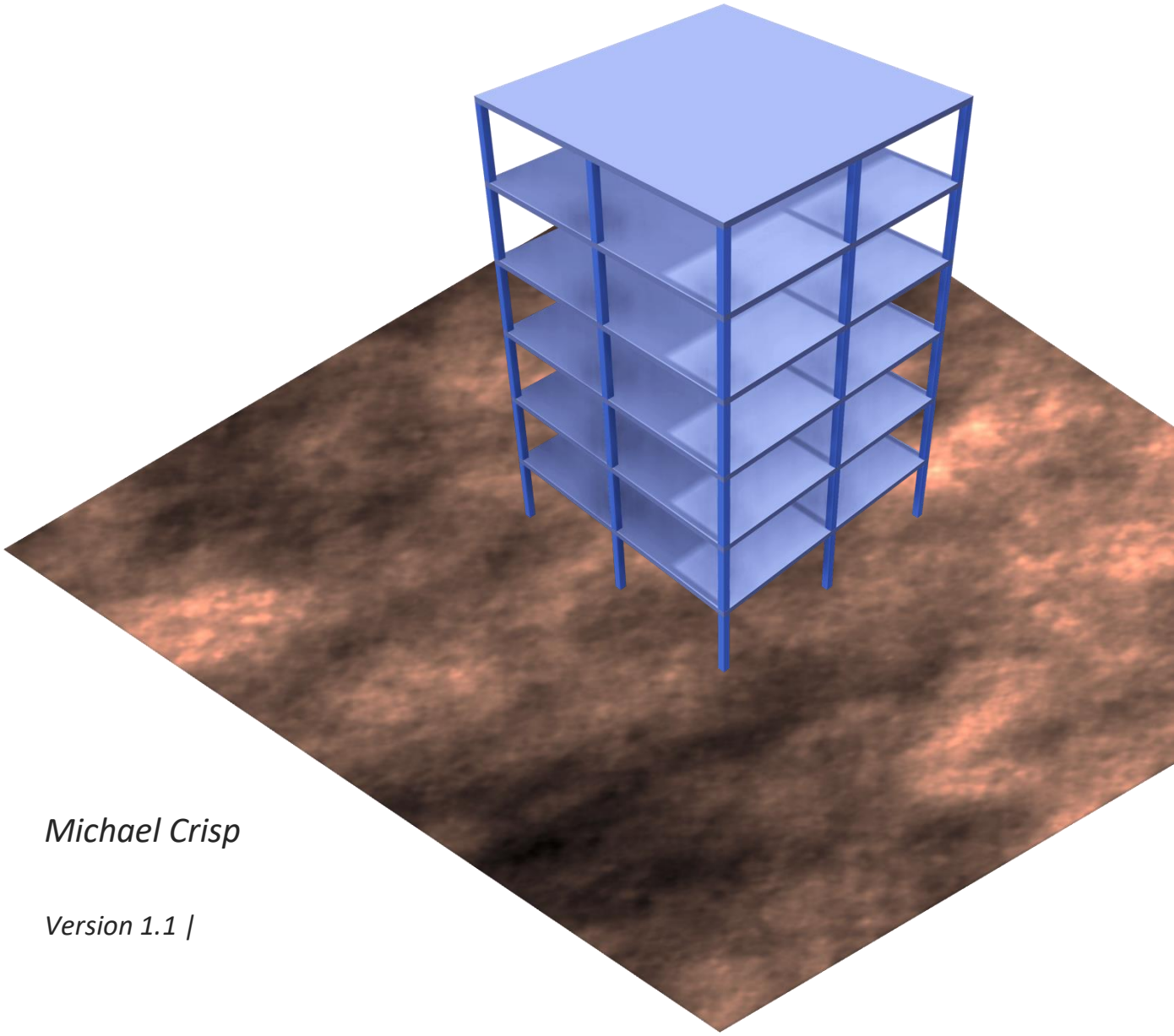
SIOPS USER MANUAL

Site Investigation Optimization for Piles using Statistics



Michael Crisp

Version 1.1 |

Table of Contents

SIOPS Manual v1.1

1 About

SIOPS (Site Investigation Optimisation for Piles using Statistics) is a program that aids in the planning and optimisation of geotechnical site investigations for pile foundations. It is open-source (written in Fortran) with a text file-driven interface. It is released under the MIT licence.

The program was developed by Michael Crisp over the 2018-2020 period as part of his PhD on site investigation optimisation. While this was not the primary software used for the research, it was written towards the end of the PhD with the explicit intention of being highly optimised for speed, and being relatively straightforward to use in an industry setting, while still being robust and versatile.

This manual can be thought of as 2 overall sections, where the first half gives an overview of the general theory and inner workings of the program, while the second half describes the program input and output.

Any questions and comments regarding the SIOPS software or this manual can be directed to the author at michael.p.crisp@gmail.com.

2 Papers based on SIOPS results

Various versions of this software have been used to generate site investigation performance data used in the following papers:

Crisp et al. (2019d): Towards Optimal Site Investigations for Generalized Structural Configurations

3 Theoretical Overview

3.1.1 The Optimisation Framework at a Glance

SIOPS is capable of generating a wide variety of information about site investigation performance, including the mean and variability of the designed foundation sizes, resulting differential settlements, probability of failure, and total project cost. The user defines attributes for the soil, piles, building and site investigations that they would like to consider.

Performance of a single investigation is assessed through 4 general steps:

1. Generate a virtual soil.
2. Do a site investigation.
3. Design the foundation according to the Site Investigation (the SI stage).
4. Get true foundation performance using the original, full virtual soil (CK stage) – Complete Knowledge of soil.

The steps 2-4 can be repeated for different scopes of investigation. By comparing the resulting performance of the different investigations, it is possible to see which is optimal as well as the relative improvement gained by using the optimal investigation. The overall procedure is elaborated in Figure 1.

The statistical nature of the framework comes from Monte Carlo analysis. Rather than performing the above steps on a single virtual soil, thousands of different, equally likely random soils are generated. Each soil is described by a set of statistics/inputs which can be matched to a real soil found in practice. For a given investigation, it is the variability of the soil across different realisations that produces the variability in the results.

Generally speaking, the aim is to minimise the expected value (the average) of the chosen metric across the Monte Carlo realisations. It is important to note that while the expected value is the best estimate, it is not intended as a prediction. For example, the overall cheapest investigation should save money in the long run over a series of many projects. However, if a result is applied to any individual project in the real world, then the consequences of that particular instance may be better or worse than this program suggests.

As with any software, there are a number of approximations and assumptions being employed, and an engineer should always incorporate expert judgement and case-specific factors into their investigation. This program is simply intended as one tool in a large toolbox for site investigation planning.

SIOPS has three different modes for assessing and comparing investigations:

1. Fixed mode. This assesses a set of investigations as defined by the user.
2. Heatmap mode. This generates a heatmap of investigation quality over a given area, informing the best testing locations for a single borehole.
3. Evolutionary mode. This employs a genetic algorithm to iteratively improve and optimise borehole locations.

Furthermore, there are two broad soil classes available; single-layer soils with variable soil properties and multiple-layer soils with variable layer boundaries but uniform properties within each layer. The latter is generally recommended, as users can more explicitly model their site, and there is no explicit pre-processing required, unlike that extensively done for the former class. It should be noted that Figure 1 describes the procedure for the single layer class. The multiple layer class follows a similar, albeit simplified process; designing piles and determining settlement directly as opposed to through the use of pre-generated curves.

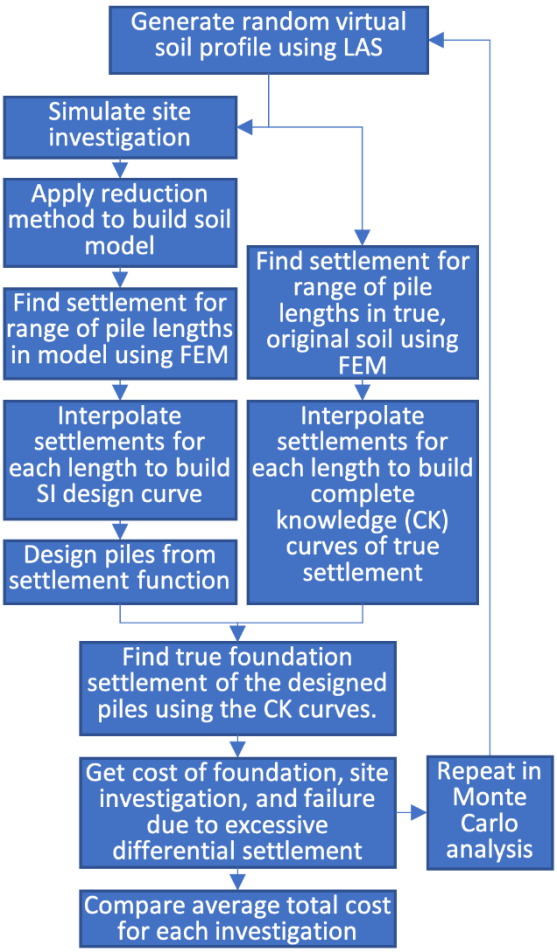


Figure 1: Flow chart showing the full site investigation assessment procedure.

3.2 Assumptions and Limitations

3.2.1 Elastic Settlement

The SIOPS program assesses pile performance in terms of linear-elastic settlement. While a detailed explanation of this decision can be found in Crisp et al. (2019a), the main reasons are as follows:

1. A numerically efficient means of determining pile performance is required due to the large number of piles being analysed within the Monte Carlo simulation. This is especially true when using the Genetic Algorithm. More sophisticated models are computationally intensive to the point of being infeasible.
2. In addition to the settlement method itself being computationally faster, the single-layer soil mode employs additional optimisations that rely on the linear-elastic nature.
3. The loads and resulting deformations involved in pile settlement calculations are typically low enough that soil mechanical behaviour is largely in the elastic region.
4. Finally, the linear-elastic assumption is constant across the full performance assessment; both in the true soil settlement and soil model settlement. As a result, any potential model errors are largely self-cancelling.

3.2.2 Structural Deformation

Furthermore, there are some key assumptions in how buildings deform. This impacts the calculations of differential settlement on which investigation performance is based. As discussed in point (4) above, these assumptions are consistent across both stages of pile assessment, limiting their impact on the results.

1. Each pile is treated independently, therefore any additional settlement caused by the proximity of other piles is not considered. As SIOPS is intended to work for multi-story office buildings constructed with individual piles, as opposed to pile groups, the centre-centre spacing is typically large enough for this not to be an issue.
2. Building rigidity is not considered. SIOPS assumes that the loading on each pile does not change with time, and that the building weight is fully supported by the piles. Therefore, SIOPS does not account for moment redistribution as differential settlement occurs, nor any additional resistance from the building resting on the ground surface.

3.2.3 Soil Geometry and Site Investigations

The soil is represented as a rectangular prism. The surface is completely flat and horizontal. Furthermore, the bottom of the soil (also flat and horizontal) is considered as perfectly-rigid rock.

3.3 Soil Representation

3.3.1 Virtual soils

Virtual soils are numerical representations of soil properties over a volume. For example, a 3D grid of discrete elements, where each element has a single set of material properties. These soils are randomly generated within SIOPS, meaning that a single set of input parameters can produce vastly different realisations of soils, which are all described by the same statistics.

3.3.2 Random Field Parameters

Each random field can be statistically represented by 3 parameters.

1. The mean. **Unit:** MPa
2. The standard deviation. Here, it is normalised (divided) by the mean to form the coefficient of variation (COV). **Unit:** Dimensionless.
3. The scale of fluctuation (SOF). This is also known as autocorrelation, and is analogous to the *range* parameter in geostatistics, as it describes the distance over which soil properties are highly correlated. In practice, a high SOF value is more likely to result in large pockets of similar material. **Unit:** Metres.
    a. Note that the SOF can be specified independently for the horizontal and vertical directions, with the former typically being larger than the latter. When these values are the same, the soil is said to be isotropic (which is arguably the worst case ratio), otherwise the soil is anisotropic.

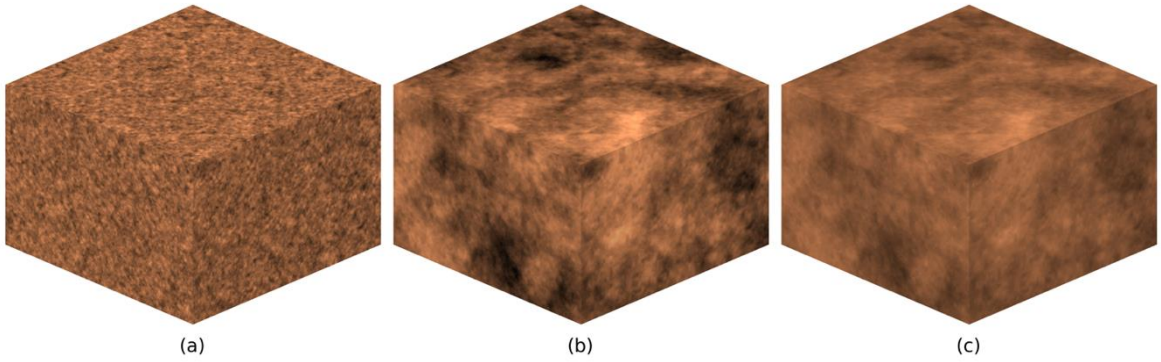


Figure 2: Impact of SOF and COV on the resulting single-layer virtual soils. (a) COV 80%, SOF 1m, (b) COV 80%, SOF 16m, (c) COV 40%, SOF 16m.

3.3.3 Generation with Local Average Subdivision

The method used is Local Average Subdivision (LAS), which can produce soils of size $a2^n \times b2^n \times c2^n$, where a, b, c and n are integers. The method works, as the name suggests, by subdividing the soil in each dimension across multiple stages, doubling its resolution at each stage while maintaining the desired statistical properties. The original stage 0 field is of size $a \times b \times c$ elements, and is generated by covariance matrix decomposition. LAS is not well suited to generating anisotropic soils, but is otherwise an efficient and accurate method.

The method was developed by Fenton and Vanmarcke (1990).

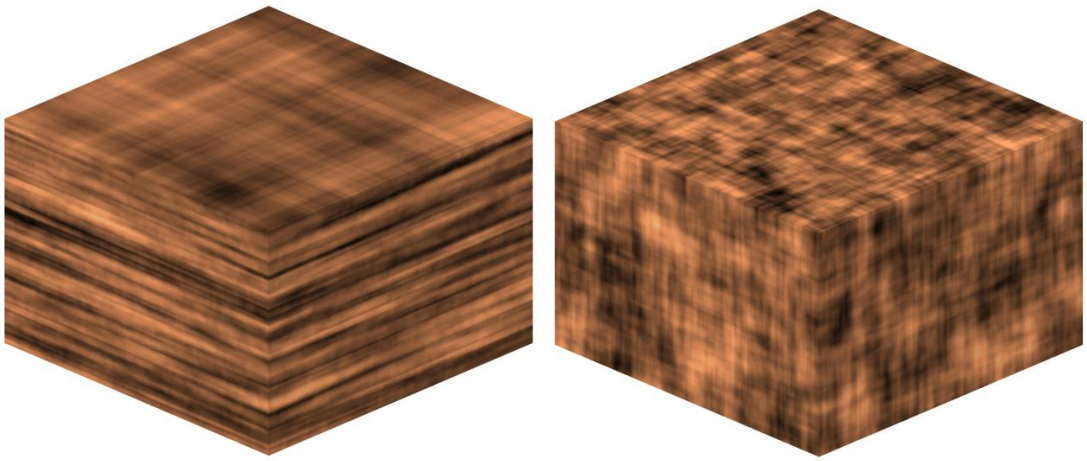
3.3.4 Generation with the Piecewise Covariance Matrix Decomposition method

New in version 1.1

An alternative means of generating random fields has been added; a piecewise implementation of the covariance matrix decomposition (CMD) method (Xiao et al. 2019). While the CMD method has existed for quite some time, it has been limited by the extremely high RAM requirements for anything other than very small fields. Furthermore, the method would tend to fail for such fields due to the accumulation of rounding errors.

The piecewise implementation, taken from theory developed Li et al. (2019)., serves to significantly reduce both the RAM and computational time requirements to generate fields compared to the standard CMD method. This is achieved through effectively applying the correlations over slices of the soil in each direction through a stepwise process, rather than across the whole soil in a single calculation. The fields can be of arbitrary size, and either 1, 2, or 3 dimensions as implemented.

