FMGM 2015 – PM Dight (ed.)
 ISBN 978-0-9924810-2-5

Geotechnical data handling from A to Z

A Thorarinsson *Vista Data Vision, Iceland*

Abstract

While geotechnical sensors of all kinds have greatest attention during design, construction and implementation of geotechnical monitoring projects, it is actually the aftermath that in the long run become most important; how to handle all that data in a sensible way to get maximum information with minimum work.

The data does not have an independent life but will influence how dataloggers are programmed, their period set and where sensor scaling is implemented. The data handling structure implemented is the same whether datalogging projects are small or large.

A data handling project may be segmented into sensor data maintenance, logger maintenance, communications, alarm handling, reports, identify data for system maintenance and for performance, and to address data with slow or very fast sampling rate. In the end, the data should be presented in a sensible manner such that it gives the user a clear view of the situation and help them to make their conclusions.

1 Introduction

This is the third episode of the journey that started at FMGM 2007 and continued at FMGM 2011, e.g. collection of subjects that matters all those who work with instrumentation and datalogging. The FMGM 2007 paper had the title 'Methods for automatic storage, visualisation and reporting in datalogging applications' (Thorarinsson 2007), and the FMGM 2011 paper had the title 'Real time, on-line monitoring for every project – FMGM 2011' (Thorarinsson & Simmonds 2011), co-author is Tony Simmonds of Geokon, USA. This time the subject is about the datalogger and its sample rate and period, the usefulness of having access to temperature data as well as geotechnical data and finally various methods of displaying geotechnical data in order to bring forward the secrets buried deep under. The title of the paper this time, 'Geotechnical data handling from A to Z', is not entirely correct, as no paper will ever cover every angle of this subject. However, it is hoped that a keen reader reading this paper and the previous two papers will get new ideas about data handling and may master the art of geotechnical data handling as well as the art of operating successfully a datalogging system with minimum work.

2 Dataloggers sample rate and period

The definition of sampling rate and a datalogger period is as follows: The sampling rate is how often a datalogger reads its sensor inputs. Depending on the application, a practical sampling rate in geotechnical application might be from once per second up to once per hour. The datalogger period is how often the datalogger store data in its data file. Clearly, there is a minimum of one sample for each period. As an example, a datalogger may read its sensors every one minute and store the average reading every 10 minutes in the data file, or simply read its sensors every 10 minutes and store that single reading in the data file. It is up to the designer to set the datalogger sampling rate and period.

2.1 Dataloggers — very short period versus longer period

This example is about to use a datalogger to capturing readings from a crackmeter. As there are blastings in the area, the datalogger was programmed to store crackmeter readings every 100 ms, therefore hoping to catch crack movements, sound an alarm if there is a movement and for the operators to be able to draw conclusions from the crackmeter data. A typical one second of 100 ms data storage is shown in Table 1.

Table 1 Crackmeter data, stored every 0.1 second

Time	Crackmeter sensor reading
15:06:07.6	30.9
15:06:07.7	30.9
15:06:07.8	30.9
15:06:07.9	30.9
15:06:08.0	30.9
15:06:08.1	30.9
15:06:08.2	30.9
15:06:08.3	30.9
15:06:08.4	30.9
15:06:08.5	30.9

Table 1 indicates that the method of storing data in a datalogger ten times every second is not an effective one as there is bound to be endless replication of data. Perhaps there is a method that will give same result but with less data.

The amount of data involved was examined. As data is stored every 0.1 second there are 10 records every second, 600 records every minute and 36,000 records every hour, and this is for each sensor being monitored. This is enormous amount of data and gives all kind of data management problems. As an example, Microsoft Excel is unable to plot more than 32,000 points. Common size of computer displays is only able to display 1,000 to 2,000 points which equals only three minutes of data at this sampling rate.

Figure 1 shows the actual data as read from the sensor. One might argue that higher resolution of the sensor data would show smoother trend line which of course is true; however, that would not solve the real problem associated. As an alternative one might consider to keep the same sample rate (100 ms) but store the average of the sensor data every five minutes instead of every 100 ms. In addition to storing the average reading the maximum and minimum reading for each five minutes may be stored as well. Therefore, three data records will replace 3,000 records, a 1,000 fold reduction of data. The important question is: Will this new arrangement show crack movement as well as before?

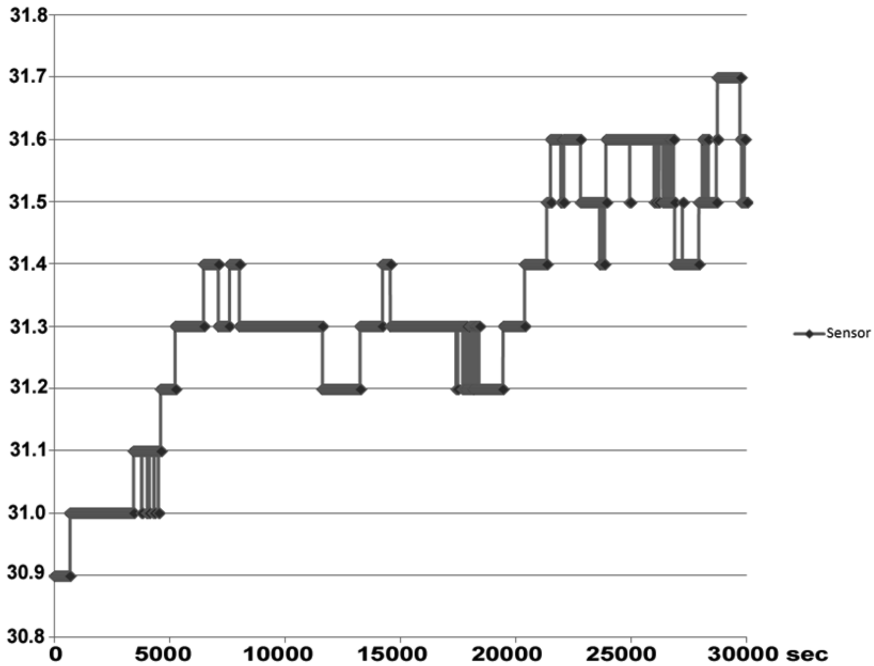

Figure 1 Trend line showing 50 minutes of 0.1 second data, altogether 30,000 records

