VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui

School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

An Investigation into Alternative Solutions to ESPAC for Assessing Earthquake Liquefaction Potential

George Charles Davie

Supervisor: William Neil L Browne

Submitted in partial fulfilment of the requirements for
Bachelor of Engineering with Honors (Software).

Abstract

Standard regression techniques for soil analysis are time consuming and subject to human bias. The Institute of Geological and Nuclear Sciences (GNS) and Victoria University have developed a proof of concept prototype as a solution to this problem, Evolutionary Spatial Auto-Correlation (ESPAC). ESPAC is an implementation of the parallel linear genetic programming (PLGP) algorithm. Due to the nature of this project no alternative solutions were investigated. This project aims to address this issue by investigating the properties of other Evolutionary Computation based artificial intelligence techniques. The aim to lend strength to the argument that PLGP is the correct solution to the problem, or find that an alternative method would be more viable.

Contents

Chapter 1

Introduction

Earthquakes are known to cause severe damage to human life and infrastructure. A significant portion of the problems associated with an earthquake event are due to ground instability. A developed site can have poor ground stability for a number of reasons. The one this body of work is addressing is liquefaction. Liquefaction occurs when sediments, silts and sands below the water table lose their structural strength during an earthquake. The sediments float within the compressed water causing the strength of the stratigraphic layer to fail. At which point it can no longer support the layers above it causing them to sink [1], [2].

Techniques to determine the probability that a site will liquefy during an earthquake event have been around for a while. Most of these involve the study of core samples. These are destructive to the site, expensive and time consuming to obtain. However a technique called SPAC was developed to measure surface micro-tremors which significantly reduces the problems associated with the recovery of core samples [2], [3], [4], [5], [6]. This process is still fairly labor intensive, though it can be done off-site, as it currently uses the simplex algorithm which requires a lot of human operator intervention. The simplex algorithm is used during the modeling of the data as part of an iterative process with the goal of finding the best fitness [2], [6].

This project is building upon Aaron Scoble's work he submitted last year in completion of his summer project and honors research. His work examines the use of AI, specifically Parallel Linear Genetic Programming (PLGP) to solve the current human problems associated with the simplex algorithm [7]. Essentially the current simplex algorithm is very labor intensive and prone to human bias and assumption. Developing a processes that would provide accurate reliable results with a reduction in human processing time would be a beneficial tool to the geotechnic industry. The benefits an AI system could achieve include:

- A reduction in the cost of site evaluations by lessening a geotechnic engineers (geotechs) billable hours.
- A reduction in the time to analyse the data, resulting in faster evaluations.
- A more transparent result that is devoid of human bias, allowing greater client confidence.

There are also several academic contributions that an AI solution could achieve, these include:

- Improved regression analysis within hard and soft constraints.

- An AI solution to an industry problem reinforcing the benefit of further AI research.

Aaron's work investigates the use of parallel linear genetic programing (PLGP) to achieve these goals. This project uses his algorithm as a benchmark with which to test several other AI techniques in an effort to determine which are better suited to this particular regression problem. PLGP is a complex algorithm that does not lend itself to transparency nor has it seen much use in commercial projects. Using a more widely known and understood technique has the potential to be quicker to implement and maintain while providing a higher degree of client confidence. The outcome of this paper is not to have a polished industry ready solution but rather to have determined which direction would be most profitable for further research.

The success of each implementation will be measured as follows:

- Run speed of the implementations.
- Performance in the detection of potential liquefaction risks.