The advantage over LAS is that it is capable of generating anisotropic fields, as seen in Figure 3(a). The disadvantage is that the fields are slightly slower to generate and also create a streaked appearance as seen in Figure 3(b). However, the streaked appearance does not affect the overall statistics of the soil.



(a) (b)

Figure 3: Fields generated by the piecewise covariance matrix decomposition method; (a) an anisotropic soil with horizontal and vertical SOFs of 100 m and 1m, (b) an isotropic soil with a SOF of 10 m.

3.3.5 Property Distributions

Single-layer, 3D soils are represented by lognormally distributed random values. This distribution is used because studies indicate it is a reasonable fit for soils found in practice. Perhaps more importantly, it ensures that all properties are non-negative.

In the case of multiple layers which incorporate random 2D layer boundaries, each boundary is represented by a normal distribution.

The SOF, which statistically describes the spatial distribution of properties, is implemented using an exponential Markov correlation model, which has been found to be the most suitable for describing soils.

3.4 Soil Modes

There are two different soil modes available:

- A single-layer, variable soil.
- A multiple-layer soil with uniform properties in each layer and variable layer boundaries.

Each mode is a simplification of soils found in practice; for example, there is no mode with variable layer boundaries and variable soil within each layer. This limitation exists because each of each of the above methods takes advantage of a different set of incompatible optimisations. To combine features of both would require the use of, at a minimum, thousands of finite element analysis simulations; something that is not feasible without a supercomputer if results are desired within a practical timeframe.

Note that it has found that total uncertainty in complex soil situations tends to be greater than the sum of individual uncertainties due to layer boundary variability and the spatial variability of soil properties within layers. Therefore, the user may wish to run both modes for their site, then add the failure costs together.

3.4.1 Single Layer Methodology

This section describes aspects of the methodology that are unique to the single-layer soil mode. This includes the representation of virtual soils, as well as the method of determining pile settlement.

3.4.1.1 Soil generation overview

The virtual soils used in the single-layer mode are 3D random fields produced by LAS. Therefore, they are generated as described in §3.3.3, and appear as shown in Figure 2. The properties vary randomly and continuously with distance. However, when viewed at a sufficiently large scale, the soil as a whole has a constant average. This is in contrast with multiple-layer soils where each layer has a distinct average. Any errors between the soil model and true soil therefore result from either an incorrect/insufficiently conservative estimation of this average, or through the uniform properties of the soil model being an oversimplification of reality.

3.4.1.2 Pile settlement model

Pile settlement in a random soil is assessed using the Pseudo-Incremental Energy (PIE) method (Ching et al. 2018). It is a substitute for linear-elastic finite element analysis (FEA) in Monte Carlo simulation, as it is intended to produce the same results as FEA in a fraction of the time.

PIE works to eliminate FEA from the Monte Carlo analysis, instead conducting FEA once in a deterministic soil with uniform properties. PIE then obtains pile settlement in a random soil by scaling the deterministic settlement value (S_{det}) by the effective stiffness of the soil (E_{eff}).

$$S = \frac{S_{det}}{E_{eff}}$$

E_{eff} is determined as a weighted geometric average of Young's modulus properties, where the weights are also derived from the aforementioned deterministic FEA simulation, using the resulting stress and strain. The distribution of weight values has been shown for a single 0.5m square pile of various lengths in Figure 4. Both plan-view and cross-sectional views are shown, where the former is at the surface, and the latter is taken at the pile location. Because the centrally located pile is rigid, a weight of zero is associated with soil within its volume. As seen in Figure 4, a different set of soil weights are needed for every pile length increment.

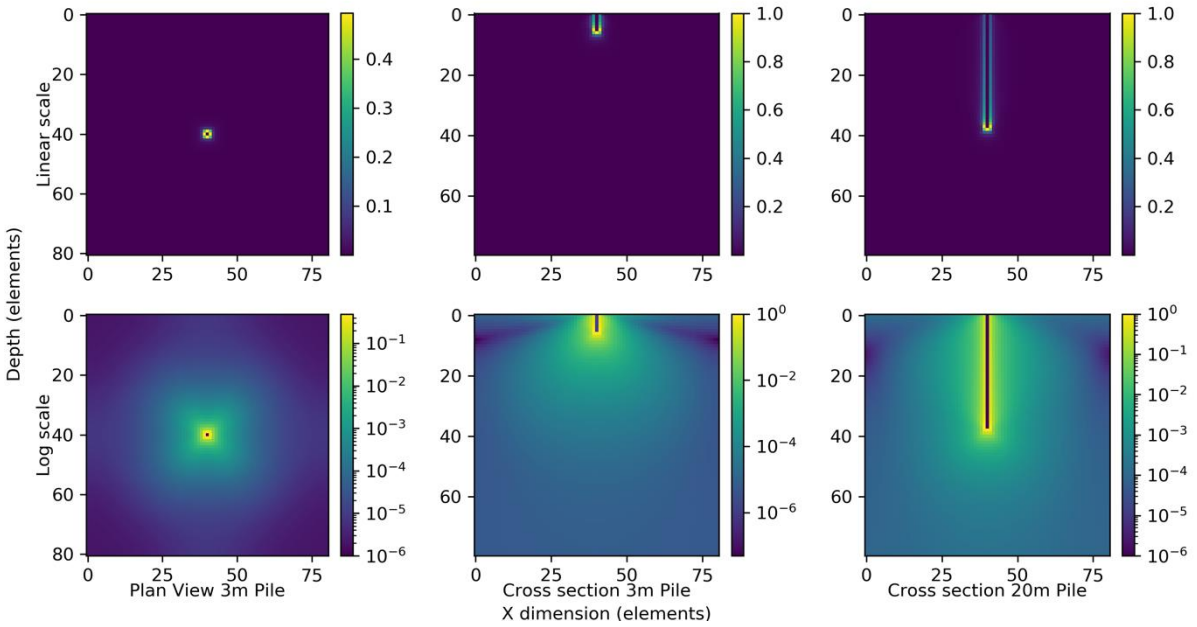


Figure 4: Distribution of soil weights for a 3m and 20m pile in plan and cross-section views.

The weight *W* of a single element is given by:

$$W = \Delta\sigma_x\Delta\varepsilon_x + \Delta\sigma_y\Delta\varepsilon_y + \Delta\sigma_z\Delta\varepsilon_z + \Delta\tau_{xy}\Delta\varepsilon_{xy} + \Delta\tau_{yz}\Delta\varepsilon_{yz} + \Delta\tau_{xz}\Delta\varepsilon_{xz}$$

While the PIE method is not perfectly accurate, the accuracy has been deemed adequate for the purposes of the analysis (Crisp et al. 2019a). This is especially true given the several-orders-of-magnitude speedup.

3.4.1.3 Pre-processing

The single-layer mode requires 2 stages of pre-processing before a site investigation analysis can be conducted, as seen in Table 1. The first stage is used to generate the PIE-related information, while the second stage uses the PIE data to calculate true pile settlement across all random soils in the Monte Carlo simulation. As a result, stage 1 is only needed infrequently, if the pile width changes or the maximum pile length is increased. However, for every new soil case examined, specifically each new combination of SOF and COV, stage 2 will need to be regenerated.

Table 1: Stages of single layer pre-processing.

Stage	Description	Frequency needed
1	Generation of soil weights and deterministic settlement	Once per pile width
2	Generation of true pile settlement in variable soil	Once per pile width, per soil

Rather than having to specify each stage of processing manually, SIOPS has been programmed to automatically detect what data is needed, and to generate the required data appropriately. However, while this pre-processing will result in SIOPS taking a long time to run initially, subsequent simulation using the same pile size and soil conditions will be considerably faster. The site investigation analysis itself generally takes under a minute to run when using recommended settings.

3.4.1.4 Pile Design and Settlement Assessment

A graphical representation of both the pile design and settlement determination processes are given in Figure 5. The stage 2 pre-processing, discussed in the previous section, involves generating a curve of normalised pile settlement with length, in terms of discrete values at 1 m length intervals. The curve is normalised in that it is associated with unit soil stiffness, and unit applied vertical load. Each individual, discrete settlement value is then scaled using the PIE method, and by applying the desired vertical load, as imparted by the structure. This true settlement curve is termed a “CK curve”, as it uses the complete soil knowledge.

By comparison, the pile design itself is undertaken using an “SI curve”, which represents the pile performance in a soil model constructed from a site investigation. As single-layer soil models consist of a single, uniform value of Young’s modulus, the SI curve is determined by scaling the full deterministic settlement curve by this value. This explains why the SI curve is smooth and consistent, while the CK curve can be somewhat undulating. Both of these processes are only possible by assuming linear-elastic properties – that the settlement scales linearly with both soil stiffness and applied load.

The curves are made continuous though interpolation using the Akima method (Akima 1970), which can be thought of as a cubic spline constructed in a way that’s designed to avoid unwanted artefacts such as overshooting. Note that pile design is rounded up to the nearest 0.1 m.

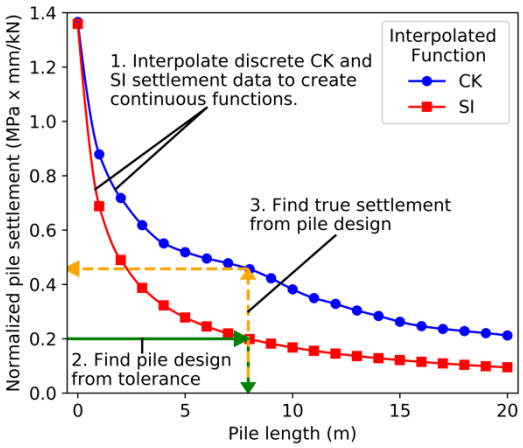


Figure 5: The pile design process and method of determining true settlement.

3.4.1.5 Pile considerations

The pile is treated as rigid, and is perfectly bonded to the surrounding soil. The distance between the pile and the edge of the soil must be larger than five times the pile diameter, and is otherwise equal to the maximum pile length. This applies both horizontally, and vertically below the pile. Furthermore, the pile is treated as a rectangular (usually square) prism.

The cross-sectional pile area must coincide in shape with soil elements. Therefore, to adjust pile sizes, different numbers of soil elements can be covered, or the size of the elements can be changed.

3.4.2 Multiple Layer Methodology

This section describes aspects of the methodology that are unique to the multi-layer soil mode. This includes the generation of virtual soils, and the method of determining pile settlement.

3.4.2.1 Soil Generation Overview

Generating a multiple layer soil is a 5-step process, shown graphically in Figure 6.

1. The overall geology for a given layer is defined by a series of points. These may be points where a borehole has intersected the layer for example.
2. The layer boundary is linearly interpolated using Delaunay triangulation.
3. A 2D, normally distributed random field is generated, representing the chaotic nature of soil formation processes.
4. This random field is added to the interpolated surface to result in a boundary that resembles the original surface, with an added noise component.
5. The soil above and below the boundary is filled with the associated material properties of each layer.

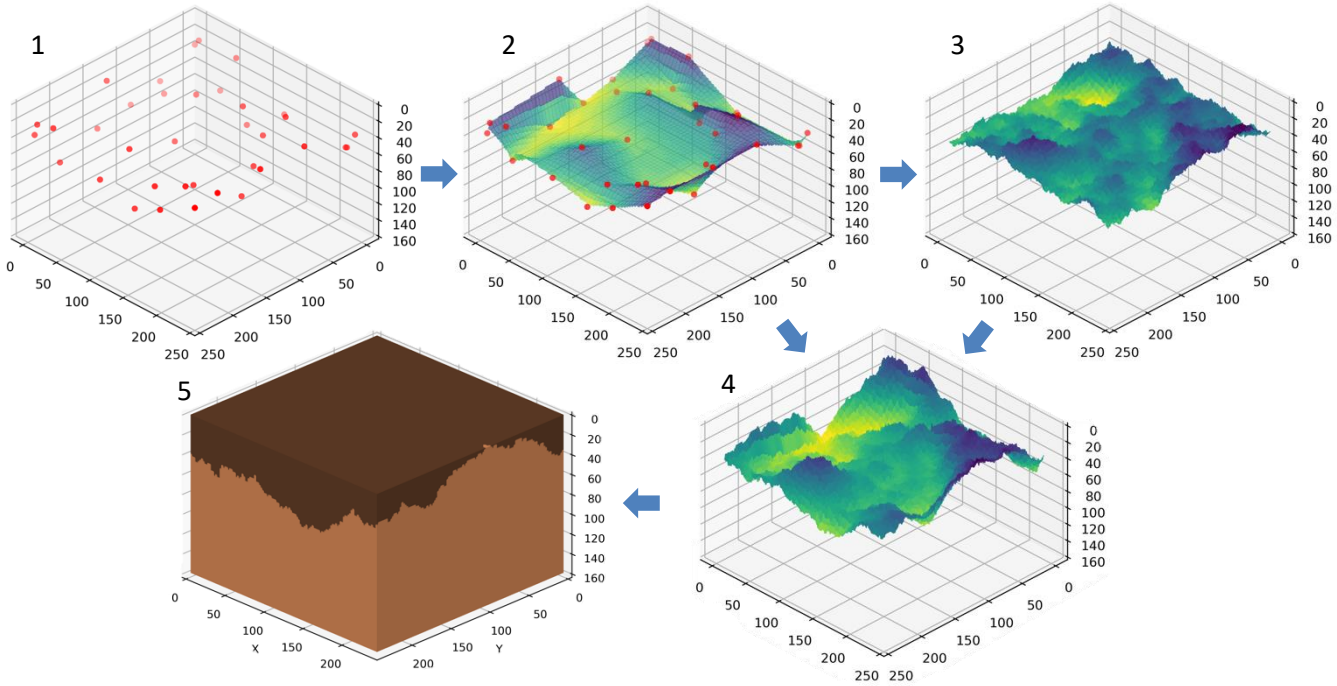


Figure 6: The 5 steps of generating a multiple-layer soil.

In the case of multiple layers which may overlap, the resulting action is inspired by the processes of erosion and deposition. For example, a layer that is newer/closer to the surface that protrudes into a deeper, older layer will replace the older layer over the intruded volume. An example soil with many layers and resulting lenses is given in Figure 7. This process is elaborated in §3.4.3.4.

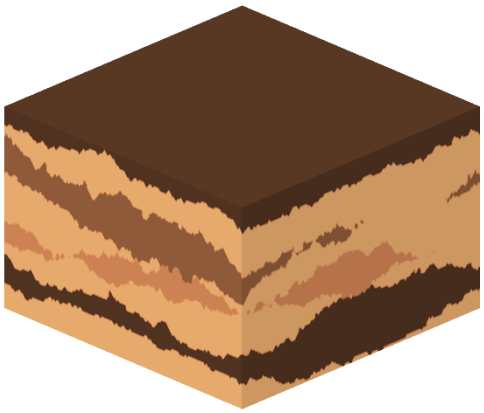


Figure 7: An example of a 7-layer soil featuring alternating layer stiffnesses.

3.4.2.2 Pile settlement model

Pile settlement is determined using the Mylonakis and Gazetas method (Mylonakis and Gazetas 1998). It can determine settlement in a 1D layered soil, where the properties within each layer are uniform. It separately assesses both shaft and base stiffness in the settlement calculation, with the former represented by distributed springs (Winker assumption). The model applies linear-elastic soil behaviour.

A limitation of the above method is that it does not account for soil below the base of the pile, which is unrealistic. Therefore, SIOPS is programmed with a modification in that the base stiffness is calculated from a weighted harmonic average of soil below the pile, as opposed to the stiffness of the layer that the pile is physically based in. A harmonic average was used as this is the appropriate choice for soil that is constant in the horizontal direction while varying in the vertical direction, as is the case here. The weighting is calculated from the integral of an exponentially decaying curve that varies with depth below the pile base, to reflect the stress variation with depth. The half-life of this decay has been set at 3 m, which generally resembles results from linear-elastic FEA under a range of tested conditions. It should be noted that minor variations of the half-life do not have a significant impact on the resulting settlement performance. Furthermore, any impact would not generally influence the results, as the behaviour would be reflected in both the SI and CK models, largely cancelling itself out.