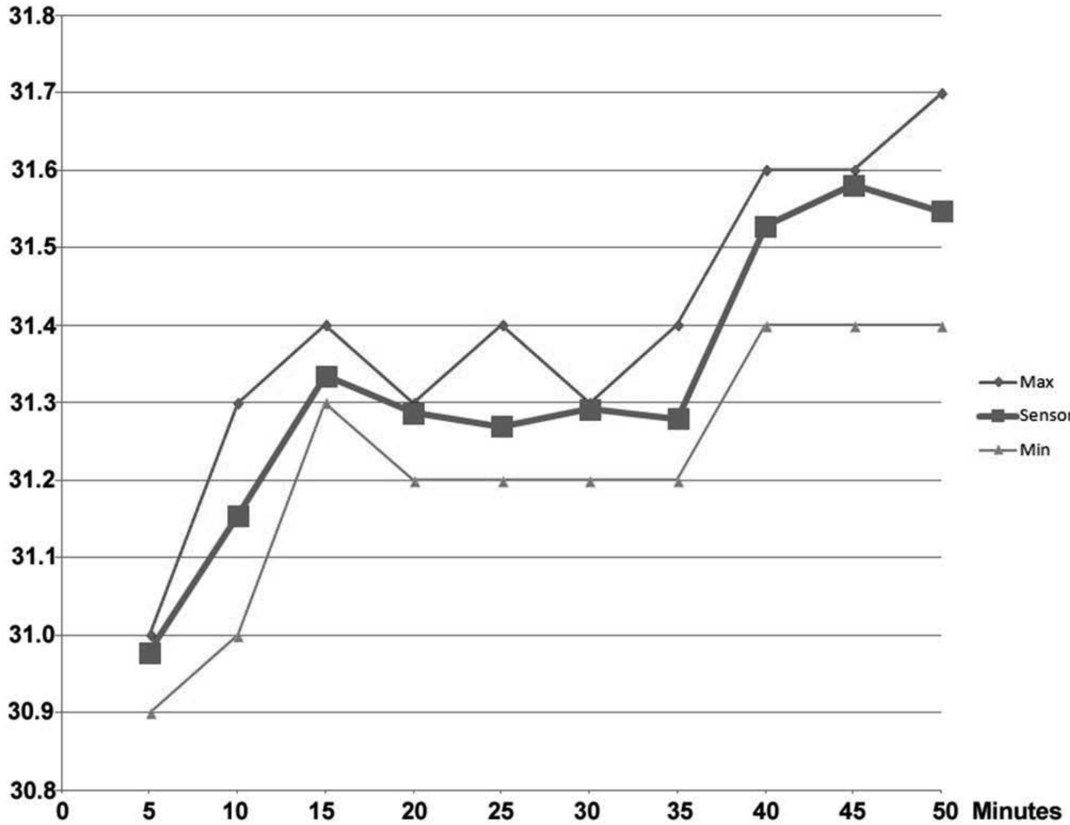

Figure 2 50 minutes of five-minute data, with the associated maximum and minimum readings for each five-minute period. The sample rate is 0.1 second as before

When comparing the two figures, it seems like the readability of the latter one is no worse than the first one and perhaps a lot better. This approach would solve one important problem: there are practical issues involved like the problem of browsing through days of data if each hour of data is as much as 36,000 records; when looking at the trends for one month, some 25 million records must be read for each sensor. That will simply not work.

This example is rather extreme but it serves the purpose of pointing out the importance of choosing datalogger sample rate and period wisely.

2.2 Long period versus shorter period

Section 2.1 pointed out that a very short datalogger period may give problems because of the amount of data involved. What if the datalogger period is longer than those minutes mentioned in Section 2.1, perhaps by several hours? Will that work well?

In this example, we look at data from a datalogger which is programmed to store data every six hours. Figure 3 shows seven days of these measurements. In order to understand the trend lines better, boxes mark the actual time of each measurement.

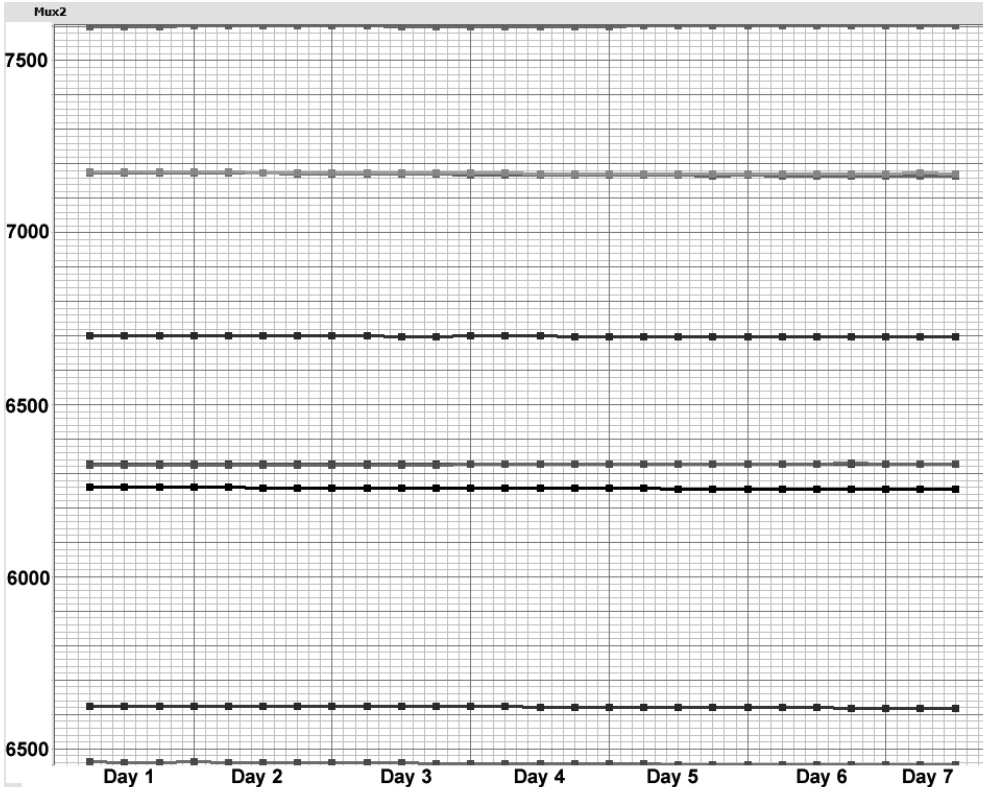

Figure 3 **Seven days of digits readings from seven vibrating wire sensors. There are readings every six hours, starting at midnight. Y-axis has units in digits**

The graphs show stable readings. One might ask if some information is hidden in the data. By removing all trend lines but the blue one (the third from top) and display the sensor data on a graph with auto scale then the data give new information.