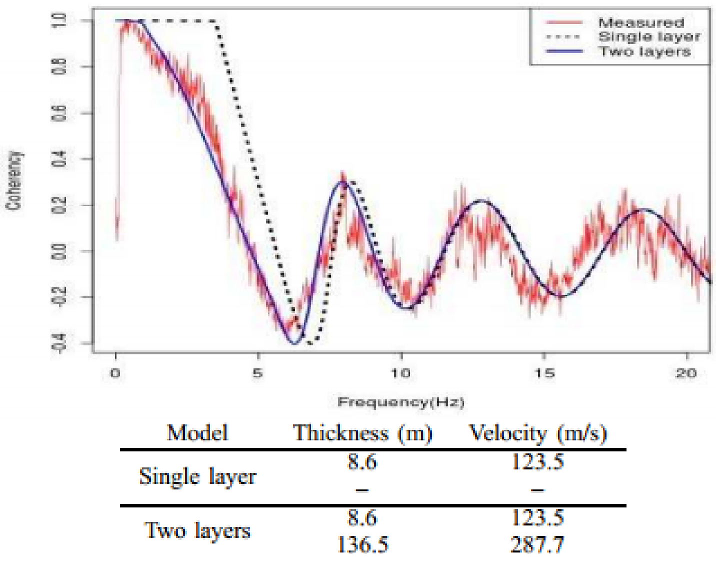
Chapter 2

Previous and Related Work

2.1 Related Work

2.1.1 Spatial Auto-Correlation (SPAC)

Soil liquefaction refers to the process of the saturation of unconsolidated sediments usually by water to the point which the sediments take on liquid like properties. This phenomenon is usually induced during an earthquake event. Sediments, silts and sands below the water table lose their structural strength during an earthquake as they compress the water filled spaces building up pressure to the point that the sediments start to float. At which point the soil layer can no longer support the material above it causing the surface layers to sink [3] [4] [5]. Spatial Auto-Correlation (SPAC) is a geophysical method that records micro tremors caused by Rayleigh waves to calculate the shear-wave velocity of a stratigraphic column. A low shear wave velocity through a soil layer indicates a very porous material that when submerged below the water table has a high potential to liquefy [2] [6] [8].



Model	Thickness (m)	Velocity (m/s)
Single layer	8.6 –	123.5 –
Two layers	8.6 136.5	123.5 287.7

Figure 2.1: GP program tree

2.1.2 The Computer Programs in Seismology (CPS)

The Computer Programs in Seismology (CPS) is a software suite containing a number of modules that are able to analyse seismic data like the measured Rayleigh wave data collected using the SPAC method [9]. The suit contains a model of the earths stratigraphic data, specifically, approximate thickness and shear wave velocity data. The modal summation software module calculates a dispersion curve which is then passed to an inversion process, resulting in a theoretical coherency curve which is matched for errors against the data obtained in the field.

2.1.3 Simplex Algorithm

For any linear programming problems the Simplex algorithm is able to deliver a solution. This is the method that is currently used by Geo-technical Engineers in determining the best fit with the theoretical coherency function [2]. The Simplex algorithm with the aid of an operator continuously calls the CPS suit refining the model until no further improvements can be made. It is this process that is slow, time consuming and currently requires heavy human involvement. Figure 2.1 shows a curve being refined with the addition of a second layer.

2.1.4 Genetic Programming for Symbolic Regression

When a model of the curve of best fit is the required objective of a problem you have a regression problem. Therefore as we are using a measured coherency as our target and are creating theoretical coherency curves using the CPS suite we have a regression problem [10] [7]. The accuracy of this theoretical curve against the measured coherency of specific sites is the fitness function for our evolutionary processes.

2.1.5 Genetic Programming (GP)

Genetic programming (GP) is an evolutionary algorithm that given data and a fitness function mutates a function to produce a fitness result. This nature makes it an ideal candidate for regression problems [11] [7]. Each evolution of the program is matched against the fitness function and using an elitism technique a subset of the total population is mutated into a new population. Typically a GP programs runs with a tree like structure as can be seen in figure 2.2 [10]. This continues for a predetermined number of cycles and the results of the final elitism selection are returned.

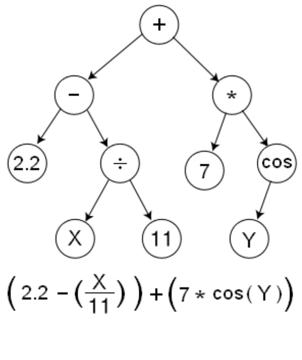

+
−
*
2.2
÷
7
cos
X
11
Y
$\left(2.2 - \left(\frac{X}{11}\right)\right) + \left(7 * \cos(Y)\right)$

Figure 2.2: GP program tree

2.1.6 Linear Genetic Programming (LGP)

Linear genetic programming (LGP) is similar in its operation to GP however rather than the tree structures that the GP produces, LGP has a graph based functional structure. This means that previously executed populations are able to be recycled into generations other than their direct children. While this is powerful it does mean minor changes to previous populations can cascade down to younger generations, which can cause large disruptions in the data [11] [12] [13].