3.4.2.3 Pile considerations

In multi-layer analysis, the pile is treated as circular. It is comprised of compressible elastic material, although is set as being several orders of magnitude stiffer than standard concrete. The base of the pile is assumed to act as a rigid circular disk.

The pile width must coincide with the width of one or more soil elements. Therefore to adjust pile sizes, different numbers of soil elements can be covered, or the size of the elements can be changed.

3.4.3 Treatment of Layer Boundaries in Soil Model

While this section falls under the scope of the previous subsection, 3.4.2, it has been made its own subsection owing to both its size and its importance. It is vital that engineers understand how the layers are being applied so that results can be interpreted properly.

3.4.3.1 Interpreting depths at boreholes

The layer boundary depths, at borehole locations are known exactly. Depth uncertainty, due to soil material ambiguity, is not incorporated in the analysis. As such, if the soil is consistently sampled with a continuous test type, then the layer would be recreated exactly in the 3D soil model. In the case of discrete tests, such as the SPT, when the borehole encounters the 2nd layer, the layer depth is interpreted as being mid-way between the first sample taken from that layer, and the previous, higher-up sample. In other words, when a change of layer is detected, it is assumed to be at the average distance between samples where the change occurred. As such, the maximum deviation the true layer can have from the interpreted depth is 0.75 m, if the sample depth spacing is 1.5 m.

Layers that exist below an arbitrary borehole, and are therefore not encountered, are placed just below the bottom of that borehole. If a layer is below all boreholes, then that layer's upper boundary is instead placed at the bottom of the soil, effectively eliminating it from the soil model.

If a layer does not exist at a particular borehole, if for example it has been eroded away, then that layer is given a thickness of zero at the borehole. Therefore, the layer will become increasingly thin as it approaches this borehole from other locations where the layer is encountered. In reality, if a layer is found to cease between two boreholes, then the layer would be assumed to stop at the lateral midpoint of those boreholes. However, while this is a simple matter in the 1D case, it is difficult to extend this concept to the 2D layer boundaries considered in SIOPS, while accounting for arbitrarily scattered borehole locations.

3.4.3.2 Recreating layer boundary surface from boreholes

This section deals with the manner of interpolating and extrapolating layer surfaces from borehole information, which depends on both the number and configuration of boreholes. Four types of cases are presented in Figure 14 and described below.

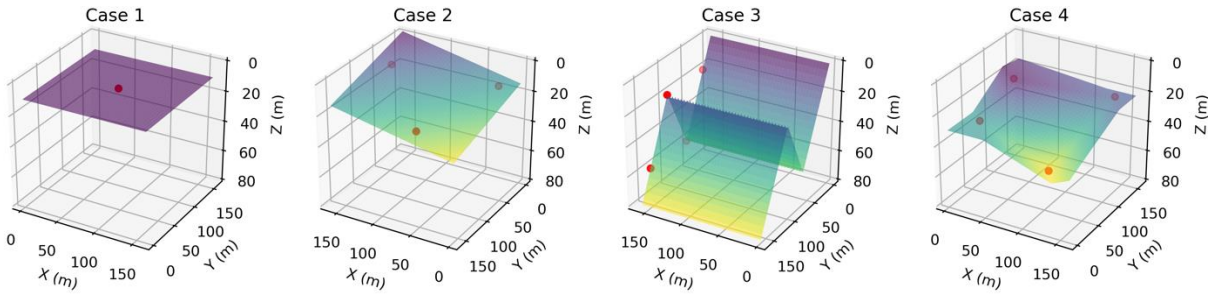


Figure 8: Different cases of creating a layer surface from borehole information.

Cases:

1. Single borehole. The entire surface is a constant depth at the borehole's intersection with the layer.
2. Three boreholes. These three points define a flat plane which is interpolated and extrapolated accordingly.
3. Colinear boreholes. In the case of two boreholes, or any number of boreholes that form a straight line, the surface is linearly interpolated between the points. The height is constant when moving 90 degrees from the line's direction. Similarly, the surface is linearly extrapolated from the two outer-most points at either end of the row. With two points, this results in a flat plane.
4. All other cases. With four or more boreholes not in a straight line, the surface is interpolated using Delaunay triangulation. Extrapolation is performed using a custom technique that attempts to approximate the surface expanding horizontally away from the outer boundary of points, discussed in the next section.

The Delaunay triangulation (Delaunay 1934) interpolation works by generating a continuous surface of triangles, where the vertices of these triangles are defined in 3D space by the borehole locations and layer depths encountered by the boreholes. The Delaunay method serves to minimize the incidence of thin triangles, by attempting to maximize the smallest interior angles. Each triangle is then treated as a flat plane.

3.4.3.3 *Extrapolating scattered points*

This subsection describes the extrapolation technique mentioned in case (4) in the previous subsection. The data from the case (4) example has been used to demonstrate the extrapolation technique in Figure 9. Note that two interior points have been added which weren't present in Figure 8, for the purpose of distinguishing between interior and boundary points. However, these interior points do not impact the extrapolation.

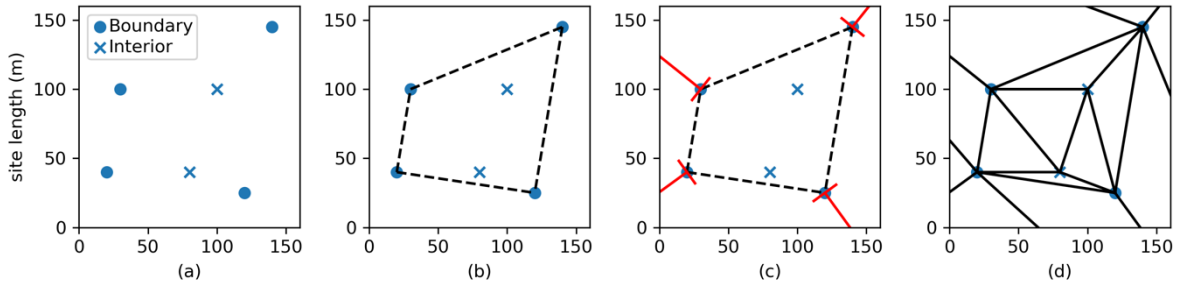


Figure 9: Process of extrapolating scattered data.

The steps are as follows:

a. A set of layer depths are defined by a site investigation. In this case, 4 boundary and 2 interior points, for a total of 6.
b. A boundary polygon is defined by the outer-most points.
c. The polygon's tangents are calculated at each point, where an additional point is created at a large offset at 90 degrees from each tangent. The height at each new point is equal to that at the corresponding vertex. This means that the layer depth is horizontal along each red line.
d. The final set of points is then interpolated using Delaunay triangulation. This technique is equivalent to expanding the boundary to cover the site, then undertaking regular interpolation.

This is not guaranteed to always provide the best extrapolation. For example, the lower interior point in Figure 9 is close to the boundary, and could arguably be used for extrapolation. However, the current method is deemed to be reasonable, and more importantly; consistent and robust.

3.4.3.4 *Layer boundary interaction*

As mentioned in §3.4.2.1, when two layers intersect, the newer (higher) layer is dominant, overwriting the lower layer boundaries with its own. This effectively serves to erode the lower layers, dragging their surfaces downwards. This behaviour applies to both the formation of virtual soils, and the treatment of layers in the soil models derived from site investigations. Note that a layer boundary cannot exceed the upper and lower soil limits. If the limits are exceeded, most likely through extrapolation, then they are truncated at the limits.

The erosion and truncation behaviours described above are illustrated in Figure 10 for a three-layer soil, where the bottom surface of the first two layers are shown. Layer 1 (orange) erodes Layer 2 (blue), from its original green position. The orange layer is truncated at the bounds of 0 and 80 respectively.

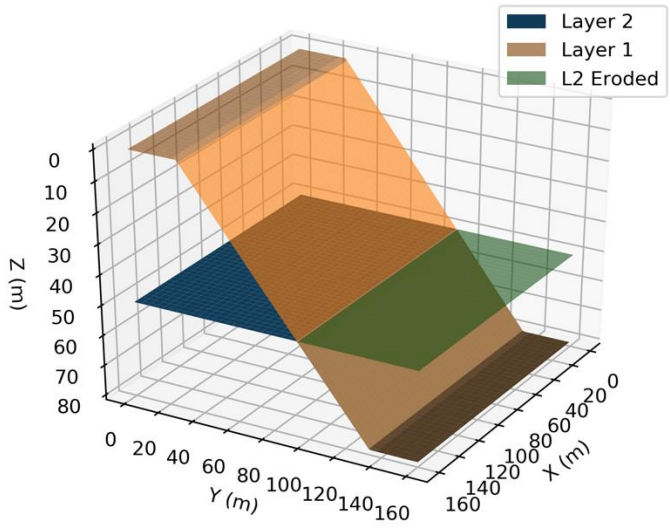


Figure 10: Example of the interaction between two layers, and the soil boundary.

3.4.3.5 Representation around pile

As mentioned in the §3.4.2.2, the settlement method requires a 1D soil profile. Therefore, the complex 2D layer depth boundary information must be simplified or transformed to a single representative depth. This transformation occurs by using a weighted average of the soil depths around the pile. When applying this simplification in the context of 2D axisymmetric FEA, the results were found to be analogous to using the full soil in 3D FEA, as discussed by Crisp et al. (2020). The weight used for each soil element is the square of the inverse distance between the element and the pile. The soil within the radius of the pile is given a zero weight. While this zero weighting is logical for layer boundaries encountered along the pile shaft, as that soil is removed and no longer exists, it is perhaps less logical for the soil layers below the pile. However, this zero weighting is applied to all layers for simplicity. The relatively small pile radius, as well as the same model being used for both design and true settlement assessment, mean that any impact the uniform zero weighting has on results is negligible. The range of this weighted average is 5 times the pile width.

3.5 Site Investigations

There are two conceptual components of site investigations as represented in SIOPS:

1. The collection of soil stiffness information (soil testing).
2. The interpretation (or reduction) of multiple stiffness samples into a single representative value. The techniques used to do this are termed “reduction methods” in the present document.

Therefore, site investigations have 5 attributes that can be specified and controlled, listed below:

1. The number of boreholes.
2. The plan-view location of each borehole.
3. The borehole depths.

4. The testing method, which dictates the following:
    a. The borehole sampling frequency with depth.
    b. Testing accuracy.
    c. Testing cost.
5. The choice of reduction method.

While the first three points are self-explanatory, the latter two are elaborated in the following sections.

3.5.1 Soil Test Types and Implementation

The tests are undertaken by extracting samples from the soil along appropriate depths at each borehole location, then applying random errors to mimic testing inaccuracy.

A set of attributes for 4 test types is given in Table 2, derived by Goldsworthy (2006). These are the standard penetration test (SPT), cone penetration test (CPT), triaxial test (TT) and flatplate dilatometer (DMT).

The testing inaccuracies are accounted for by applying a set of 3 random errors to each sample:

1. Random bias per borehole, based on each borehole's mean.
2. Random error per sample.
3. Random global bias based on the global mean. This represents the transformation error involved in converting test results to young's modulus. Therefore the TT, which obtains this value directly, has zero error for this component.

These errors are applied in the order given above and are unit-mean lognormal random variables. Note that the CPT, as a continuous test, has a frequency dictated by the size of the soil elements.

Table 2: Sampling frequency and error components for the selected test types.

Test type	Sampling interval (m)	Uncertainties measures as COV (%)		
		Transformation model	Measurement	
			Bias	Random
SPT	1.5	25	20	40
CPT	Element size	15	15	20
TT	1.5	0	20	20
DMT	1.5	10	15	15

3.5.2 Data Interpretation – Reduction Methods

This section provides theoretical information on both implementing and interpreting various reduction methods. The choice of method should be made carefully, as it has been found to have a significant impact on the resulting financial risk (Crisp et al. 2019c).

3.5.2.1 Available Methods and Implementation

The currently implemented methods are provided in Table 3, with equations for giving the representative stiffness value from a set of n soil samples (x). The representative stiffness value is referred to as the effective Young's modulus (E_{eff}). The methods are listed in order of increasing

conservatism, with the standard arithmetic average being the least conservative. The reduction method is applied independently to the samples in each soil layer with all layers having a unique E_{eff}.

There are several alternatives which are not available as options, including methods which weight samples' importance based on distance from the foundation. However, research has found that sufficiently conservative methods tend to out-perform such weighted methods, which in combination with their administrative complexity makes the latter quite impractical (Crisp et al. 2019b).

Table 3: Programmed reduction methods and their equations.

Reduction Method	Symbol	Equation for effective stiffness
Standard Arithmetic Average	SA	$\frac{1}{n}\sum_{i=1}^{n} x_i$
Geometric Average	GA	$\left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}} or \exp\left(\frac{1}{n}\sum_{i=1}^{n} log(x_i)\right)$
Harmonic Average	HA	$\frac{1}{n}\left(\sum_{i=1}^{n}\frac{1}{x_i}\right)^{-1}$
1st Quartile	1Q	$\frac{1}{4}(n+1)^{th}\ value$
Standard Deviation	SD	$\mathrm{GA}/\sigma, \text{where } \sigma = \frac{1}{n}\sum_{i=1}^{n}(\log(\mathrm{GA}) - log\ (x_i))^2$

3.5.2.2 Interpretation and Recommendations

When interpreting the various reduction methods, one must remember that the virtual soils' properties are lognormally distributed. This resembles a positively skewed normal distribution, which tends towards a small proportion of exceedingly stiff values. As such, methods which are sensitive to these outlier values, such as the arithmetic average, are highly unsuitable. In contrast, the 1st Quartile method, which is rank based like the median, is quite effective at ignoring outliers.

The geometric mean and geometric standard deviation is well suited to describing the lognormal distribution, as both of these describe multiplicative processes. For example, the geometric mean is identical to the median of lognormally distributed values. Furthermore, the geometric mean, and to a greater extent the harmonic mean, are low value weighted compared to the arithmetic mean, allowing them to disregard the high outlier values.

Another consideration is the effect of soil variability on the mechanics of foundation settlement. Pockets of soil with low stiffness have been noted as having greater influence on settlement than pockets of high stiffness. For example, Griffiths and Fenton (2009) noted that in an extreme case where soil elements were alternating between high and low stiffness in a checkerboard pattern, the settlement was greater than the stiffness' arithmetic average would suggest. Therefore, even if full site knowledge were obtained, a conservative reduction method is required to reflect this low-value weighted nature, depending on the magnitude of the soil's variability.

Current research suggests that the geometric standard deviation method is the best choice. It was found to almost uniformly provide the lowest total project cost (Crisp et al. 2019c), while also being

relatively insensitive to distance between the testing location and foundation (reference). This is because it meets the requirements of being low value dominated, as it incorporates the geometric average. Furthermore, soils with high variability have a significantly greater associated risk. Therefore, these soils should be treated more conservatively, as is done by incorporating the geometric standard deviation. In theory, the degree of conservatism should also reflect the number of samples obtained, however there is insufficient research to quantify this relationship. As such, the method consists of one geometric standard deviation below the geometric mean for the sake of simplicity.

An example comparing the different estimates of E_{eff} from the various reduction methods is given in Figure 11. In this example, the SPT globally overestimates the stiffness as a consequence of parameter transformation error or unlucky sampling locations. The potential for unconservative overestimation is why conservative reduction methods are desirable. Note that it is also possible for this error to result in an underestimation.

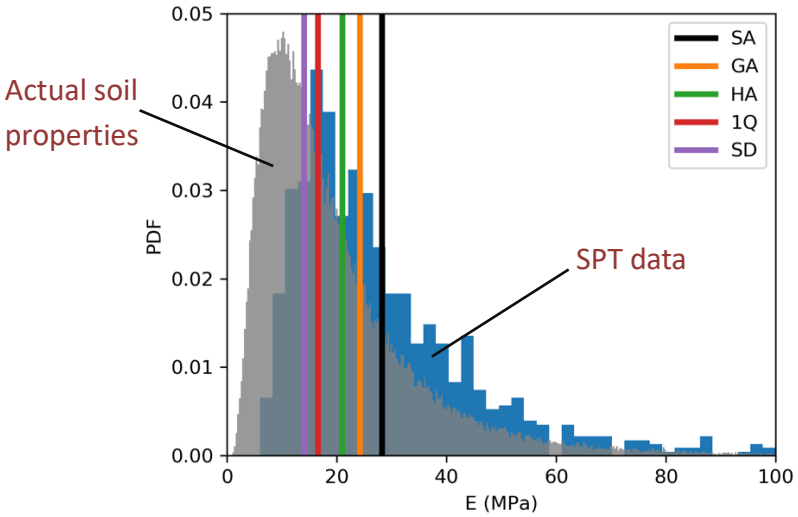


Figure 11: Comparison of E_{eff} from different reduction methods based on SPT samples which, in this instance, overestimate the soil stiffness.