Figure 4 **The trend line shows interesting upward trend with daily fluctuations. These seven days show an increase of the sensor digit readings by five digits**

When plotted alone the trend line of Figure 4 does not reflect the steady state situation indicated in Figure 3 but has an upward trend. Next question address the fluctuations where every fourth reading (at noon) is highest and the next reading thereafter (at 0600 h) is lowest. Is it possible to find the solution to that mystery? Perhaps the solution is found in the datalogger's panel temperature reading.

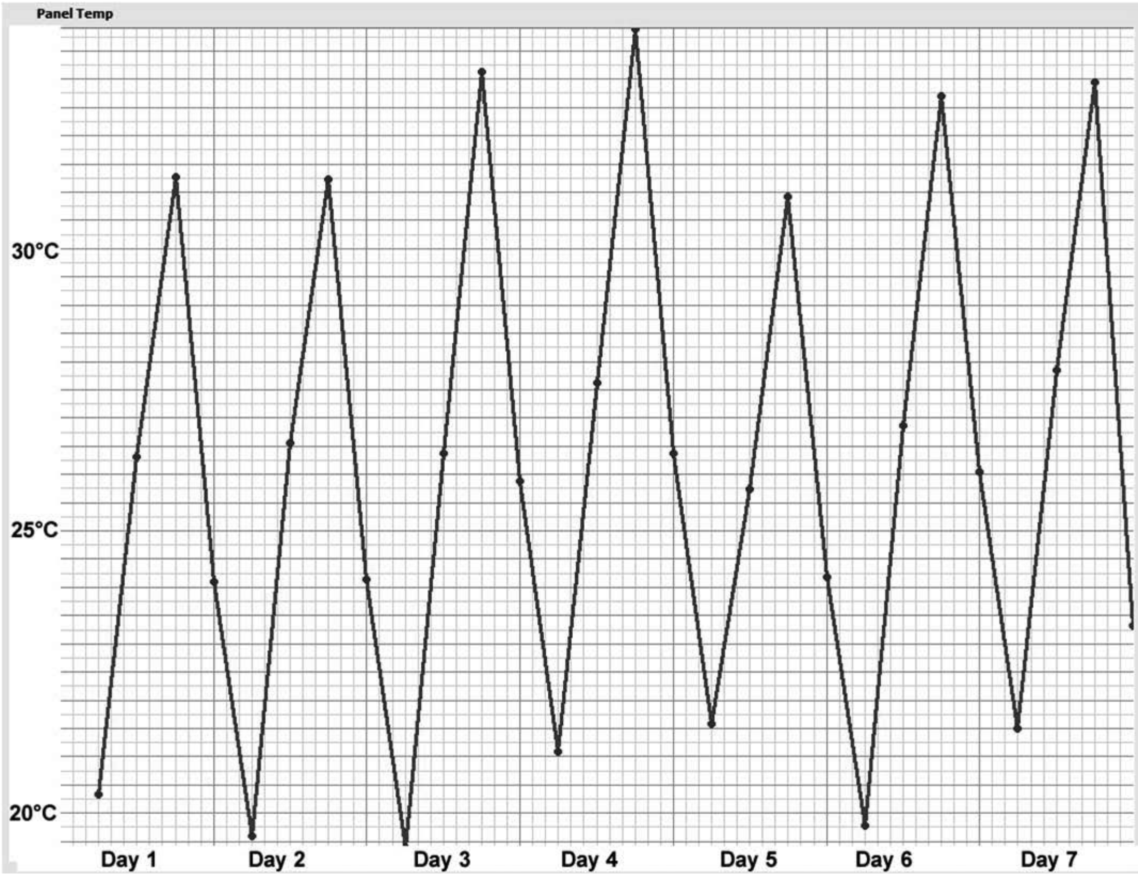

Figure 5 Panel temperature readings of the datalogger. Readings are every six hours starting at midnight. Maximum is at 0600 h every day

What may be learned from Figure 5 is that the datalogger is located such that its internal temperature swings every day, being highest in the afternoon and lowest in the early morning. The temperature is highest in the afternoon when the sun is shining. There is a correlation between the vibrating wire sensor and the temperature such that when the temperature is highest there is a drop in the vibrating wire sensor reading. It would be tempting to plot on an XY graph the sensor readings as a function of the temperature but with only four readings every day the data sample is rather low for extracting useful information. See Section 2.4 about temperature correlation of sensor data and 2.3 about adjusting the period time to response time.

2.3 Periods and alarms: how datalogger period time affects alarm response time

Geotechnical sensors are installed in order to discover movements and to send alarms to operators if movements cross the thresholds set. Every now and then, the datalogger may not record the actual sensor value but rather a spike signal which may initiate a false alarm.

As the datalogger period is one hour then the maximum delay of issuing an alarm is also one hour; this is because the alarm situation may rise the moment after the current period starts. Even if one hour delay may be long, it is quite possible that it will cause no problem in the actual project. Then there is the question of spike signals which may occur for various reasons (Figure 6).

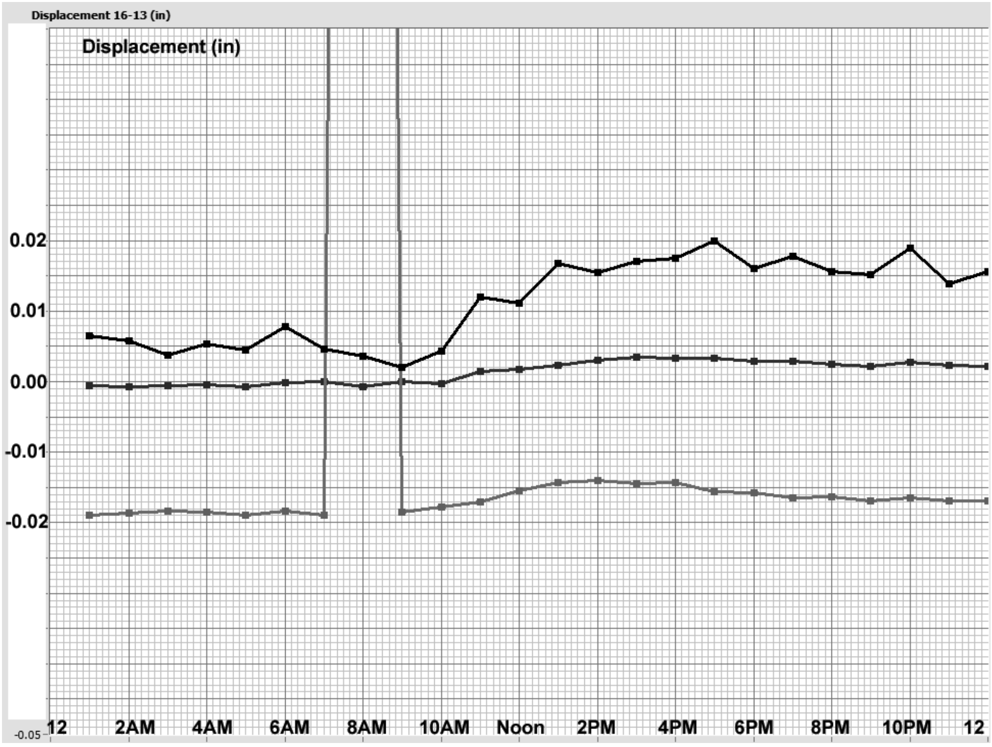

Figure 6 24 hours of displacement readings. There are readings every one hour. At 0800 h the bottom trend line indicates a spike signal that is way out of range