2.1.7 Parallel Linear Genetic Programming (PLGP)

Parallel linear genetic programming (PLGP) limits the dependencies the instructions have between one another. PLGP essentially has a number of LGPs that run separately to one another. Crossovers and mutations are made across the LGPs rather than vertically over themselves. This limits the mutations to equivalent generations. The results is subpopulations that may mutate within themselves but not between populations. Upon completion the final results are added together across the LGPs [14] [15] [16] [17].

2.2 Previous Work

2.2.1 Aaron Scoble's Contribution

Aaron Scoble successfully implemented the ESPAC system to solve a real world regression task. This achievement was a first and major contribution for PLGP [7]. The problem itself was to create a model that could reliably estimate the liquefaction potential of a specific site during an earthquake event using an autonomous process. The benefits and reasons for this work have previously been outline within Chapter 1.

Aaron's work used a number of assumptions and simplifications that will be carried over to this project due to the scope. However it needs to be noted that these need to be addressed before any of the techniques can be commercially deployed. The main three are:

- A fixed value of 2 for the number of layers present within each experiment. Aaron suggests a possible solution to this within his work.
- The assumption that each of these layers are perfectly horizontal and that no folding or other type of geological deformation has occurred.
- The assumption that libCPS and CPS are functioning correctly and are current.

Another major contribution of Aaron's work is the libCPS library that successfully wraps the existing CPS modal summation module into a Java Native Interface (JNI) library [7]. This will be used through each of the Java AI implementations within the project.

Chapter 3

Progress

3.1 Geological Background

During the mid-semester break there was a trip to Nelson as part of an introductory field geology paper, ESCI-241. The relevancy of this exercise was to conceptualize the gathered data the various AI techniques would be working upon. This trip gave a lot of insight into understanding the scale of various stratigraphic layers and their composition. It also brought up a number of points that are out of scope for this project but will need addressing before any of the proposed solutions are industry ready.

The main point being that the contact between layers is rarely horizontal. Over an aperture distance of 20-30 meters the reading of these contacts would become distorted, producing data that is hard to accurately model using the proposed AI solutions. This problem exists within the current human operated simplex method but an experienced technician will have relevant structural background to understand these features that the AI solutions would find hard to model. This leads further credence to the need to fix a number of simplifications used within previous work.

Another generalization that this work and previous work uses is that there are only ever two layers. This excursion demonstrated that while there are often only one or two layers that would have an effect on the surface layer, the current work still does not support this variance. Examples of a number of sites that had more than 2 layers were also witnessed. This is the easier of the two generalization to amend.

3.2 Implementation of Existing Work

A major achieved milestone was the handover of the existing work and the familiarization with the system to the point where the past experiments can be repeated. This has involved updating the existing implementation to work on the updated ECS system that is now running Java 7 rather than 6. A refactoring of several duplicated classes has occurred to make the system more readable and allow the extraction of the libCPS library. Scripts have been successfully written to allow batch testing and experiment duplication over the grid.

3.3 Benchmark Data Acquisition

In order to successfully compare the different AI methods a substantial amount of data needs to be collected. The data that Aaron collected from the PLGP is not sufficient to fairly contrast the other techniques against. Due to time constraints and the slow nature of libCPS, generations were limited to 50. The oscillation phenomena Aaron experienced may be reducible using a larger number of generations [7].

Aaron acknowledges his work as a proof of concept and does not provide rigorous testing of the ESPAC technique. So a new step within the project has to be to gather a set of statistically strong data. There were a number of decisions related to the configuration of these experiments that needed to be made. As the best operational conditions have yet to be determined for the PLGP implementation additional experiments to contrast a number of different run configurations need to occur. The next section details the following experiments that have been run or are in the process of running to fulfill this.

3.4 PLGP Results

Still waiting on more grid results before the data can be empirically evaluated. Once these results are in a study of the data using both Student's paired sample t-tests and Wilcoxon rank-sum test will be conducted.