3.5.2.3 Relationship with COV

It is important to understand the relationship between COV and the resulting E_{eff} for each reduction method, as the difference between E_{eff} and the true soil mean could be quite large. Regardless of the input soil mean, if the E_{eff} is too high or low, the soil model will result in infeasible pile designs. To help mitigate this, Table 4 is given to provide estimations for the average E_{eff} as a proportion of the input mean, as a function of the input COV. Therefore, it is possible to know in advance whether pile designs will be feasible.

Some variation is expected among the Monte Carlo realisations, as only a subset of the soil is tested at any given time, leading to inaccuracy. Note that the 1st Quartile method has been approximated by transforming the statistics to the normal distribution, then calculating the value corresponding to the bottom 25% based on its cumulative density function. The given equations assume (correctly) that the soil properties are log-normally-distributed.

Table 4: Approximation of the reduced representative value as a proportion of the soil mean.

Symbol	Equation
SA	1
GA	$\exp\left(\frac{-log(1+COV^2)}{2}\right)$
HA	GA^2
1Q	$\exp\left(\log(GA) - 0.675\sqrt{\log\left(1+COV^2\right)}\right)$
SD	$\frac{GA}{\exp\left(\sqrt{\log\left(1+COV^2\right)}\right)}$

3.6 Site Investigation Performance Metrics

SIOPS is programmed with a variety of performance metrics in terms of investigation quality. It is up to the user to decide which metric they prefer. The equations for each metric are given in Table 5 where n is the number of Monte Carlo realisations, $\Delta\delta_i$ is differential settlement in a given realisation, and $\Delta\delta_{tol}$ is the differential settlement failure threshold.

Total cost is arguably the best performance metric to use in practice, as it can be directly related to real-world consequences. It is defined as the best estimate of what an investigation will cost the user overall. The total cost is calculated as the sum of the average failure cost, pile construction cost and site investigation cost. The latter two costs are simply given by a unit cost ($/m) multiplied by the total lengths of piles and boreholes. Note that the average pile cost tends to be reasonably constant regardless of the number of boreholes used. This is because the instances of over-design and under-design largely cancel themselves out. Therefore, while increased boreholes greatly reduce the variation of pile length, the average itself is largely unaffected. As such, it could be excluded from the total cost trade-off.

The failure cost is a function of differential settlement, $C(\Delta\delta)$. Specifically, it is a linear function, as seen in Figure 12. The function is in terms of normalised failure cost as a proportion of the building's construction cost. As such, it is bounded by a value of 0, which represents no failure, and a value of 1, which approximates demolishing and rebuilding the structure. The original failure costs were derived from Rawlinsons (2016) and correlated with descriptions of structural damage in terms of differential settlement by Day (1999). As such, failure cost only considers cracking due to excessive differential settlement. This is as opposed to excessive total settlement, either due to bearing capacity failure or gross overestimation of the global soil stiffness.

Furthermore, there are several consequences of failure not explicitly considered, such as construction delays causing additional expenditure and/or loss of profits, as well as contract disputes and litigation. Furthermore, there are subjective consequences of failure not considered, such as damage to the engineering company's reputation, or psychological distress to the building owners due to displacement or loss of income.

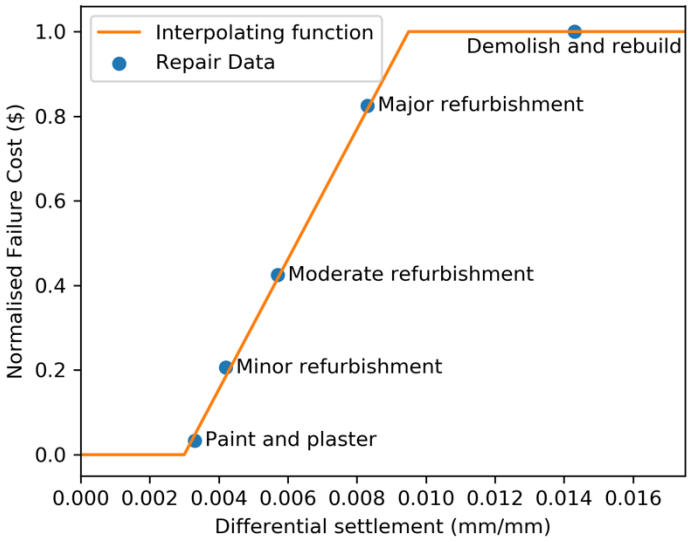


Figure 12: Bounded linear failure cost function, in terms of differential settlement, showing the original data it was derived from.

The probability of failure is defined by the proportion of realisations where the maximum differential settlement of the foundation is greater than or equal to the failure threshold. The resolution is strongly dictated by the number of Monte Carlo realisations. For example, if there are 10,000 realisations, then the precision is 0.00001. In practice, the number of Monte Carlo realisations should be several orders of magnitude higher than the desired precision in order to achieve an accurate value, as the events of interest have a low likelihood of occurring.

There is also an option for taking a certain number of geometric standard deviations above the geometric mean. This method is termed the 'geometric statistic', as it is generalized such that specifying zero standard deviations results in the geometric average. The geometric statistics are used since differential settlement approximately follows a lognormal distribution. While this is not an exact fit, at least for the purpose of calculating probability of failure, it has been deemed a suitable fit for direct comparison of investigations.

Table 5: Site investigation performance metrics.

Metric	Equation
Expected failure cost	$\frac{1}{n}\sum_{i=1}^{n} C(\Delta\delta_i)$
Total cost	Failure cost + investigation cost + pile construction cost
Average differential settlement	$\frac{1}{n}\sum_{i=1}^{n} \Delta\delta_i$
Probability of Failure	$\frac{1}{n}\sum_{i=1}^{n} \Delta\delta_i \geq \Delta\delta_{tol}$
Value Geometric standard deviations above the geometric mean (geometric statistic)	$GA \times \sigma^{Value}$, where GA and σ are calculated as per the SD method in Table 3, for differential settlement values instead of stiffness samples.

3.7 Optimising Locations with a Genetic Algorithm (Optional)

SIOPS incorporates a basic genetic algorithm (GA) which can be used to iteratively improve borehole locations. The GA employs a population of different site investigations which are identical in every manner, except that each member has a different set of borehole locations. The population is then "evolved" over a successive set of generations, through a combination of the fittest members producing offspring, and random mutation. The former mechanism (crossover) moves the population towards an optimum, while the latter keeps the population diversified to help explore the full solution space. A balanced mutation rate is required, as a low rate may converge the population into a local optimum as opposed to a global, while a high rate will prevent any convergence.

The GA used in SIOPS is adapted from code by Haupt et al. (1998), which also provides further description and background on GA's in general. The algorithm employs single point crossover and uniform mutation. While it is a real-valued GA, and the borehole locations are internally stored by their discrete element index, the conversion is handled by rounding the values to the nearest element. Furthermore, borehole locations are constrained to exist within the virtual soil. This is handled by moving any boreholes that move outside the field to its nearest location on the field's edge.

Other modifications include checking whether any two members of the population are identical. If this occurs, then one of them is randomized. This check is needed to ensure that the population is utilised to its fullest potential, as duplicates results in the processing of the same information twice, effectively reducing the population size. Duplicates could occur by pure chance, or potentially if the population has spent several generations in convergence. In the latter case, the applied randomisation serves to explore the solution space while continuing the original convergence. Similarly, a check is applied to make sure that no two boreholes in a given investigation will occupy the same space. If this occurs, then one of the boreholes is randomly scattered. This check is needed because two boreholes in the same physical space does not make sense, and because it would otherwise crash the program when undertaking the triangulation and interpolation of layer surfaces.

Note that any investigation that has more than 2/3rds invalid Monte Carlo realisations is given a worst-case performance to ensure that they don't reproduce to the next generation. This is mainly to prevent situations where 2 or 3 boreholes are located close together resulting in highly inaccurate extrapolation over most of the soil, leading to a large number of invalid realisations and effectively a random performance.

Other features have been added to GA such as elitism, dynamic mutation, and a 2nd stage GA for local optimisation. Please see §4.2.2 for details.

4 Program Input

4.1 Overview

This section describes the various options and parameters used by the program. There are 4 different input files based on overall categories, as outlined in Table 6.

Table 6: Input file descriptions.

File Name	Description
EA_input.txt	Over-arching program options and genetic algorithm parameters.
si_input.txt	Site investigation data and parameters.
pile_input.txt	Inputs related to both the structure and foundation.
soil_input.txt	All soil-related options.

Note that the above files must be in a folder labelled “input” which is located in the same directory as the SIOPS executable. Similarly, results are saved to a folder named “si_results” which must be in this directory. When single-layer soils are analysed, some pre-processed data is saved to a folder directory specified in the EA_input file as discussed later. These folders are required to exist and be in the expected locations for the program to function. Note that if the above files do not exist or are in the wrong location, SIOPS will give the user the option to generate example input files.

The following sections describe the program’s inputs, with varying levels of information depending on each input’s complexity. Potential information could include: justification, options, theoretical background, examples.

Note that each row of the tables in the following sections describes one line in the input file, except where an asterisk is included in the title, which indicates that a table of information is needed.

4.2 EA_input

This file contains both general SIOPS input, as well as that specific to the genetic algorithm (GA). See §3.7 for details on the GA.

Table 7: EA_input file line description.

Variable	Data Type	Range	Suggested
Program Run Mode	Integer	0,1,2,3	NA
Number of Monte Carlo Realisations	Integer	> 0	8000
Performance Metric to Use	Integer	>= -1	-1
Random Seeds: Virtual soil, genetic algorithm	Integer	>= 0	100,100
Single Layer Processing Mode	String	PP,CK,SI,AL	AL
Output Mode	Integer	1-4	4
Data Folder Directory	String	NA	".\data\"
Comment – leave blank			
Maximum Number of GA Generations	Integer	>= 0	100
Maximum Number of Consecutively Equal Values	Integer	>= 0	5
GA Stopping Mode	Boolean	T / F	.false.
Population Size	Integer	> 0	500
Percentage error for stopping criteria	Real	0-100	0.1
Mutation rates: Initial, minimum, maximum	Real	0-1	0.1,0.001,0.2
Mutation Mode	Integer	1,2,3	1
Fraction of Population to Keep as Parents	Real	0-1	0.5
Number of Elite Individuals	Integer	>= 0	1
Manner of Controlling Borehole Locations in GA	Integer	0,1,2	0
GA Starting Distribution	Boolean	T / F	.false.
Use 2nd Phase of GA on Optimal Solution	Boolean	T / F	.true.

4.2.1 Over-arching Options

4.2.1.1 Program Run Mode

SIOPS is programmed with three useful modes which serve different purposes, as given in Table 8.

Value 0 is a special test mode that does not perform any site investigation analysis. Rather, it quickly approximates the average pile design across all Monte Carlo realisations, for both the single layer and multi-layer inputs. This is useful because it is possible for some realisations to have invalid pile designs (required optimal pile is too short or long). If the average pile length is invalid, then this indicates that the inputs require changing; either increasing/decreasing the soil stiffness or changing the applied load on the piles.

Table 8: SIOPS run mode descriptions.

Value	Mode Name	Description
0	Test	Get approximate average pile designs to test for validity.
1	Fixed	Determine performance for a specific set of investigations
2	Heatmap	Generate a heatmap where the value at each location represents the performance of a single borehole at that location.
3	Evolutionary	Apply the GA to optimise borehole locations.

When more than one value is specified for a given investigation parameter, SIOPS will analyse all combinations of each parameter. For example, 4 boreholes, 3 test types, 2 reduction methods, and 6 borehole depths will result in 144 investigations in the *Fixed Investigation* and *Genetic Algorithm* modes. For the *Heatmap* mode, 36 investigations will be analysed, since only a single borehole is considered.

For the *Fixed Investigation* mode, the user can specify a series of investigations where boreholes are arranged in a regular grid pattern over a particular area of soil. Examples of such arrangements are shown in Figure 13. Note that prime number larger than 5 do not easily form a regular grid and will be automatically excluded. Alternatively, the user can input specific borehole locations directly. This is discussed in §4.6.1.

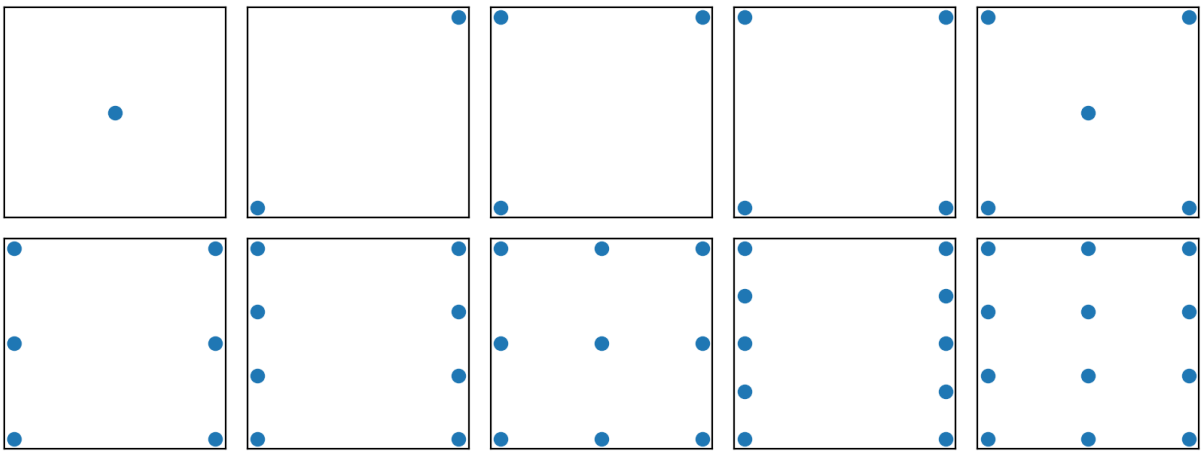


Figure 13: Examples of automatic borehole placement over a regular grid for sets of 1-12 boreholes, excluding prime numbers larger than 5.

4.2.1.2 Number of Monte Carlo Realisations

SIOPS conducts statistical analysis using Monte Carlo simulation. This option specifies the number of realisations used. Higher numbers result in greater accuracy, but also requires more computational time.

A sensitivity analysis suggests that a minimum of 6000 realisations is required, with 8000 being more preferable (Crisp et al. 2019a).

4.2.1.1 Performance Metric to Use

This parameter determines what metric is used as the objective function for the genetic algorithm and main heatmap. The values are given in Table 9, and the equations for these metrics are given in Table 5. Further descriptions of these metrics are given in §3.6.

Table 9: Choices for GA performance metric.

Value	Metric
-4	Expected failure cost
-3	Total cost
-2	Average differential settlement
-1	Probability of Failure
>= 0	*Value* Geometric standard deviations above the geometric mean (geometric statistic)

4.2.1.1 Random Seeds: Virtual soil, genetic algorithm

SIOPS extensively uses random number generators (RNGs) which produce a chain or series of pseudo-random numbers. RNGs are pseudo-random in that they are initialised by a deterministic component; an integer number referred to as the random seed. An RNG that is initialised with the same seed will always produce the same series of random numbers.

There are separate seeds specified for both the generation of virtual soils and the genetic algorithm. The former uses an explicit custom RNG, while the latter relies on Fortran's built-in random_number subroutine, for historical reasons. The latter may give different results on different systems or with different compilers.

If a seed is set to a value of "0", then it will determine the initial seed from the computer's system clock, which effectively randomises the process upon each running of the program. There is no disadvantage to leaving both seeds as a constant, positive integer.

It should be noted that the population's random initial distribution may have an impact on both the convergence speed and the quality of the final solution. Therefore if the user is comparing the processing time with different GA settings, they should run the program several times with a different integer random seed, and compare the overall timing and results of each set.

4.2.1.2 Single Layer Processing Mode

This controls the stage of pre-processing or site investigation analysis to conduct. The pre-processing stages are described in §3.4.1.3, and the corresponding values are given in Table 10. The input must be a character variable of length 2.