False alarms are never good and repeated false alarms will simply slow down the operator's response time. When adjusting alarm settings the designer has two options: accept that a single spike signal will initiate false alarm, or configure a filter function that will not initiate an alarm unless the alarm situation is recorded at least twice (Figure 7).

Deadband (%)	Delay (minutes)	Repeat (minutes)	Skip Off Alarms
2	61	0	☐

Figure 7 The delay filter has been set to 61 minutes

The datalogger's period is 60 minutes. As may been seen in Figure 7 the alarm delay filter has been set to 61 minutes; however, any setting up to 119 minutes (two times the one hour period) will work. This means that the alarm will not be initiated until the second high sensor reading is read. The consequence is that the maximum delay of issuing alarm will double from one to two hours. If this longer alarm response time is not acceptable, then solution is to shorten the datalogger's period from one hour to 30 minutes, adjust the delay to 31 minutes and therefore keep the one hour alarm response time with the alarm spike filter. Of course, if the project is sensitive then the datalogger period could be still shorter. If there are more than a single datalogger in the project the designer need to think about synchronised time period, see paragraph 5 in the paper 'Real time online monitoring for every project' (Thorarinsson & Simmonds 2011).

2.4 The effect of temperature correlation

It is not uncommon to see trend lines that seem to be affected by day and night temperatures. The cause may be that the sun is shining on the sensors or on the structure or on the datalogger. Therefore, there may be actual movement caused by temperature expansions or the sensors or the datalogger may be sensitive to temperature. Perhaps the sensor readings are affected by all three. An example of such data is seen in Figure 8 showing trend lines of real world tilt sensor data.

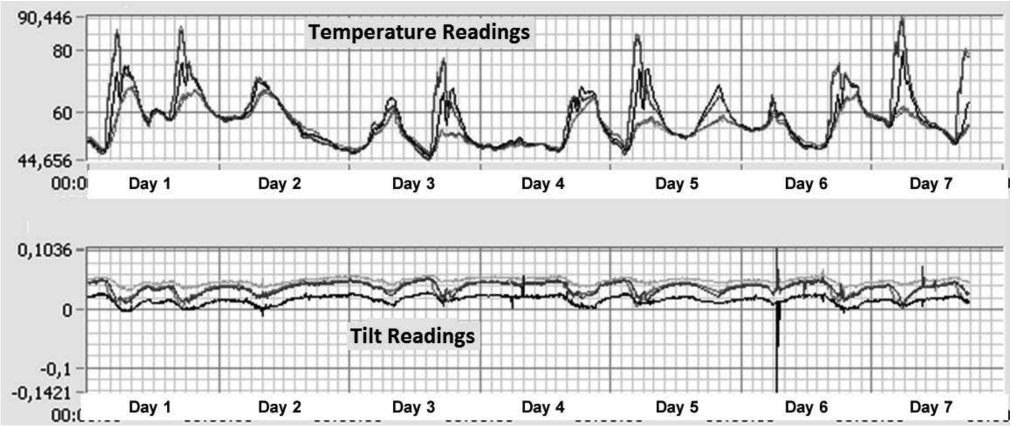

Figure 8 Seven days of tilt data and the corresponding temperature readings of each tilt sensor

The tilt sensor which is represented by the lowest tilt readings trend line and its corresponding temperature sensor from the top graph was analysed. To understand the correlation between the tilt reading and the tilt sensor's temperature an XY graph is the perfect tool (Figure 9).

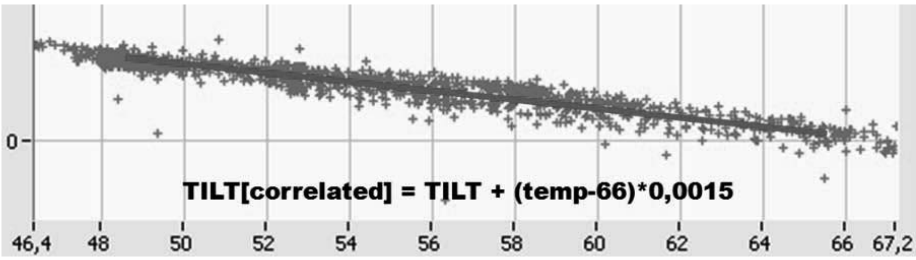

Figure 9 XY graph showing tilt on Y-axis and temperature on X-axis

As Figure 9 shows, there is in this case a clear correlation between the tilt sensor's temperature and the tilt reading. By implementing a simple linear equation and then using a virtual variable feature to recalculate the tilt data into temperature correlated tilt data a new trend line is created which may be plotted and smoothed (Figure 10).