3.4.1 Population 100 with 80 Generations

This was the first experiment to have a significant number of runs, 30, on the grid. The numbers were chosen with a focus on determining the speed at which the program executed. These runs took on average 22 hours to complete.

The graphs in Figure 3.1 show a cross section of the data collected for the PLGP running with a population of 100 with 80 generations. It is expected that the variance is due to the small population size and that decreasing the number of generations by half will have little effect on the fitness.

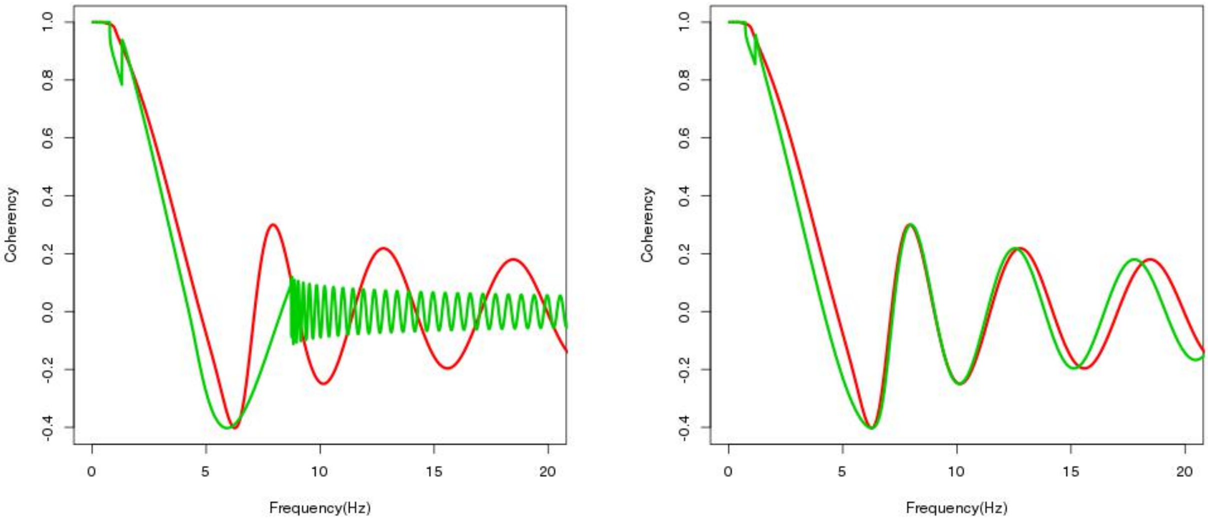


Figure 3.1: Two graphs highlighting the variance found within 30 runs of the PLGP with a population of 100 and 80 generations

3.4.2 Population 100 with 40 Generations

The purpose of this experiment is to determine the difference found between changing the number of generations over a small population.

Results Pending

3.4.3 Population 1000 with 40 Generations

The purpose of this experiment is to determine the difference found between changing the population and to determine the speed of the application. The average run time for these was 4.8 days.

The graphs in Figure 3.2 show a cross section of the data collected for the PLGP running with a population of 1000 with 40 generations. The variance seen here is very similar to that seen with a population of 100 and 80 generations. As we proposed that 100 populations with 40 generations would be similar this indicates our smallest set of runs is providing an optimal solution given the current fitness function.