"AL" is recommended as it will automatically pre-process all needed information for the given inputs. Using any other input without first having run the mode under a prior stage will cause an error. Pre-processed data is saved in the nominated data folder for a given pile width, element size, soil COV and soil SOF. These changes are detected automatically, along with an increased maximum pile length or increased number of Monte Carlo realisations, which require re-processing.

Note that there are certain conditions where the user will need to manually re-process stage 2 (CK), which will over-write the previous data. For example:

- If the pile configuration or locations change without increasing the number of piles.
- If switching between the 'store soils in memory' option in §4.5.1.4.
- If changing soil attributes not stored in the file name, such as the soil size, or if changing the average layer boundary depths or stiffness ratios for multiple layers other than the first two.

Table 10: Single layer processing mode input values and description.

Value	Description
PP	Do stage 1 pre-processing and exit.
CK	Do stage 2 pre-processing and exit.
SI	Do site investigation analysis.
AL	“all” – Automatically generate the needed files then do site investigation analysis.

4.2.1.1 Output Mode

The output mode specifies the quantity of data being exported by the program, with lower numbers producing minimal, essential information, and higher values producing more in-depth and extraneous information. The effect of these values changes depending on the run mode specified; either 2 or 3. Descriptions of the output files are given in §5.

A value of 5 works with all run modes, and is intended as a debugging mode that is highly discouraged for regular use. It creates files containing all pile designs and all maximum differential settlements associated with each investigation and for each Monte Carlo realisation. Additionally, when running multiple layer mode, it will also save layer boundaries as interpolated for the soil models. Layers are only saved for the range of Monte Carlo realisations specified in §4.5.1.9, and only if the parameter in §4.4.2.1 is set to 3.

Table 11: Run mode 2 description.

Value	Description
1	Save heatmap of chosen performance metric
2	Additionally, save percentage of invalid Monte Carlo realisations
3	Additionally, save total cost, probability of failure, average differential settlement
4	Additionally, save pile construction cost, failure cost, and geometric statistic
5	Additionally, save per-case, per-realisation pile and differential settlement data.

Table 12: Run mode 3 description.

Value	Description
1	Save final, optimal information only
2	Save borehole location evolution information
3	Save full final population of borehole locations
4	Output all of the above
5	Additionally, save per-case, per-realisation pile and differential settlement data.

4.2.1.1 Data Folder Directory

The “Data” folder stores a collection of intermediate data from stage 1 and stage 2 pre-processing as described in §3.4.1.3 with regards to single layer analysis. The virtual soils are also stored here, if specified, as described in §4.5.1.9.

It is desirable to store this data in a single folder if multiple programs are to be run simultaneously, as opposed to each instance of the program having its own copy of the data. The location of this folder is specified by a string containing the absolute or relative path. For example, if the data folder

is in the same location as the SIOPS executable, then this parameter should be " '.\data\' " on windows and " './data/' " on MacOS and GNU/Linux without outer double quotation marks.

4.2.2 Genetic Algorithm Options

4.2.2.1 Maximum Number of GA Generations

As the title suggests, this value is the absolute maximum number of generations that can occur before the program stops evolving the population. This value is unlikely to be significant, as the GA will likely converge under normal stopping conditions. Therefore, a value in the order of several hundred is recommended.

4.2.2.2 Maximum Number of Consecutively Equal Values

It is difficult to determine when a global solution has been found. SIOPS attempts to find this by checking that a criterion has been met for several consecutive generations. This parameter controls this number of checking generations. Larger numbers help guarantee that a global solution has been found assuming an ideal mutation rate is chosen, however it will also extend the computational time (perhaps unnecessarily). A value of 5 or more is suggested, as it is possible for the GA to stall for a few generations before continuing to improve. The stopping criteria is elaborated in §4.2.2.5.

When the GA stopping mode is .true., then the program checks for when the optimal performance is unchanged for this number of consecutive generations, according to the specified tolerance.

When the GA stopping mode is .false., then the program checks that there has been no improvement from the historical best solution.

4.2.2.3 GA Stopping Mode

See the above section for a description of the effect.

.false. is usually recommended if elitism is set to zero and the mutation rate is sufficiently high. This is because it is possible for the best solution within a given population to degrade over time, and may therefore fail to stabilise. Otherwise, .true. and .false. have the same effect.

.false. is also highly recommended if random testing errors are applied, as an investigation's performance may vary slightly across different generations.

4.2.2.4 Population Size

In general, a larger population size may result in fewer generations required to find the optimal solution. However, individual generations will take longer to process.

Higher populations may allow smaller mutation rates. This is because the initial population will have covered more of the solution space, and will be more likely to have a member near the global optimum.

Each member of the population consists of an *x*-coordinate and *y*-coordinate per borehole. Therefore, a 4-borehole investigation has 8 components.

4.2.2.5 Percentage error for stopping criteria

This is the tolerance used to determine when a solution has been reached, as described in §4.2.2.2. This is generally in the order of 1% or less. Larger tolerances may potentially be used if the 2nd Phase GA is employed. For example, if the difference between the current and previous generations' fitness is consistently less than 1% of the latter for many generations, then the GA will exit as convergence is assumed.

4.2.2.6 Mutation rates: Initial, minimum, maximum

When a component (individual *x* or *y* coordinate) is mutated, it is randomly replaced by any coordinate along the length of the nominated area. This mutation is applied using the uniform distribution.

The mutation rate determines the proportion of the population components that are mutated in a given generation. For example, if the mutation rate is 0.5 (50%), and there is a population size of 2 with 1 borehole, then the population consists of member 1: x_1,y_1 and member 2: x_2,y_2. It is equally likely that say, x_1 and y_1 are both mutated (in which case member 1 is completely randomised), and that x_1 and x_2 are both mutated (in which case member 1 and member 2 are half randomised).

In the case that dynamic mutation is specified, the last two parameters of the row set the minimum and maximum mutation that may occur. See §3.7 for more details on the impact of mutation rate.

4.2.2.7 Mutation Mode

One addition to the Haupt et al. (1998) includes the option for a dynamic mutation rate, as opposed to constant.

1. Means mutation is constant.
2. Means adapt based on population fitness.
3. Means adapt based on proximity in normalised parameter space.

The literature does not present a clear argument on whether dynamic mutation offers an advantage. If dynamic mutation is specified, then after every generation SIOPS compares the fittest population member against the member with the median fitness.

If the difference in fitness is minimal, or if the members are in close proximity (similar investigation locations), then the mutation rate will increase by 50%. This is an implicit anti-crowding mechanism, where the program assumes the GA has largely converged, and attempts to increase mutation to find better solutions that are nearby.

On the other hand, if the difference is sufficiently high, then the program will assume that the solution space is adequately explored, and will reduce the mutation rate by 50% in order to promote convergence.

The equation for difference based on population fitness is given as follows:

$$Difference = \frac{Fitness_{best} - Fitness_{median}}{Fitness_{best} + Fitness_{median}}$$

The equation for difference based on normalised proximity is as follows, where the parameter values are the borehole *x*, *y* coordinates, and *npar* is the number of these coordinate components, i.e. 2 × the number of boreholes:

$$Difference = \frac{1}{npar}\sum_{1}^{npar}\sqrt{(Parameter\ Values_{Best} - Parameter\ Values_{Median})^2}$$

The difference thresholds for increasing and decreasing the mutation rate are 0.05 and 0.25 respectively.

4.2.2.8 Fraction of Population to Keep as Parents

The next generation is comprised of a combination of parents and children, the latter being various combinations of different parents. This parameter determines the proportion of parents which are used to produce offspring. For example, if the value is 0.1 (10%), then the remaining 90% will be children. Note that mutation is subsequently applied to both parents and children.

4.2.2.9 Number of Elite Individuals

Elite individuals are members of the population that continue into the next generation unchanged. In other words, they are immune from mutation. Having a small number of elites (at least 1) will guarantee that the fitness of the population will improve monotonically, or not degrade. This is because the best solution at any given time is always saved.

4.2.2.10 Manner of Controlling Borehole Locations in GA

There are options for fine control over the initial population, and the population's evolution, given in Table 13. This control is given for two reasons:

1. The optimal solutions are not truly random. Rather, boreholes are typically best placed near pile locations, or in a regular grid over the building footprint. Therefore, it is logical to have an initial population that generally reflects this tendency in order to reduce the number of GA generations to convergence.
2. It may be the case that engineers are restricted in where they may undertake an investigation, in which case boreholes must be constrained to occur within the valid investigation area.

Table 13: Options for controlling borehole distribution and evolution in the GA.

Value	Description
1	Distribute boreholes randomly, uniformly distributed over the full soil.
2	Distribute boreholes randomly, uniformly distributed over the specified site investigation area.
3	As per the above option, but additionally constrain the borehole to always remain within the site investigation area.

These options are shown graphically with a plan view of 4 boreholes and a population size of 100 in Figure 14. The black square represents the nominated site investigation area in this example. Note that the site investigation area is independent of the foundation location and size, although the area can be specified to coincide with the building if desired. The site investigation area is described in §4.3.1.1 and §4.3.1.2.

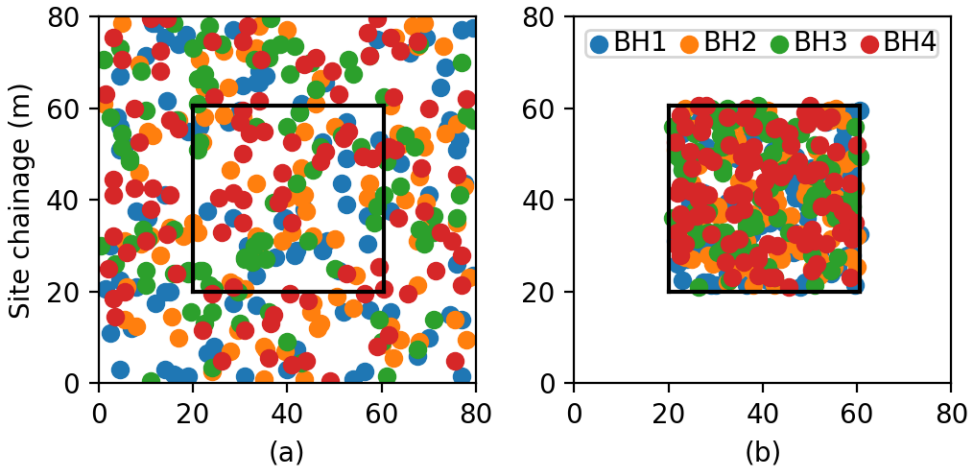


Figure 14: Demonstration of initial borehole coordinates relative to the site investigation area with 4 boreholes, using (a) value 1, (b) value 2 or 3.

Figure 14 does not distinguish different members of the population. To demonstrate this, Figure 15 shows a series of population members, where each number is the member number, and each colour corresponds to the order of the borehole, as shown in the Figure 14 legend.

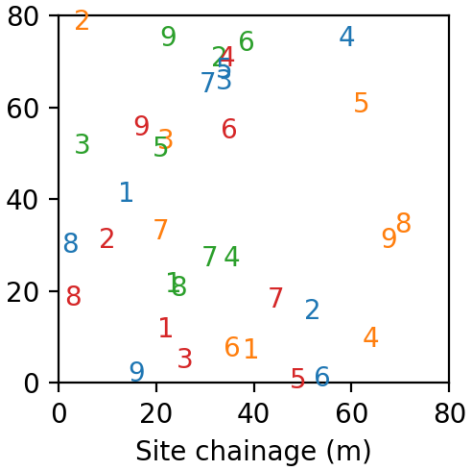


Figure 15: The first 9 instances of the population with 4 boreholes. Each number represents a generation

4.2.2.11 GA Starting Distribution

As discussed in point (1) of the previous section, the optimal borehole locations typically occur near pile locations, or otherwise over a regular grid covering the building footprint.

Setting this option to .true. causes the initial GA population to be a series of random offsets from a regular grid covering the investigation area. The offsets are normally distributed with a standard deviation equal to √2 × (investigation length × width) / 8 which is sufficient for a single centrally-located borehole to cover the full building area.

Note that if the number of boreholes is a multiple of the number of piles, then SIOPS will evenly distribute the boreholes at the pile locations as opposed to a regular grid. For example, 4 piles with 8 boreholes will lead to 2 boreholes at each pile.

This option overrides the "Manner of Controlling Borehole Locations in GA" parameter described in the previous section when it is set to 1. However, if that parameter is set to 2 or 3, then the borehole locations will be constrained to the investigation area for the initial and ongoing populations respectively.

The effect of setting this parameter to .true. can be seen in Figure 16 for a set of one and four boreholes. It can be seen that the initial population follows a more logical and structured distribution. This is in comparison to the uniformly distributed coordinates in Figure 15, where all four boreholes are in the lower left region for the first population member as indicated by the number 1s.

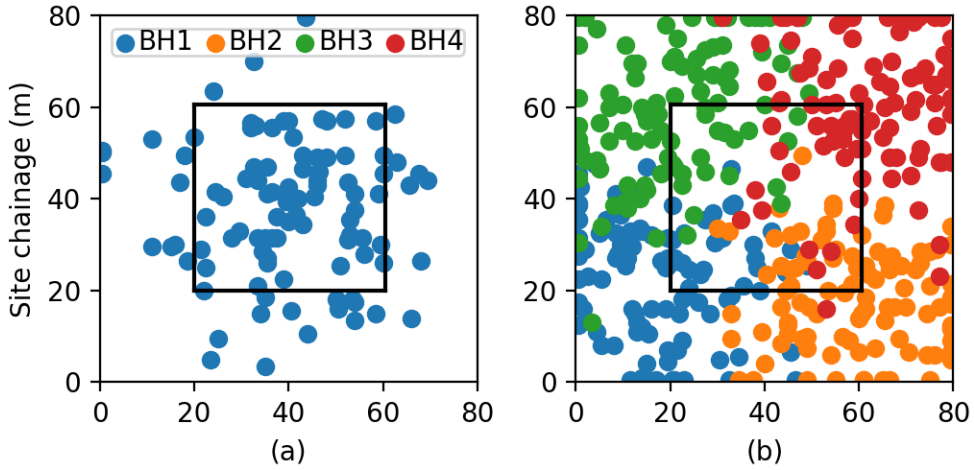


Figure 16: Demonstration of initial borehole coordinates for a set of (a) 1 and (b) 4 boreholes, when normally distributed.

4.2.2.12 Use 2nd Phase of GA on Optimal Solution

The GA's rate of solution improvement tends to diminish over time. Therefore, instead of running it for an exorbitant amount of time for marginal improvement, there is an argument of running it until the improvement rate diminishes, then performing a new stage of the GA with variations of the optimal solution from the first stage. The theory is that GAs are typically good at finding solutions in the general vicinity of the global optimum, but are inefficient at determining the exact optimal solution. Therefore, this process assumes that such a solution in the vicinity of the optimum has been found, and that the 2nd stage will be able to find the true solution nearby.

The initial population of the 2nd phase is generated by duplicating the optimal population of the 1st phase, then creating further duplicates with randomly generated, normally-distributed offsets, as is the case with the normal mode in the previous section.

4.3 si_input

The inputs for this file are described in Table 14. The terms “fwidth” and “fdepth” refer to the width and depth of the soil respectively, in units of soil elements.

Table 14: si_input file line information.

Variable	Data Type	Range	Suggested
Comment – leave blank			
Initial borehole x,y offset from corner of soil in elements	Integer	1-fwidth	
Dimensions of site investigation area in elements	Integer	1-fwidth	
Borehole step size in each dimension for heat map mode	Integer	1-fwidth	
No. test types (leave at 6)	Integer	> 1	6
Use confidence interval to truncate unrealistic sample values	Boolean	T / F	.true.
Confidence interval z-score for the above truncation	Real	> 0	2.576
'add_errors' - add random errors to tested samples	Boolean	T / F	.true.
Comment – leave blank			
Test descriptions*			
Reduction methods - standard deviation below mean, percentile	Real	> 0	1, 0.25
Input custom site investigation	Integer	1,2,3	1
Number of borehole cases to investigate	Integer	> 0	
Number of tests to investigate	Integer	> 0	
Number of reduction methods to investigate	Integer	> 0	
Number of borehole depth cases to analyse	Integer	> 0	
Borehole numbers	Integer	> 0	
Test numbers corresponding to the above tests	Integer	> 1	
Reduction methods, one of: (SA, GA, HA, 1Q, SD)	String	See left.	
Borehole depths (elements)	Integer	1-fdepth	

4.3.1.1 Initial borehole x,y offset from corner of soil in elements

The default manner of specifying borehole locations is by nominating a rectangular area within which the investigation takes place. This input contains both the *x* and *y* coordinates of the top left corner of this area.