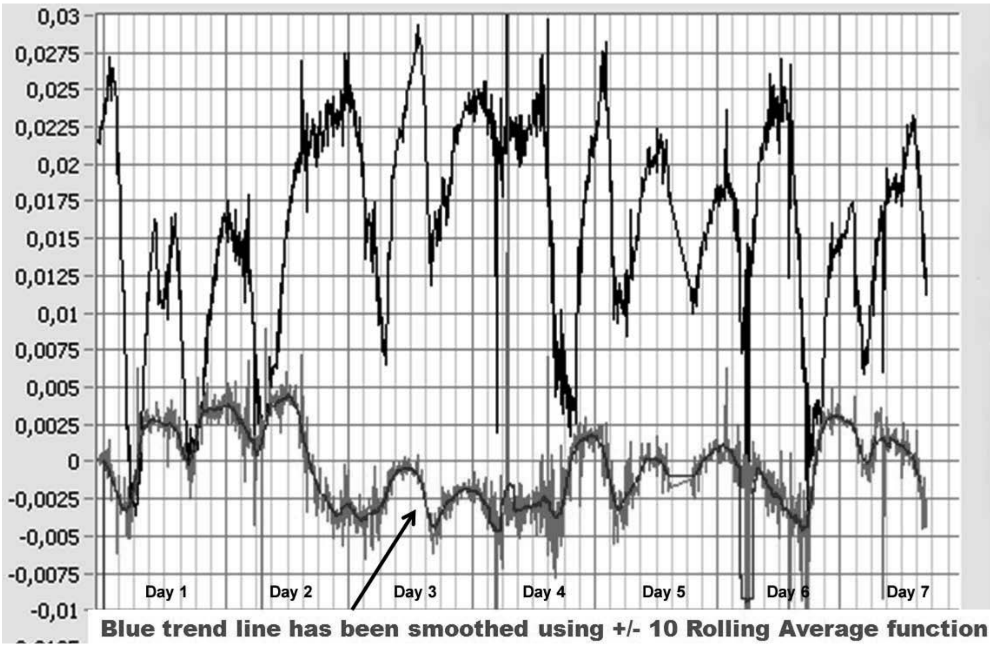

Figure 10 Original tilt data (top), the temperature correlated tilt data (lower) and the smoothed version (thin line)

This exercise reveals that the correlated tilt data swings around zero while the un-correlated data has an offset; the cause for this offset may be seen in Figure 9, where temperature lower than 67°F will add offset to the sensor reading. The two trend lines only touch when temperature is 67°F. The data itself does not indicate if it is the datalogger or the sensor that is sensitive to temperature changes or if it is the structure that expands with temperature. The designer should check the sensors and dataloggers in a workshop to understand its sensitivity to temperature changes.

3 Various methods of displaying geotechnical data

Most commonly, trend lines and project overviews are used to present geotechnical data. There are other visualisation tools that may be useful to help the operator to understand the situation and to report on. Three of these visualisation tools are rate-of-change, intensity plot and overlay graph.

3.1 The rate of change (RoC)

The RoC trend lines display the changes rather than the actual sensor reading. RoC may be configured as the difference between a sensor reading and the reading before, or the reading a few hours before, or the reading 24 hours ago. RoC handling solve some of the alarm problems of drifting data as it frees the operator of constantly readjust the alarm trigger levels to follow the drift; in such cases, the RoC is fine tool to plot drifting data as sudden changes become clearly visible. RoC is also great for plotting sensors of widely different scale when the interest is in sudden changes in sensor readings. A number of examples are shown in Figures 11, 12 and 13.

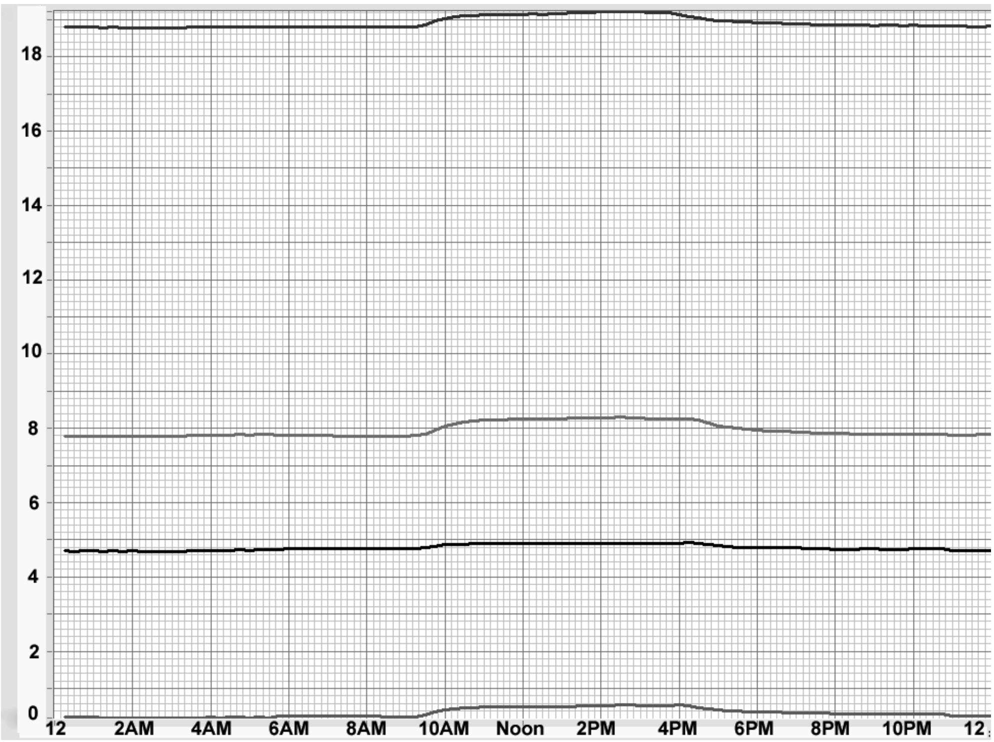

Figure 11 Trend lines showing one day of data from four extensometers. The period is 15 minutes

The trend lines of Figure 11 indicate a stable situation but there are secrets underneath that may be made more visible using a different plotting method. The same sensors are plotted again, but this time using a RoC plot.