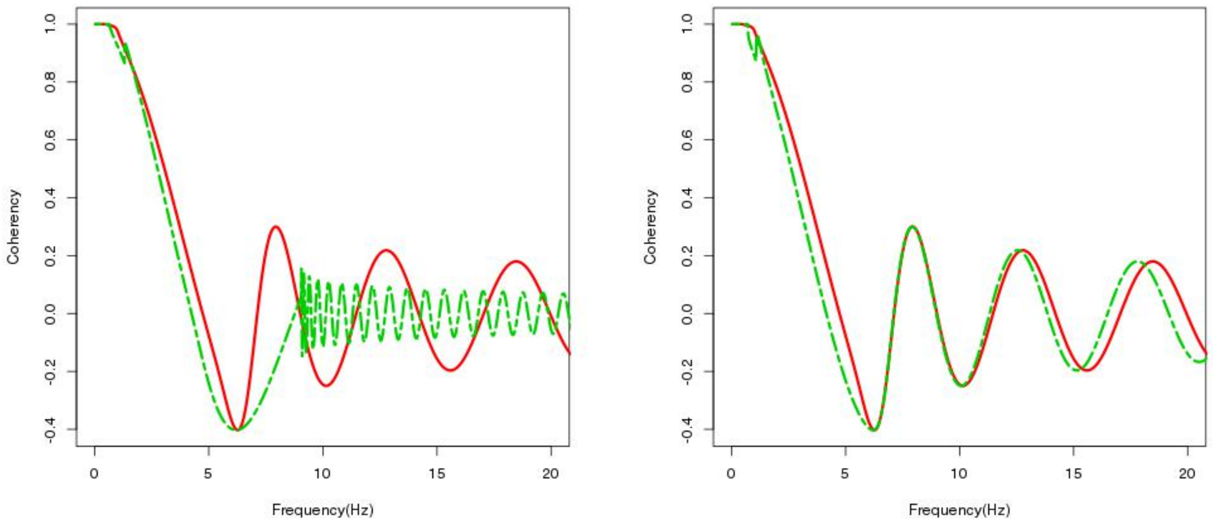


Figure 3.2: Two graphs highlighting the variance found within 30 runs of the PLGP with a population of 1000 and 40 generations

3.4.4 Population 1000 with 80 Generations

The purpose of this experiment is to satisfy our hypothesis that a population of 1000 with 80 generations will have the same variance and local optimization as a run with a population of 100 with 40 generations.

Still waiting on grid results before this hypothesis can be evaluated.

3.5 GA Implementation

A basic GA has been implemented following a number of ECJ tutorials. The purpose of this was to gain an understanding of the library before attempting the full GP. The GA is fairly basic and not suited to the regression problem so out side of this exercise it will not be used or test.

3.6 GP Implementation

Currently just the ECJ tutorial has been implemented. A full implementation of the GP is the next major hurdle that will be accomplished.

Chapter 4

Plans

4.1 Proposal Review

4.1.1 Stage 1

This will primarily be research into the problem itself.This includes the physical science behind it and the algorithms mentioned above. While a portion of this will be continuous throughout the project the bulk should be completed by 22 April with a presentation given to GNS around this time.

Completed but presentation has yet to be given but will be arranged at a time convenient to GNS.

4.1.2 Stage 2

Each of the above techniques will then be further investigated to determine their fit for the proposed problem. This is to remove any redundancy and avoid complication during stage 3. This should be completed by 10 May with the completion of a one page report.

Completed - work can be found within background section.

4.1.3 Stage 3

The third task will be determining the evaluation methods used to benchmark the success of each technique. A fitness function for the given problem will be determined using the pre-existing work and communication with a geotech at GNS. This should be completed by 19 July by GNS acceptance of the documented methods.

This has changed as the fitness function will be the same as that found within Aaron's work. However the collection of benchmark data from Aaron's work has commenced. This will be completed by the 21st of June.

4.1.4 Stage 4

The techniques that meet the problems criteria will then be developed in Java and tested against existing benchmarks/standards a number of times to ensure statistical confidence. These results will be collated into an appendix by 8 September, this will be attached to the final report.

The implementation of these techniques will be completed by 15 July.

4.1.5 Stage 5

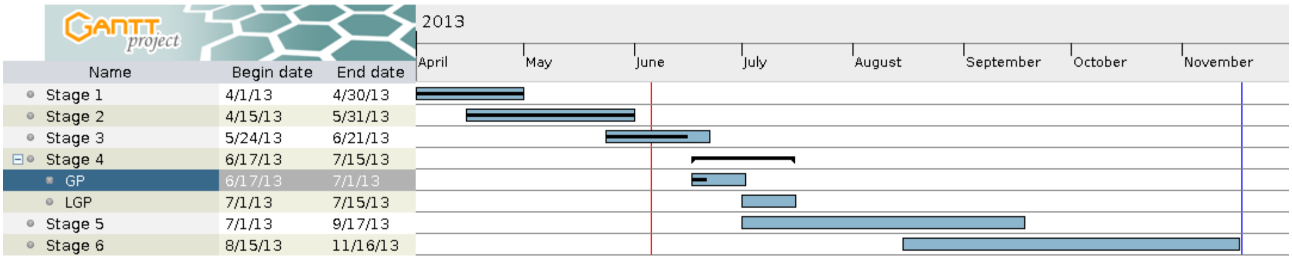
The evaluation techniques that were developed during stage 3 will then be tested against the remaining algorithms and the existing parallel linear genetic programming solution. Again these tests will be run multiple times to ensure statistical confidence and be written into an appendix by 27 September.