4.3.1.2 Dimensions of site investigation area in elements

Two integers describing the length and width of the aforementioned investigation area.

4.3.1.3 Borehole step size in each dimension for heat map mode

The element increment in each dimension when generating a heat map of site investigation quality. For example, a value of “1, 1” produces a continuous heat map over the soil elements, “2, 2” assesses every 2nd element. Depending on the size of the heat map and number of elements, the heat maps can take a moderate amount of time to generate, and could require prohibitive amounts of RAM. In these circumstances, larger step sizes are necessary.

4.3.1.4 No. test types

The total number of tests described for the program, regardless of which tests are to be used in the investigations. Changing test type data is not recommended, and could arguably be hard-coded within the program, however it is exposed here for experts if needed.

4.3.1.5 *Use confidence interval to truncate unrealistic sample values*

When test errors are applied, it is possible that unrealistic sample values are produced. Examples of this include negative, zero, or near-zero soil stiffnesses. Applying a confidence interval, such that all samples outside this interval are removed from consideration, should be applied to mitigate this. The confidence interval is geometric to account for the lognormal distribution. It is recommended that this value is set to true when testing errors are applied, and false when they are not, as the latter will noticeably reduce the program's run-time. See §4.3.1.7 for more details.

4.3.1.6 *Confidence interval z-score for the above truncation*

This z-score sets the degree of confidence for the previously described interval. A value of 2.576 is recommended, which corresponds to 99% confidence.

4.3.1.7 *'add_errors' - add random errors to tested samples*

This Boolean variable determines whether or not random errors are applied to samples. Setting this value to false means that all tests are perfectly accurate, which could result in different tests giving the same results. While setting this value to true provides more realistic results, it also requires a longer computational run-time. Therefore it may be desirable to set this to false when running the genetic algorithm, due to its long run time, and as testing errors are likely to have a negligible impact on optimal borehole location.

4.3.1.8 *Test descriptions**

A set of *n* rows containing information describing soil tests, where *n* is the number of test types. See the section on Soil Test Types and Implementation for further details. These rows form a table that is referenced by position, rather than name.

Each row contains:

Table 15: Line of test description input.

Variable	Transformation error	Bias error	Measurement error	Sampling frequency (elements)	Cost per metre ($)	Test name
Data type	Real	Real	Real	Integer	Real	String
Range	>= 0	>= 0	>= 0	> 0	>= 0	<=4 char

4.3.1.9 *Reduction methods - standard deviation below mean, percentile*

A list of 2 values relating to reduction method implementation.

1. The number of standard deviations below the mean for the SD method.
2. The percentile as a decimal used for the 1st Quartile method. 0.25 corresponds to the first quartile, although other values can be specified including 0.1 for the 10th percentile, or 0.5 for the median.

4.3.1.10 *Input custom site investigation*

If this is set to 1 or 2 as detailed below, then separate input files will be read in order to define site investigations explicitly. Alternatively, site investigations will be defined by inputs in the following subsections. See §4.6.1 for details.

The options are:

1. Generate investigations from the below variables.
2. Read investigation configurations from 'input/si.txt' data.
3. Also read test type and depth on a per-borehole basis. *(New in version 1.1)*

4.3.1.11 Number of borehole cases to investigate

This is the number of different borehole cases analysed, where each case has a specified number of boreholes. This allows the user to investigate the impact of different numbers of boreholes on investigation quality. The cases are given in §4.3.1.15.

4.3.1.12 Number of tests to investigate

The number of test types to analyse. The cases are given in §4.3.1.16.

4.3.1.13 Number of reduction methods to investigate

The number of reduction methods to analyse. The cases are given in §4.3.1.17.

4.3.1.14 Number of borehole depth cases to analyse

The number of different borehole depths to analyse. The cases are given in §4.3.1.18.

4.3.1.15 Borehole numbers

A list of different numbers of boreholes that the user wishes to compare. The total number of borehole cases is specified in §4.3.1.11.

4.3.1.16 Test numbers corresponding to the above tests

A list of test types to analyse. Each number refers to the position in the “test descriptions” table in §4.3.1.8. For example, “1” refers to the test described in the first line. The number of tests is given in §4.3.1.12.

4.3.1.17 Reduction methods

A list of reduction methods to analyse. A method is specified by a two-character string, one of: SA, GA, HA, 1Q, SD. These are described previously in §3.5.2. The number of reduction methods to analyse is given in §4.3.1.13.

4.3.1.18 Borehole depths (elements)

A list of maximum borehole depths to analyse, in terms of elements. The number of depths to analyse is given in §4.3.1.14. The borehole depths are currently programmed to be equal for all boreholes in a given investigation.

4.4 pile_input

Variable	Data Type	Range	Suggested
Comment – leave blank			
Width of the pile in each dimension (elements)	Integer	>= 1	
Differential settlement design tolerance (m/m)	Real	>0	<= 0.0025
Absolute design tolerance if positive (mm)	Real	=/= 0	
Maximum pile length for single-layer soil (m)	Integer	0-fdepth	
FEM convergence tolerance and max no. iterations	Real, integer	>0 >0	<=0.0002 5000
Manner of treating layer boundaries at pile	Integer	>0	
Toggle to specify pile coordinates directly or to generate grid	Boolean	T / F	
Number of piles when giving direct coordinates	Integer	> 0	
List of pile x coordinates (elements)	Integer	1-fwidth	
List of pile y coordinates (elements)	Integer	1-fwidth	
Pile relative load indices	Integer	>= 0	
Number of piles in x, y dimensions of grid	Integer	> 0	
x, y offset of pile grid (elements)	Integer	1-fwidth	
Foundation plan x, y dimensions for grid	Integer	1-fwidth	
Grid of pile relative load indices*	Integer	>= 0	
Proportional tributary area for each pile case	Real	> 0	
Plan view building area	Real	> 0	
Number of floors	Integer	> 0	
Applied loading (MPa)	Real	> 0	8
Differential settlement bounds for failure cost calculation	Real	> 0	0.003, 0.009
Construction cost of building	Real	> 0	
Pile cost per metre	Real	> 0	

4.4.1 General parameters

4.4.1.1 Width of the pile in each dimension (elements)

The width of the pile in the *x*, *y* dimensions in terms of soil elements. See the §3.4.1.5 and §3.4.2.3 for details, for single and multiple layer modes respectively. In the multi-layer case where the pile is treated as circular, the average of the *x* and *y* widths are used in the settlement calculation.

4.4.1.2 Differential settlement design tolerance (m/m)

A differential settlement design tolerance can be specified here. SIOPS will calculate the minimum distance between any two piles, and multiply it by this tolerance value to automatically obtain an absolute settlement design tolerance. For example, if 0.0025 is specified and piles are arranged at 10 m spacings, the design tolerance will be 25 mm.

4.4.1.3 Absolute design tolerance if positive (mm)

If this value is positive, then it will be used as the value for absolute differential settlement design tolerance, regardless of what is set in the previous option. Otherwise, the differential settlement tolerance is used as described above.

4.4.1.4 *Maximum pile length for single-layer soil*

This is the maximum length of the pile in metres in the single layer soil. This must be specified as the required finite element mesh and soil size must increase with pile length. In multiple-layer soils, the maximum length is taken as the bottom of the soil, as there is no computational penalty for doing so.

4.4.1.5 *FEM convergence tolerance and max no. iterations*

Two numbers; the FEM convergence tolerance as a decimal, and the maximum number of iterations as an integer.

The implementation of linear-elastic FEA uses an iterative conjugate gradient solver to avoid constructing the full stiffness matrix, thereby greatly reducing RAM usage. As a result, there is a trade-off between the number of iterations (hence computational time) and accuracy. Note that it is the number of elements in the mesh that is the main factor controlling run time.

The convergence tolerance is calculated as the maximum change in deformations between the current and previous iterations as a proportion of the largest current deformation. A value of 0.0005 should be taken as an absolute minimum, while 0.0001 or less is ideal. By comparison, the maximum number of iterations is largely arbitrary. It is incorporated to stop the program in the unlikely event that an infinite loop occurs. However, hitting this limit will cause the program to exit, so setting an exceedingly large value like 5000 is recommended. See Crisp et al. (2019a) for further information.

4.4.2 Pile parameters

4.4.2.1 *Manner of treating layer boundaries at pile*

The pile must be assessed in a 1D soil, where the effective soil layer depths along the pile are calculated from an inverse-distance-weighted average as discussed in §3.4.3.5. However, it is faster to approximate the layer depths as simply the average layer boundary within the pile radius. The user can specify which method to use for either the soil model (SM), true soil (TS), or both by choice of the following values:

1. Simple average for SM and TS.
2. Simple average for SM, weighted average around pile for TS.
3. Weighted average for SM and TS.

The difference between the modes is usually very small, but can be moderate in highly random layer boundaries. It is recommended that the simple average is used for the soil model when the genetic algorithm is used, due to the long processing time involved. It should arguably also be used for the true soil in this case for the sake of consistency. The difference in speed between the two methods in the site investigation analysis is that the full layer surfaces are interpolated with the inverse distance weighting, while only the needed points are interpolated with the simple average.

4.4.2.2 *Toggle to specify pile coordinates directly or to generate grid*

There are two options of specifying pile locations in SIOPS. If this option is true, then the pile coordinates are given directly (see below). If false, then pile coordinates are derived from a grid (see §4.4.2.7 - §4.4.2.11).

4.4.2.3 Number of piles when giving direct coordinates

This specifies the total number of piles if the above option is set to true.

4.4.2.4 List of pile x coordinates (elements)

A list of *x* coordinates for the piles in terms of elements.

4.4.2.5 List of pile y coordinates (elements)

A list of y coordinates for the piles in terms of elements.

4.4.2.6 Pile relative load indices

A list of integers corresponding to each pile, where each integer is linked to the proportional tributary load. See §4.4.2.10 and §4.4.2.11 for details.

4.4.2.7 Number of piles in x, y dimensions of grid

If piles are specified in the grid manner, then these two values give the number of piles along the *x* and *y* dimensions of the grid.

4.4.2.8 x, y offset of pile grid (elements)

These two numbers specify the pile grid's *x, y* offset from the corner of the site in terms of elements.

4.4.2.9 Foundation plan x, y dimensions for grid

This specifies the length and width of the foundation in terms of elements when the piles are given as a grid (see §4.4.2.2). The piles are spaced equally along each dimension. For example, if there is a set of 3 × 3 piles and the foundation is given as 20 × 30 elements, then the pile spacing will be 10 and 15 elements in the *x, y* dimensions respectively. Note that this is independent of the building area.

*4.4.2.10 Grid of pile relative load indices**

This input consists of a table of integers, where each integer's relative location in the table corresponds to the pile's relative position in the grid. In particular, the integer specifies the pile's relative tributary area, in combination with the next set of inputs given in §4.4.2.11. As such, pile loads are not given directly, rather they are proportional to the building area and related to the number of piles.

For example, if the next set of inputs after this grid were "0.25, 0.5, 1.0", then it would allow for 3 sets of piles, with the 3rd supporting the biggest tributary area, the 2nd supporting half of this area, and the 1st supporting a quarter. The integers given in the table correspond to the order of the above tributary proportions, with 1 = 0.25, 2 = 0.5 and 3 = 1.0.

This is further illustrated by 4 examples given in Table 16 and Figure 17. With "0.25, 0.5, 1.0" specified as proportional tributary areas, the piles corresponding to integers 1, 2 and 3 act as corner, edge and internal piles respectively. Case-by-case descriptions are given below, where all other inputs are identical unless specified otherwise.

Note that any instance of a 0 integer means that the pile carries no loading. This is useful for specifying pile locations that follow a regular grid except where occasional piles are missing. Note that these 0-load piles are still assessed in the pre-processing stage. As such, given a single set of pre-processing, any subset of piles can be subsequently assessed, allowing for the investigation of different building areas, numbers of piles or pile spacings that lie on the original grid.

Table 16: Example of various inputs for pile loading grid.

1	1		1	2	1		0	0	0	1	0	1
1	1		2	3	2		1	1	0	0	0	0
			1	2	1		1	1	0	1	0	1
(a)			(b)				(c)			(d)		

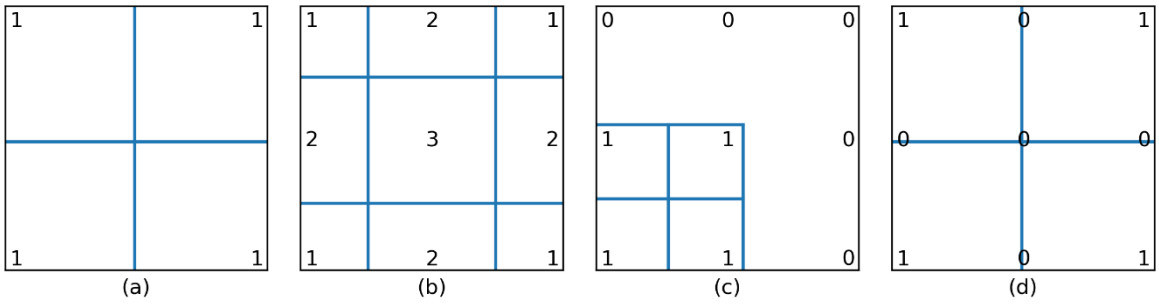


Figure 17: Plan-view demonstration of the building size and tributary loads corresponding to different inputs.

a) 4 piles at the building corners, each carrying a quarter of the building's load.
b) A grid of 3 × 3 piles featuring corner, edge and internal piles, carrying different proportions of the building load.
c) 5 piles have been given a zero load and are ignored. In combination with reducing the building area value, this effectively represents a 4-pile building which is a quarter of the original size.
d) While 9 piles were original specified, only the 4 corner piles have an applied load. Therefore this instance is treated identically to case (a).

4.4.2.11 Proportional tributary area for each pile case

A list describing the set of relative tributary areas as a proportion of the largest tributary area. See the previous input for details.

4.4.3 Building Parameters

4.4.3.1 Plan view building area

The plan view area of the building in terms of square metres. This is used in loading information, so do not consider voids such as internal gardens which do not carry any weight. It is assumed that the building has a similar profile on every floor.

4.4.3.2 Number of floors

Number of floors in the building. Used to calculate building weight.

4.4.3.3 Applied loading (MPa)

The uniformly distributed load supported by the structure. This is applied per square metre, per floor to determine the structure's total weight.

4.4.4 Cost Parameters

4.4.4.1 Differential settlement bounds for failure cost calculation

Values of differential settlement corresponding to zero damage, and extreme damage requiring demolition and rebuilding of the structure. Found to generally be at 0.003 and 0.009 for generic multi-story office buildings. See §3.6 for details on failure cost calculations.

4.4.4.2 Construction cost of building

The cost of constructing the building. This is used in failure cost calculations described in §3.6. If the structural cost is unknown, then it can be approximated by the equation given below, based on Rawlinsons (2016). The equation is only technically valid for the Australian market and the 2016 Australian dollar. However it can be adjusted for inflation, and is likely to be reasonable for comparable countries.

$$C = 1{,}537.5A\, n^{1.2858}$$

Where *C* is the construction cost (A$), *A* is the plan-view area (m), and *n* is the number of floors.

4.4.4.3 Pile cost per metre

The construction cost for piles, per metre.