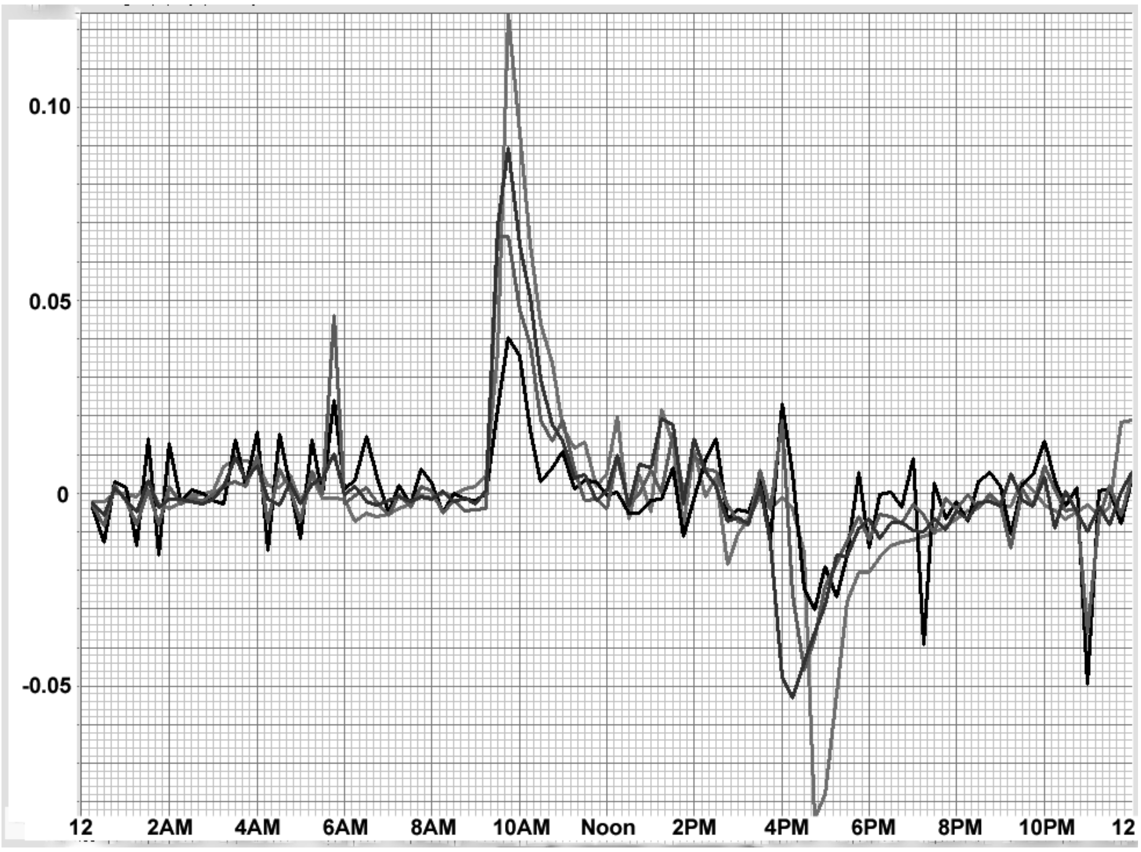

Figure 12 RoC plot for the same trend lines as in Figure 11. The RoC function is the difference between a reading and the reading before

Figure 12 is interesting for various reasons. First is the fact that all four trend lines which have difference scale are now on the same scale and may easily be compared. Then the morning and afternoon fluctuation become clearly visible where readings in general jump up at 1000 h and fall down at 1600 h. Finally, specific with this sample of sensor data, there are small synchronous movements in the early morning (from midnight to 0700 h) on three of the four sensors; movements that are so small that they are hardly visible but may give important indication of the system behaviour.

RoC may be configured in various ways depending on what the designer is seeking. One configuration is such that daily temperature fluctuations disappear and it therefore becomes easier for the operator to understand if all is stable or if there is an ongoing trend.

Figure 13 shows that daily temperature fluctuations will disappear if 24-hour RoC is applied. This is great for all studies of long term stability. Alarm functionality may improve, as only actual movement will show.

Figure 13 Three RoC trend lines created from a single set of extensometer data with 15-minute period. The trend line with highest peak is the difference between a reading and the reading before; the trend line with the second highest peak is the difference between a reading and the reading one hour ago and the straight trend line is the difference between a reading and the reading 24 hours ago

3.2 The intensity plot

The purpose of the Intensity Plot is to display long periods of sensor data such that daily trends become clearly visible. Long periods are not only days or weeks but rather as long as a year. As the discovery of daily trends requires datalogger periods that are short comparing to a day it is practical to choose datalogger period shorter than one hour and as short as 10 minutes. Therefore, there may be as many as 30,000 to 60,000 data points on display in a single intensity plot of one year of data.

In the example shown in Figure 14, there is one year of extensometer data with readings every 15 minutes, altogether 35,040 readings. As may be seen on the trend line the sensor has been re-adjusted early summer and again late summer which is indicated by a jump in the sensor readings. There is one week of missing data in August. In general, this extensometer shows slow ongoing movement as indicated by the declining readings.

The trend line of Figure 14 mainly shows the slow going drift of the sensor data as well as the re-adjustment of the sensor. Is there any hidden information in this data? Let's plot this one year of extensometer data on an intensity plot (Figure 15).

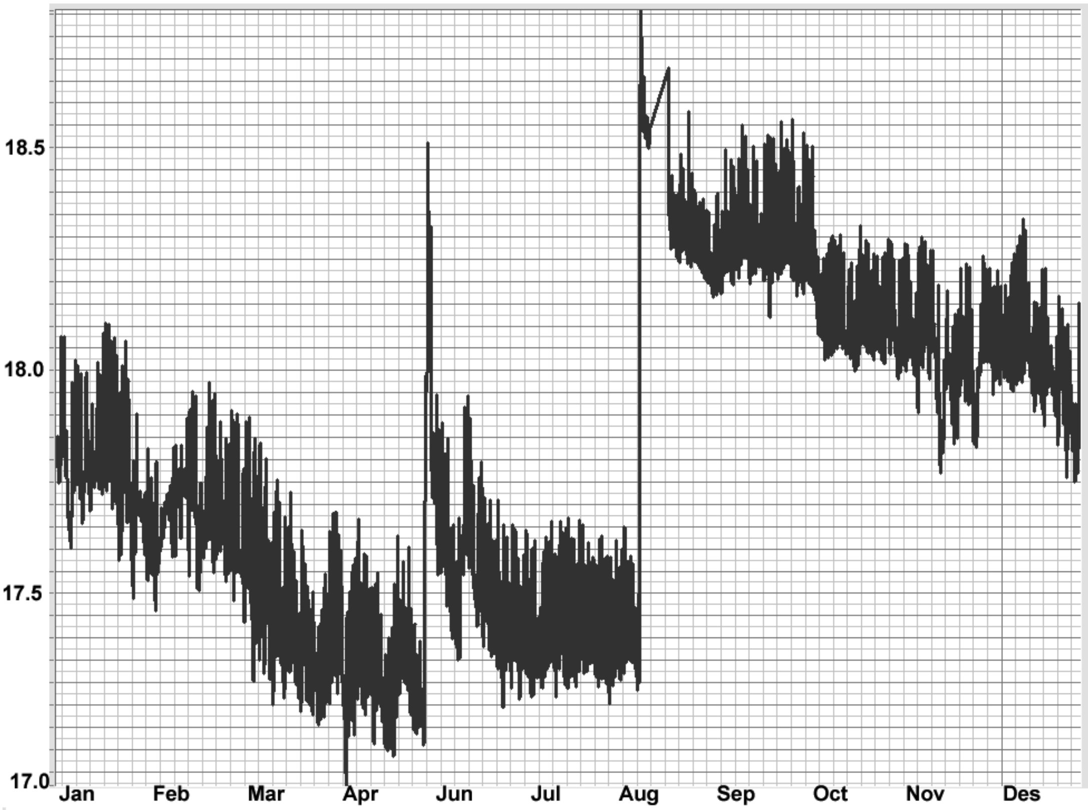

Figure 14 Trend line showing one year of extensometer data with readings every 15 minutes