4.5 soil_input

Variable	Data Type	Range	Suggested
Single layer vs multiple-layer toggle	Boolean	T / F	
Comment – leave blank (single layer soil options)			
X, Y, Z dimensions of original soil (elements)	Integer	> 0	
X, Y, Z dimensions of site (m)	Real	> 0	
Store soils in memory	Boolean	T / F	.true.
Single Layer upscale factor	Integer	> 0	5
Correlation functions for 3D and 2D soils	String	NA	dlavx3 dlavx2
Size of elements (cube length) (m)	Real	> 0	<= 1
Limit for stage 0 matrix size and max no. subdivisions	Integer	> 0	<=8000 <=10
Range of soils to save to disk (lower, upper)	Integer	NA	
Comment – leave blank (single layer soil options)			
Soil distribution	String	n, l, b	l
Mean and Coefficient of variation for soil stiffness	Real	> 0	
Horizontal and vertical scale of fluctuations (m)	Real	> 0	
Comment – leave blank (multiple layer soil options)			
Layer boundary scale of fluctuation (m)	Real	> 0	100
Layer boundary standard deviation (elements)	Integer	>= 0	
Number of layers	Integer	> 1	
List of mean layer boundary depths (elements)	Integer	1-fdepth	
Young's modulus for each layer	Real	> 0	
Standard deviation of Young's modulus for each layer	Real	>= 0	
Toggle to read in custom layer description file	Boolean	T / F	
Toggle to enforce fixed layer depths at boreholes	Boolean	T / F	

4.5.1.1 Single layer vs multiple-layer toggle

Setting this to true runs the program in single layer mode, where a variable random field is generated. Otherwise, the multiple-layer mode is used where each layer consists of a uniform set of stiffness properties while boundaries between layers are variable in depth. See §3.4 for details.

4.5.1.2 X, Y, Z dimensions of original soil (elements)

This set of 3 integers describes the number of elements of the generated virtual soil in the *x, y, z* dimensions. For various reasons, the generated site is at least slightly larger than the final site used, with the latter being a subset of the original. Each value must be representable by the equation $2^a b$, where a and b are integers.

Note that the program will exit if infeasible soil sizes are required. See §3.3.3 and the next option for further details, along with the option in §4.5.1.8.

4.5.1.3 X, Y, Z dimensions of site (m)

The dimensions of the desired site in metres. Note that these dimensions should be smaller than the original generated soil. For example, if the desired soil is 10 × 10 × 10 m and the element size is 0.5

m, then the original soil in the previous option must be at least 20 × 20 × 20 elements. Ideally it should be larger still, to allow the random subset to eliminate subtle bias inherent to the LAS algorithm.

4.5.1.4 Store soils in memory

When this option is set to .true., the virtual soils for all Monte Carlo realisations are generated in advance of the site investigation analysis. This results in a greatly reduced processing time when evolutionary mode is used, as it avoids the recalculation of soil information for every genetic algorithm generation. The only cases where it is desirable to set this to .false. would be if a very large and high-resolution soil were specified, or if a very large number of Monte Carlo realisations (>10,000) were needed, as the RAM requirements would be prohibitive.

For the multi-layer mode, this pre-processes and stores the true layer boundaries for all realisations. For the fixed and heatmap modes, this option has no effect on performance.

For the single-layer mode, it is intended that SIOPS generates a significantly larger random field than the desired site. This allows for individual soils in each Monte Carlo realisation to be obtained by taking a subset of this large field. As the desired site is much smaller than the original, with random offsets for each subset, each one is essentially an independent random field, even if there is significant overlap. Note that setting this option to .true. will result in a significant speedup regardless of mode, as a single large soil can be generated much more quickly than many small soils.

4.5.1.5 Single Layer upscale factor

This is the factor by which the original single layer soil size is upscaled when the 'store soils in memory' option is specified. This factor is applied in each dimension. For example, a value of 2 means the soil is 2 × 2 × 2 = 8 times greater in volume. A value of 5 is sufficient. Alternatively, the user could keep this value at '1' and increase the original soil *x, y, z* dimensions manually.

4.5.1.6 Correlation functions for 3D and 2D soils

Different correlation functions are available, describing the manner of self-similarity in the soil. Almost universally, an exponential Markov correlation function is recommended. Therefore "'dlavx3', 'dlavx2'" should be given on this line.

4.5.1.7 Size of elements (cube length) (m)

The width of the soil elements in metres.

4.5.1.8 Limit for stage 0 matrix size and max no. subdivisions

These two values represent the maximum product of the stage 0 field dimensions, and the maximum number of soil subdivisions as discussed in §3.3.3.For example, if the first value is 512, then the generated 3D soil stage 0 field could have a maximum size of 8 × 8 × 8, 16 × 16 × 2 elements, or some other combination with the product of 512.

2048 and 8 are reasonable choices. As the values increase, so too does the required RAM usage. These inputs do not need to be very large, as the size of the generated soil grows exponentially with each subdivision. Rather, one has to be more mindful that these values are flexible enough to allow the specific dimensions of generated soil requested.

4.5.1.9 Range of soils to save to disk

The user can specify for soils from individual Monte Carlo realisations to be saved as text files. This is done by specifying a lower and upper bound for the range of realisations. For example, "1, 10" will output the first 10 realisations, while "11, 20" will output the second 10.

If the first value is negative, then no soils are saved. To stop the program after soils are exported, make the upper bound negative. For example, "1, -10" will output the first 10 realisations, then exit without performing site investigation analysis.

The soils are saved as a column vector, given in the *X, Y, Z* order where *X* is varying fastest, and *Z* is varying slowest. Furthermore, if SIOPS is run in multi-layer mode, then heatmaps of each layer's lower boundary surface are also saved, except for the bottom layer. The surfaces of depths are defined in terms of elements.

Warning: It is advised that the user be selective about which Monte Carlo realisations they want to export, as saving a large number of soils can require an excessively large amount of storage space.

4.5.2 Single Layer Options

4.5.2.1 Soil distribution

This string specifies the statistical distribution of soil stiffness values within the single layer soil. In theory, the normal, lognormal and beta distributions are implemented with input options "*n*", "*l*" and "*b*" respectively. The lognormal distribution is strongly recommended.

4.5.2.2 Mean and Coefficient of variation for soil stiffness

The mean and coefficient of variation for Young's modulus. See §3.3.2 for details.

4.5.2.3 Horizontal and vertical scale of fluctuations (m)

Two values representing the horizontal and vertical scale of fluctuation for Young's modulus respectively. Note that LAS is used when the soil is isotropic (vertical and horizontal SOFs are within 1 mm of each other), otherwise the piecewise CMD method is used. See §3.3.2 for details.

4.5.3 Multiple Layer Options

4.5.3.1 Layer boundary scale of fluctuation (m)

The scale of fluctuation for the boundary of depths between layers. In practice, it is very difficult to determine what this value should be, owing to the large quantity of layer depth information required for its calculation, and the difficulty in obtaining this information. Limited research has suggested that large values in the order of 100 m should be adopted. The same SOF is used for all layers' boundaries.

4.5.3.2 Layer boundary standard deviation (elements)

The standard deviation of layer boundary depths in elements. For example, if the value is 4 and the element size is 0.5 m, then a standard deviation of 2 m is applied.

4.5.3.3 Number of layers

The number of layers in the soil.

4.5.3.4 List of mean layer boundary depths (elements)

Unless a custom boundary input file is used, each layer is assumed to be horizontal on average, with zero slope. This line of input takes a list of values describing the average depth of each layer in terms of elements. For *n* layers, *n*-1 boundary depths are required, as only boundaries between layers are considered, excluding the upper and lower bound.

4.5.3.5 Young's modulus for each layer

This is a list of mean Young's modulus for each soil layer. Note that the stiffness within each layer is uniform.

4.5.3.6 Standard deviation of Young's modulus for each layer

This is a list of standard deviations of Young's modulus for each layer. As noted above, the stiffness within each layer is uniform. This list of parameters describes variation of E across the Monte Carlo realisations. This option accounts for the inherent uncertainty in knowing the true mean of a layer's stiffness, which could be higher or lower than what is measured.

If a real soil is being modelled and there is a standard deviation of samples obtained, then it is suggested that a standard deviation somewhere between zero and that of the samples is specified. The random stiffness values are lognormally distributed.

4.5.3.7 Toggle to read in custom layer description file

If this is set to true, then the above multiple-layer information is ignored, and input is taken from a separate file with more specific information. This other file allows for boreholes and layer depths at those boreholes to be specified, allowing for real-world sites to be approximated. See §4.6.2 for details.

4.5.3.8 Toggle to enforce fixed layer depths at boreholes

When real-world sites are being modelled as described above, and random noise is being applied to the layer boundaries, then layer depths will randomly deviate from their specified inputs at borehole locations. Arguably, this deviation is not an issue as it helps reflect the uncertainty and subjectivity associated with layer boundaries. However, it is possible to force the layer depths to remain unchanged at borehole locations through an additional processing step by setting this value to .true. This will result in a longer start-up time.

4.6 Optional Input Files

There are additional input files which may be used in order to specify finer control over various parameters. These include files for defining multiple-layer soils and custom site investigations. When specified for use, these files must be in the input folder along with the mandatory input files.

4.6.1 Custom site investigations

While the *si_input.txt* file is suitable for defining boreholes arranged over a regular grid, there may be occasions where alterative patterns or irregular sampling is considered. For this purpose, a set of investigations may be explicitly defined for input through 3 files:

- si.txt – site investigation attributes: number of boreholes, test type, reduction method, borehole depth.
- si_Xcoords.txt – x coordinates of boreholes in metres.
- si_Ycoords.txt – y coordinates of boreholes in metres.

Table 17 describes the overall format of *si.txt*, while Table 18 details the format of individual lines defining the attributes of each investigation in *si.txt*.

Each investigation defined from line 7 onwards in *si.txt* corresponds to borehole coordinates given in *si_Xcoords.txt* and *si_Ycoords.txt* from line 2 onwards. Each column in these latter two files is associated with a borehole. The first row of *si_Xcoords.txt* and *si_Ycoords.txt* should be blank.

Note that these files are formatted such that one can take the SIOPS output from the fixed site investigation analysis mode and use the renamed files as input. Therefore please examine the files generated as referenced in §5.3 if further clarification is needed.

Use of the custom site investigation files is specified through the option in §4.3.1.10.

Table 17: Input description for si.txt.

Variable	Data Type	Range
Total number of investigations	Integer	> 0
Maximum number of boreholes in any investigation	Integer	> 0
-leave blank-		
-leave blank-		
-leave blank-		
-leave blank-		
A set of rows describing each investigation. See Table 18.		

Table 18: Format of each row describing an investigation in si.txt.

Variable	No. boreholes	Test type	Reduction Method	Borehole depth (m)
Data type	Integer	String	String	Real
Range	> 0	CPT,SPT,DMT,TT	SA,GA,HA,1Q,SD	1 - fdepth

New in version 1.1: Additionally, if test type and borehole depth is to be specified on a per-borehole basis, then the files *bh_depths.txt* and *bh_tests.txt* must be included respectively. Their format and use are identical to *si_Xcoords.txt* and *si_Ycoords.txt.* The tests are defined in terms of the test number, and the depths are in terms of soil elements.

4.6.2 Custom multi-layer soils

One can explicitly define multi-layer soils with complex layer geology through specifying the use of *layer_data.txt* in §4.5.3.7. This can be used to approximate real-world soils as uncovered through previous site investigations. This file is required to generate layer boundaries that have an average shape other than a flat horizontal boundary.

The system works by specifying a number of horizontal, plan-view locations where layer depths are known. These could be boreholes for example, and will be referred to as such for the remainder of this section. The depth of each layer is then specified at every borehole.

Variable	Data Type	Range
Number of boreholes	Integer	> 0
Number of layers	Integer	> 0
Young's modulus for each layer (see §4.5.3.5)	Real	> 0
Standard deviation of Young's modulus for each layer (see §4.5.3.6)	Real	>= 0
Borehole x coordinates (list) (in terms of elements)	Integer	1-fsize
Borehole y coordinates (list) (in terms of elements)	Integer	1-fsize
Comment – leave blank		
Layer depth information (elements)* See §4.6.2.1	Integer	1-fdepth

4.6.2.1 Layer depth information

This is a table of information describing the layer depths at known locations. Each row is for a layer, while each column represents a borehole. Note that a valid value must be given for each table entry. If a layer does not exist at a borehole location, then either give the layer a thickness of zero at that location, or potentially put the higher, newer layer depth below the lower one so that it erodes away.

5 Output

This section details the SIOPS output; the various text files produced and how to interpret them. The files can easily be imported into excel, or a scripting environment such as Matlab or Python.

The following files contain a set of information about the overall analysis conditions in the filename. Because these attributes are common, they will be represented by [description], which details the number of piles, building area, and a soil description. For single layers, the soil description is the COV and SOF. For multiple layer soils, it's the number of layers, stiffness ratio of the first two layers (rounded to the nearest integer), and depth of the 2nd layer.

5.1.1.1 Pile Locations - pile_locations-[description].txt

This file describes the pile locations in metres, and is produced regardless of running mode. Note that the soil description is absent as the foundation location is independent of soil conditions. It's a fixed-width file with two columns, where the first row consists of the headings "X" and "Y", and each row after that describes the *X*, *Y* coordinates of a particular pile.

5.2 Test mode

As described in §4.2.1.1, this mode serves as a quick test to determine whether SIOPS will produce reasonable pile lengths for the given input. The results are written to "deterministic_report.txt" saved in the same directory as the executable. There are two sections in this file for the single- and multiple-layer classes respectively. Pile designs which are negative mean that no valid design was found.

While these designs approximate the average pile designs across all Monte Carlo realisations, it should be noted that there may be high variability in such designs across the realisations themselves. If a portion of the spread of designs is not feasible, then they are not considered in the final results. As such, if there are a moderate number of invalid realisations, then the apparent average design could vary from the true one due to being calculated from a truncated distribution. For both soil classes, the maximum pile length is given.

For the single layer class, average pile designs for all reduction methods are given, regardless of which are specified as input. Each row represents settlement associated with a reduction method, while each column represents a pile load case. As the soil model is identical at all locations, all piles that have the same applied load will be given the same design. As such, a design is given for each pile load case, as opposed to each pile. The pile load cases are headed by their relative tributary area.

For the multiple layer class, the designs of all piles are shown, as they could potentially vary if the average layer surfaces are anything other than a flat horizontal boundary. Each column represents a pile, which is headed by the pile load case in terms of relative tributary area, as well as the piles' *x* and *y* coordinates.

5.3 Fixed Mode

5.3.1.1 Investigation Attributes and Statistics - Population-stats_[description].txt

This file describes the majority of site investigation attributes and results.

The first four rows of the file describe the total number of investigations, the maximum number of boreholes, the number of tests, the number of borehole depths and the number of reduction methods analysed. The 5th row has a set of headings describing the following 11 columns:

- Number of boreholes
- Test type
- Reduction method
- Borehole depth (m)
- Failure cost ($)
- Pile construction cost ($)
- Site investigation cost ($)
- Probability of failure (%)
- Average differential settlement (m/m)
- Geometric statistic of differential settlement (m/m)
- Percentage of invalid Monte Carlo realisations (%)

Each row below the headings corresponds to a specific site investigation, the order of which is consistent across all files.

5.3.1.2 Borehole Locations – Population-X/Ycoords_[description].txt

These two files contain the *X* and *Y* coordinates of the boreholes for each investigation in metres. They are fixed width formatted. The first row is a set of headings describing the borehole number. Each row after that corresponds to a particular site investigation, the order of which is consistent across all files. The length of the row corresponds with the number of boreholes associated with that investigation. As such, the rows will likely have a variable length.

5.4 Heatmap Mode

The output of this mode consists of a set of heatmaps titled “Heatmap-[attribute]-[description].txt” where [attribute] describes the contents of the file, which depending on the choice and/or quantity of information requested, can include the:

- Number of invalid Monte Carlo realisations
- Performance metric of choice, with potential additional metrics (one or more of):
    - Total cost
    - Probability of Failure
    - Average differential settlement
    - Failure cost
    - Pile construction cost
    - Geometric statistic

The quantity of additional heatmaps depends on the choice of the "output mode" parameter as described in §4.2.1.1. The performance metrics are described in §3.6, with input information given in §4.2.1.1.

5.4.1.1 File format

The format is consistent across all heatmap files. The first row features headings for the second and third rows. These latter two rows describe the *X* and *Y* coordinates (m) respectively for the top left and bottom right corners of the box containing the heatmap, in case it is a subset of the full soil area. They also give the resolution of the heatmap in the *X* and *Y* dimensions. The resolution is given as "1/element step size", where the step size is discussed in §4.3.1.3.

5.5 Evolutionary Mode

5.5.1 Primary Output

The files described in this section contain essential information, and are always generated when SIOPS is run in evolutionary mode.

5.5.1.1 Investigation Attributes and Statistics - FinalEA-stats_[description].txt

This file describes the majority of site investigation attributes and results.

The first four rows of the file describe the population size, the maximum number of boreholes/chosen performance metric, the number of tests, the number of borehole depths and the number of reduction methods analysed. The 5th row has a set of headings describing the following 12 columns:

- Number of boreholes
- Test type
- Reduction method
- Borehole depth (m)
- Percentage of invalid Monte Carlo realisations
- Number of genetic algorithm generations
- Time taken to convergence (seconds)
- Failure cost ($)
- Pile construction cost ($)
- Probability of failure (%)
- Average differential settlement (m/m)
- Geometric statistic of differential settlement (m/m)

Each row below the headings corresponds to a specific site investigation, the order of which is consistent with the borehole location files. The performance metrics are described in §3.6, with input information given in §4.2.1.1. The site investigation parameters are described in §4.3.

5.5.1.2 Borehole Locations – FinalEA-X/Ycoords_[description].txt

These two files contain the final (optimal) *X* and *Y* coordinates of the boreholes for each investigation in metres. They are fixed width formatted. The first row is a set of headings describing the borehole number. Each row after that corresponds to a particular site investigation, the order of

which is consistent with the stats file. The length of the row corresponds with the number of boreholes associated with that investigation. As such, the rows will likely have a variable length.

5.5.2 Optional output

The following files may be produced depending on the choice of the "output mode" parameter described in §4.2.1.1.

5.5.2.1 Borehole evolution - EvolutionEA-coords_Inv-[investigation No.]_[description].txt

This file describes how the borehole locations evolve with each genetic algorithm generation. The [investigation No.] in the heading refers to the order of the investigation analysed, e.g. 1 for the first, 2 for the second, through to the total number of investigations assessed. As such, each file describes a single investigation.

The first row contains the headings for each column, and each subsequent row corresponds to a generation. The first column gives the performance associated with the generation, and the second provides the percentage of invalid Monte Carlo realisations.

The subsequent columns detail the *X* coordinates for all boreholes, followed by the *Y* coordinates for all boreholes, in metres. The number in the headings represents a particular borehole. For example, the first and second "1" in the first row gives the *X, Y* coordinates of the first borehole, and the latter also marks the start of the *Y* coordinate columns.

5.5.2.2 Final Population - Population-stats_Inv-[investigation No.]_[description].txt

This file describes the final generation's population used in the genetic algorithm. This file is arguably the least important of those described in this section, as the population consists of sub-optimal solutions. It may potentially be useful for showing the sensitivity of investigation performance with different borehole locations, however the final generation is not guaranteed to contain solutions similar to the optimum.

The format of this file is identical to that described in the previous section. However, each row corresponds to a member of the population, rather than the optimal solution at each generation.

5.6 Intermediate Files

The single layer pre-processing stages save several files to the nominated data folder, to be re-used in subsequent runs if needed. While it is not important to understand these files in detail, and since the majority are self-explanatory, the descriptions will be kept brief.

5.6.1 Stage 1 Pre-processing

The Stage 1 pre-processing, which generates the pile settlement curve and soil weights for the PIE method, generates three files.

- settlement_prad-[pile width (elements)]_esize-[element size].txt
- model_bounds_prad-[pile width (elements)]_esize-[element size].txt
- soilweights_ prad-[pile width (elements)]_esize-[element size].txt

5.6.1.1 Settlement

This file simply contains the normalised pile settlement (MPa mm/kN) associated with 1 m increments of pile length; the settlement curve. The first line contains the headers of two columns, with the first being the pile length in metres, and the second being the settlement values.

5.6.1.2 Model Bounds

When generating the settlement curve and soil weights, a large FEA mesh is initially required to minimize the impact of boundary effects. However, as subsequent settlement analysis manipulates the initial FEA results, boundary effects are no-longer a consideration. As such, it is possible to apply the PIE method with a smaller volume of soil compared to the FEA mesh volume, which minimizes computational time in the 2nd stage pre-processing.

Moving away horizontally from the pile's location, and downwards from the pile's base, PIE weights are truncated beyond certain slices such that all weights beyond a slice are less than 0.01% of the field maximum.

This file has three columns, which represent the number of elements from the pile to truncate the soil in the *X*, *Y* and *Z* directions respectively. Each row represents a pile length increment in increasing order. Therefore, as lower rows are associated with longer piles, they should have higher values, corresponding to larger volumes of soil weights to apply.

5.6.1.3 Soil weights

The soil weights are stored as a single binary file, which reduces the amount of hardware space required, and results in fast loading times. The downside is that the file is not human-readable, meaning the PIE values cannot be directly viewed through a text editor for inspection.

The file stores the soil weights associated with all pile lengths.

5.6.2 Stage 2 Pre-processing

Stage 2 generates a single type of file, albeit in multiple instances. These files contain the true pile settlement for all piles and pile lengths for the nominated soil properties.

- ck_pile-[Pile number]_prad-[pile width (elements)]_esize-[element size]._sof-[SOF]_cov-[COV]_anis-1.txt

For *n* piles, SIOPS generates *n* files. Each column in the file represents the settlement associated with a pile length increment, for increasing lengths from left to right. Each row corresponds to a Monte Carlo realisation.

6 Source Code Information

6.1 File description

SIOPS consists of 90 Fortran files, largely following the 95 standard with occasional use of newer features. A set of relevant files are given with a brief description in Table 19. All files not listed are part of the GAF library for generating 2D and 3D random fields.

Note that GFortran is the recommended compiler for generating an executable file from the source code. GFortran is free and open source, allowing for commercial use of SIOPS without charge. Further details can be found here.

Table 19: Description of Fortran source files.

File Name	Description
Main.F90	The main program unit.
Akima.F90	Performs 1D Akima interpolation.
Checkfiles.F90	Checks that necessary files exist for single layer mode, as well as producing example input files.
Despile.F90	Design piles for the single layer mode.
Detcheck.F90	Perform a deterministic analysis for single and multiple layer mode to check that inputs produce valid pile designs.
Edivide.F90	Undertake local averaging to fit soil elements into a finite element mesh with varying element sizes.
ESETT.F90	Various subroutines related to assessing pile settlement for the multiple-layer mode.
Extrafuncs.F90	Library of subroutines to assist in interpolation.
Fem_2d.F90/Fem_3d.F90	Undertake 2D/3D linear elastic finite element analysis.
Fem_*.F90	Remaining "fem_" files contain various subroutines that support the aforementioned finite element analysis.
Ga.F90	Subroutine for performing the Genetic Algorithm
Getdiff.F90	Calculate differential settlement for single layer soil mode
Getcosts.F90	Calculate failure costs from differential settlement.
Getperfmulti.F90	Over-arching file for obtaining information about the ground, such as layer properties and boundary depths
Int2D.F90	Various subroutines for different cases of 2D interpolation via triangulation
Output_results.F90	Save site investigation performance information to disk
PROCESS_CK.F90	Over-arching file for pre-processing soil and pile information prior to site investigation analysis.
PROCESS_SI.F90	Over-arching file for assessing single layer investigation performance.
PROCESS_SI_multi.F90	Over-arching file for assessing multiple layer investigation performance.
pwl_interp_2d_scattered.F90	Various subroutines that support 2D interpolation.
qsort_c_module.F90	Sorting a list in increasing order.
r8lib.F90	Arithmetic utility library, also used to support 2D interpolation.
READINM.F90	Various subroutines for reading input information to control the program's settings.
Reducem.F90	Performing reduction methods for multiple-layer site investigations

SETUP_EA.F90	Over-arching file for preparing the genetic algorithm and processing results
SETUP_SI.F90	Various subroutines for defining different cases of site investigations
si_stats.F90	Defining site investigation performance metrics based on differential settlement
SI.F90	Site investigations and reduction methods for single layer soils.
Sim.F90	Site investigation for multiple-layer soils
SIM2SD_init.F90	Generate correlation matrix for 2D random fields.
SIM2SD.F90	Generate 2D random soils.
sim3de_init.F90	Generate correlation matrix for 3D random fields.
sim3de.F90	Generate 3D random soils with LAS.
Piecewise_CMD.F90	Subroutines to prepare and generate 3D soils with the piecewise CMD method.
Soilgen.F90	Create the multiple layers needed for multiple layer soils.
Variables.F90	Store various global variables, as well as generate the aforementioned correlation matrices.
Weights.F90	Generate the soil weights needed for single layer soil true settlement calculations.
WRITESOILS.F90	Various subroutines for savings soils to disk in different cases, several of which are deprecated.

6.2 Third-Party Code

This section contains references to websites which host code that was incorporated into SIOPS. Note that the code authors did not necessarily develop the theory or algorithms. Rather, they had written an open-source Fortran implementation that was readily available for use.

6.2.1.1 Books with accompanying code:

Fenton and Griffiths (2008) 3D and 2D random fields through local average subdivision.

Smith et al. (2014) Finite element analysis and related subroutines.

Haupt et al. (1998) Simple genetic algorithm.

6.2.1.2 Websites with download links

Mirkov (2017) write up of the aforementioned genetic algorithm code.

Ning (2014) 1D Akima interpolation.

Burkardt (2012) 2D interpolation of scattered data through Delaunay triangulation.

7 Version Tracking

This section details the improvements and new features compared to older versions of the program, as well as plans for new features to be added in the future.

7.1 Changelog

7.1.1 Version 1.1

1. The piecewise CMD method was implemented for generating anisotropic, 3D, variable, single layer soils. See §3.3.4.
2. SIOPS now allows for individual boreholes to have different depths and test types within the same investigation, when using the custom investigation input files. See §4.6.1.

7.2 Planned features

The features discussed in this section are potential candidates for additions to future SIOPS versions. The author does not guarantee that these features will be implemented, however they are detailed here to inform users that they are at least hypothetically feasible to do so.

7.2.1 Hybrid soil model

This would be an enhanced version of the multi-layer mode whereby the properties within each layer will vary in the horizontal direction while being constant in the vertical direction. As such, the soil properties in each layer can be fully defined by 2D random fields. The uniform soil properties required for each pile's true settlement will be determined by an inverse-distance weighted average as is done with the layer boundaries.

The constant properties with depth is analogous to an infinite vertical SOF, which is a worst-case scenario from a pile settlement perspective. However, as a short borehole would provide the same information as a long one, which is unrealistic. To compensate for this, the soil properties within each layer, as encountered by boreholes, would be optionally represented by 1D random fields for which the means are defined by the boreholes' horizontal location. As such, the soil model for the site investigation would be decoupled from that of the true pile settlement. However, the two would be very similar, and in fact identical if the boreholes were sufficiently long in each layer.

This hybrid model would be considered only a rough approximation of the worst case soil conditions, due to both the unrealistic infinite vertical SOF and the decoupling of the aforementioned soil models. It is intended to be used as an upper bound of soil variability, whereas the current multi-layer method with its 0% within-layer variability, would be a lower bound.

7.2.2 Bridge Mode

While SIOPS is currently implemented to simulate multi-story buildings supported by piles, it could theoretically be used to model any structure supported by piles, including bridges. However, there are a number of limitations that would be associated with such an implementation:

1. Failure cost could not be used as a metric, as the cost relationship has been derived specifically for generic multi-story buildings.
2. Bridges tend to be supported by pile groups which have an associated cap that causes each pile to settle at the same rate. The pile settlement models implemented in SIOPS do not consider the proximity of other piles in any way, shape or form. As such, the pile group

would have to be implemented as a group of individual piles, potentially with their settlement being averaged or approximated in some other manner. However, since the method of foundation modelling is consistent in both the soil model and true soil, any errors from approximation would be minimised, as the discrepancies are self-cancelling.

8 Licence

SIOPS is released under the MIT licence, which is given below:

MIT License

Copyright (c) 2020 Michael Crisp

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

9 Acknowledgements

I would like to thank my PhD supervisors Mark Jaksa and Yien Kuo for their years of generous guidance, both technical and otherwise. This program would not have been possible without their support.

I also wish to thank Gordon Fenton and Vaughan Griffiths for their roles in developing the virtual soil generation and finite element analysis subroutines, which are incorporated in the program. These components together form the Random Finite Element Method, a powerful statistical technique which this program is based on. Mark Jaksa also conceived the initial concept of site investigation optimisation, using this technique, that ultimately led to this program.

Finally, I thank Jason Goldworthy and Ardy Arsyad for their early work in this field, as well as Jianye Ching for providing early access to the Pseudo-Incremental Energy method equations that form one of the significant speed optimisations.

10 References

Akima, H. 1970. A new method of interpolation and smooth curve fitting based on local procedures. Journal of the ACM (JACM), **17**(4), 589-602.

Burkardt, J. 2012. Piecewise Linear Interpolant to 2D Scattered Data. Retrieved 2019, from https://people.sc.fsu.edu/~jburkardt/f_src/pwl_interp_2d_scattered/pwl_interp_2d_scattered.html

Ching, J., Hu, Y. G., and Phoon, K. K. 2018. Effective Young's modulus of a spatially variable soil mass under a footing. Structural Safety, **73**, 99-113.

Crisp, M. P., Jaksa, M. B., and Kuo, Y. L. 2019a. Framework for the Optimisation of Site Investigations for Pile Designs in Complex Multi-Layered Soil, Research Report, School of Civil, Environmental and Mining Engineering, University of Adelaide, Adelaide, S.A.

Crisp, M. P., Jaksa, M. B., and Kuo, Y. L. 2019b. Influence of distance-weighted averaging of site investigation samples on foundation performance. *In* Proceedings of the 13th Australia New Zealand Conference on Geomechanics, Perth.

Crisp, M. P., Jaksa, M. B., and Kuo, Y. L. 2019c. Toward a generalized guideline to inform optimal site investigations for pile design. Canadian Geotechnical Journal.

Crisp, M. P., Jaksa, M. B., and Kuo, Y. L. 2019d. Towards Optimal Site Investigations for Generalized Structural Configurations. *In* Proceedings of the 7th International Symposium on Geotechnical Safety and Risk, Taipei.

Crisp, M. P., Jaksa, M. B., and Kuo, Y. L. 2020. Characterising Site Investigation Performance in a Two Layer Soil Profile. Canadian Journal of Civil Engineering.

Day, R. W. 1999. Forensic geotechnical and foundation engineering: McGraw-Hill New York.

Delaunay, B. 1934. Sur la sphere vide. Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk, **7**(793-800), 1-2.

Fenton, G. A., and Griffiths, D. V. 2008. Risk assessment in geotechnical engineering. Hoboken: Wiley.

Fenton, G. A., and Vanmarcke, E. H. 1990. Simulation of random fields via local average subdivision. Journal of Engineering Mechanics, **116**(8), 1733-1749.

Goldsworthy, J. S. 2006. Quantifying the risk of geotechnical site investigations. Ph.D. thesis, School of Civil, Environmental and Mining Engineering, The University of Adelaide, Adelaide, S.A.

Griffiths, D. V., and Fenton, G. A. 2009. Probabilistic settlement analysis by stochastic and random finite-element methods. Journal of Geotechnical and Geoenvironmental Engineering, **135**(11), 1629-1637.

Haupt, R. L., Haupt, S. E., and Haupt, S. E. 1998. Practical genetic algorithms (Vol. 2): Wiley New York.

Li, D.-Q., Xiao, T., Zhang, L.-M., and Cao, Z.-J. 2019. Stepwise covariance matrix decomposition for efficient simulation of multivariate large-scale three-dimensional random fields. Applied Mathematical Modelling, **68**, 169-181.

Mirkov, N. 2017. Practical Genetic Algorithms. Retrieved 2019, from https://github.com/nikola-m/practical-genetic-algorithm

Mylonakis, G., and Gazetas, G. 1998. Settlement and additional internal forces of grouped piles in layered soil. Geotechnique, **48**(1), 55-72.

Ning, A. 2014. Fortran 1D Akima Implementation. Retrieved 2019, from https://github.com/andrewning/akima

Rawlinsons, A. (2016). Australian Construction Handbook (34 ed., pp. 1005). Perth, Australia: Rawlhouse Publishing Pty. Ltd.

Smith, I. M., Griffiths, D. V., and Margetts, L. 2014. Programming the finite element method (5th ed.): John Wiley & Sons.

Xiao, T., Zhang, L.-M., Li, D.-Q., and Cao, Z.-J. 2019. Efficient 3-D Random Field Simulation with Separable Correlation Functions.