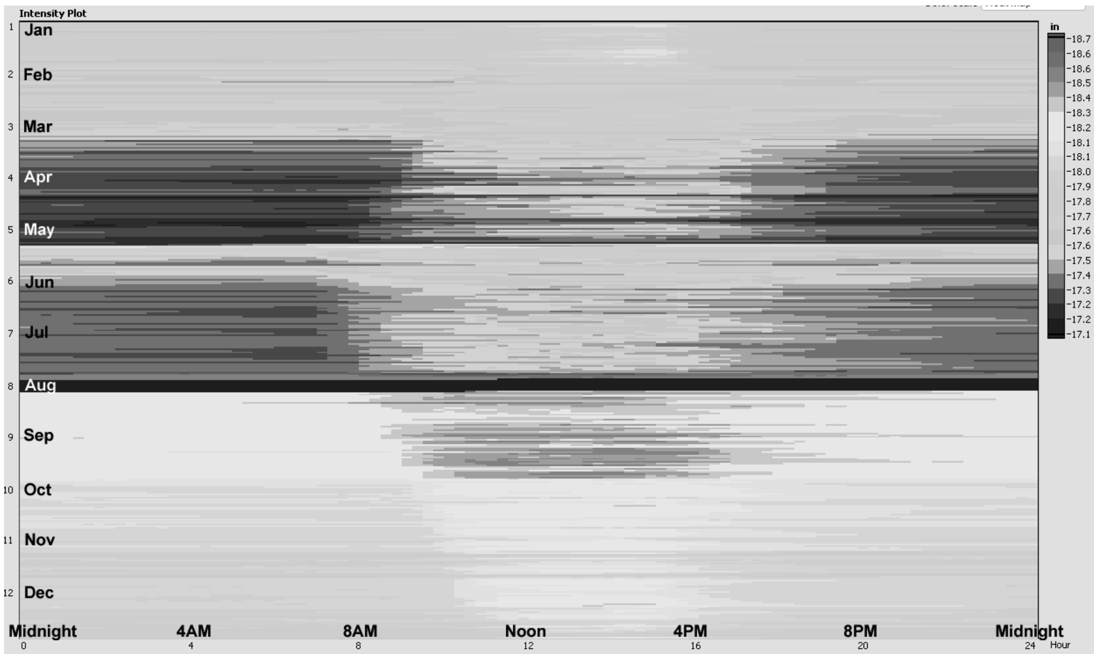

Figure 15 Intensity Plot showing the same data as Figure 14, e.g. one year of extensometer data with readings every 15 minutes. The Y-axis shows the days and months and the X-axis show the hours within each day

The Intensity Plot in Figure 15 shows all the 15-minute data for one year on a single display. In this dataset the typical solar footprint is recognised, being cantered on, clearly showing how sunrise and sunset affects sensor readings. The missing data in August is clearly visible as the horizontal dark line. The time of the readjustment of the extensometer sensor in May and August is clearly visible. Intensity Plot is an excellent tool for investigating if daily trends are hidden in large datasets.

3.3 The overlay graph

The purpose of the overlay graph is to plot days of data on top of each other such that any daily trends of the data become easily visible. The operator has several options regarding the display.

Daily trends are those changes in sensor readings which repeat themselves at the same time day after day. Quite often, these trends are caused by the heat of the sun or by some action related to the work flow. Overlay graphs and their cousin the Intensity Plot are perfect tools to investigate daily trends. In this example, for the sake of simplicity, we will use the same extensometer data, as shown in Figure 14.

Figure 16 shows that the general trend is that readings jump up in the morning and drop in the afternoon. Some days, the jump is greater; other days, it is less. However, during these 28 days there is not a single day without this trend.

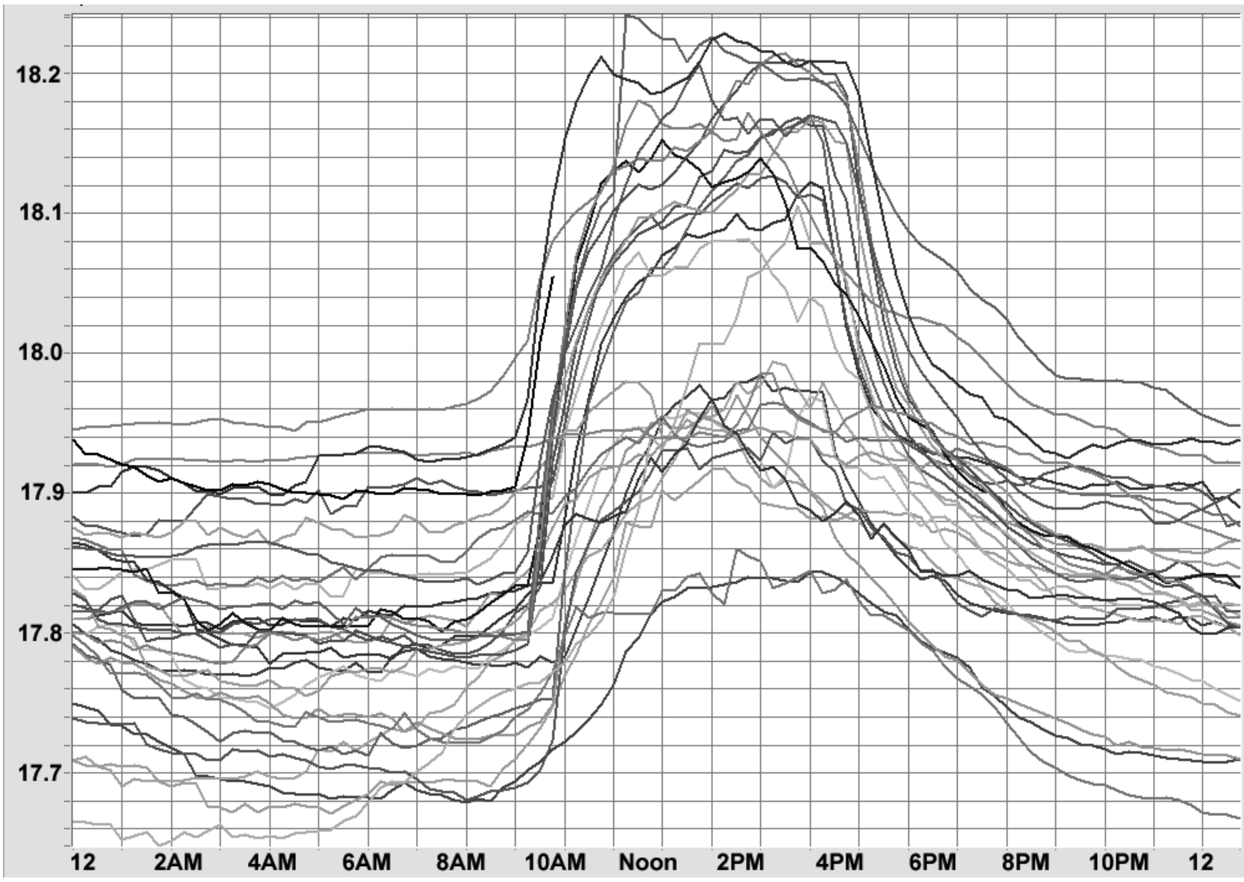

Figure 16 Overlay graph, showing 28 days of extensometer data on top of each other

Figure 17 shows a total of 28 weeks which equals 196 days' worth of data. There is no doubt that this morning/evening trend has been ongoing for a long time.

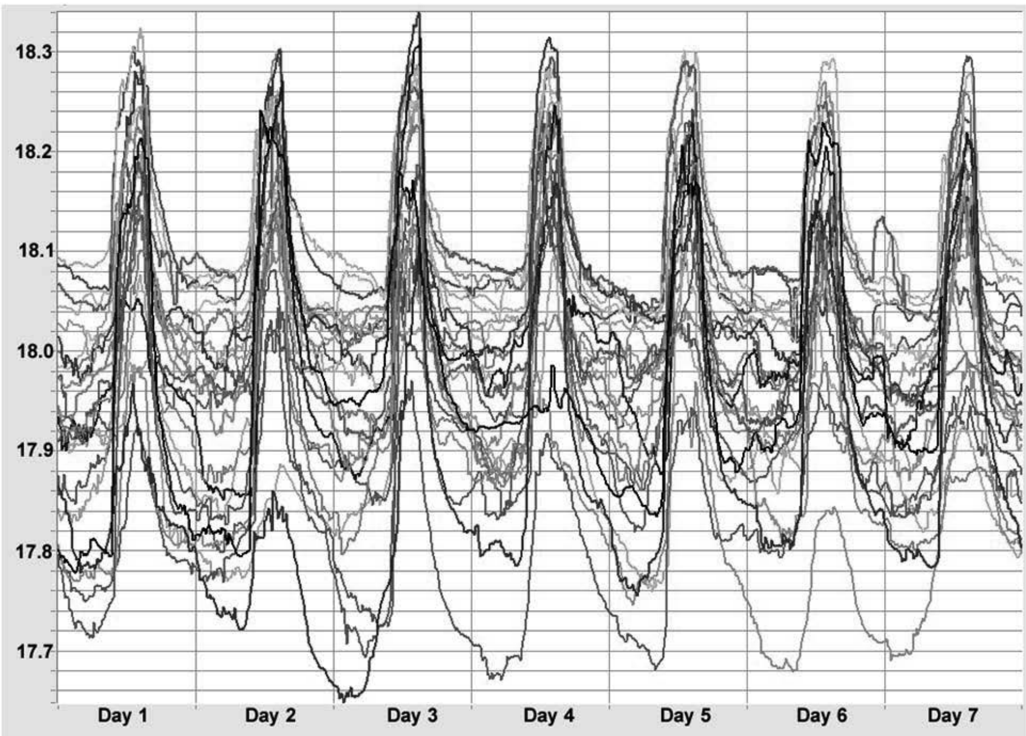

Figure 17 Overlay graph showing 28 weeks of displacement data on top of each other

The 28 months on display in Figure 18 is more than two years of 15-minute data. Those who study long-term sensor behaviour find overlay graphs of tremendous value. This study could be taken a step further by plotting all the 28 months as RoC plots and therefore omitting the long-term drifting but that study has to wait.

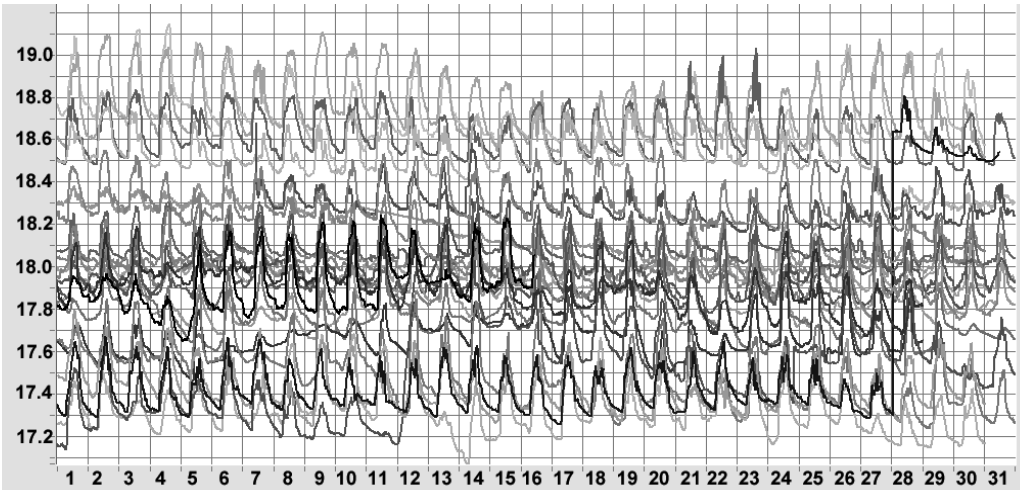

Figure 18 Overlay graph showing 28 months of displacement data on top of each other

4 Conclusion

Effective data handling in geotechnical projects, and in any datalogging projects as well, require the use of powerful software tools which will allow a single operator, or a small group of operators, to perform all the tasks needed. This include the task of presenting data in a meaningful way to make it easy to understand the situation, the task of alarm handling of sensors and systems, the task of performing studies that will revile if all is well or if there is need to act and the task of reporting. High-performance software systems will replace manual methods and offer powerful data handling that is of tremendous value to operators and their clients with its lower overall cost of any data monitoring project, shorter time that is needed every day for overview of the measured data and shorter time needed for generating reports. Some examples of effective data handling are discussed in this paper and the papers mentioned in the references.

References

Thorarinsson, A 2007, 'Methods for automatic storage, visualization and reporting in datalogging applications', in J DiMaggio & P Osborn (eds), *Proceedings of the Seventh International Symposium on Field Measurements in Geomechanics*, American Society of Civil Engineers, Reston, VA, pp. 1-14.

Thorarinsson, A & Simmonds, A 2011, 'Real time, on-line monitoring for every project', *Proceedings of the Eighth International Symposium on Field Measurements in Geomechanics*, Technische Universität Braunschweig, Berlin.