4.1.6 Stage 6

The final stage will be to ensure all the resources used and developed during the project are in a presentable format. This includes a written report, oral presentation and a tidy code repository for any future work.

Written Report - 18 October
Oral Presentation - 16 November
Tidy Code Repository - 16 November

4.2 Gantt Chart



Name	Begin date	End date
Stage 1	4/1/13	4/30/13
Stage 2	4/15/13	5/31/13
Stage 3	5/24/13	6/21/13
Stage 4	6/17/13	7/15/13
GP	6/17/13	7/1/13
LGP	7/1/13	7/15/13
Stage 5	7/1/13	9/17/13
Stage 6	8/15/13	11/16/13

Figure 4.1: Gantt Chart Created 3 June 2013

4.3 AI Libraries

Rather than reinventing the wheel there already exist a number of Java libraries built specifically for the implementation of genetic programming programs. Within Victoria University there has been a substantial amount of work done using the ECJ library. Due to this background and the amount of documentation available this will be used in the creation of the GP. There are currently no open source Java LGP libraries. Slash-A is the only out of the box LGP supporting library that has been developed and released but this is written in C++ and creating a new libCPS wrapper would be redundant. Carlton Downey has produced a Java LGP but it is currently not in the public domain. If this software cannot be obtained a LGP library will be built specifically for this research.

Chapter 5

Summary

5.1 Achievements to Date

There have been a number of achievements to date. All of the background material has been covered and researched. This included research on existing software libraries and evaluating which ones are most likely to fit the problem. The best solutions that are currently available are the George Mason University's GP package named, ECJ and Carlton Downey's LGP library if this can be obtained.

Aaron Scoble's existing work has been handed over and modified to work on a slightly different architecture. Benchmark data from this is being gathered and is near completion with a wait on a number of grid jobs awaiting that are still executing.

5.2 Major Hurdles Remaining

There is a significant amount of work left within the project with the next major hurdle being the implementation of the GP. This will be closely followed by the implementation of the LGP. Once these are complete experiments will be run to determine each of the algorithms most efficient operating parameters. Data will then be collected with a heavy reliance on the ECS grid system. Once all this data is collected a statistical evaluation of the results will be written into a report. This report will become the results and discussion sections of the final write up which will be completed by the end of the year.

5.3 Contribution Goals

The contributions of the project will be an implementation of both a GP and LGP used to determine the liquefaction potential of horizontal 2 layered stratigraphic records obtained during a site investigation using the SPAC system. These two systems will also be used in a study also featuring the PLGP to determine which of the three techniques is the most successful at solving the problem of liquefaction potential detection.

Bibliography

[1] C. Downey and Z. Mengjie, "Parallel Linear Genetic Programming," in *Victoria University Wellington*, 2011.

[2] W. Stephenson, P. Barker, Z. Bruce and R. Beetham, "Immediate Report on the Use of Microtremors (SPAC Measurements) for Assessing Liquefaction Potential in the Christchurch Area," in *GNS Science, Tech. Rep.*, 2011.

[3] K. Aki, "Space and time spectra of stationary stochastic waves, with special reference to microtremors," in *Bull. Earthquake Res. Inst. Univ. Tokyo.*, 1957.

[4] H. L. H. Cox and K. Bell, "Spatial correlation for directional sensors in arbitrary noise fields," in *in Proceedings of the 43rd Asilomar conference on Signals, Systems and Computers.*, 2009.

[5] D. R. P. Apostolidis and K. Pitilakis, "The use of micro-tremors for the definition of soil properties and bedrock depth in an urban area," 2004.

[6] R. Beetham, W. Stephenson, P. Barker, and N. Perrin, "a non-invasive site investigation method for determining site class from microtremor records," in *in Proceedings of the 11th IAEG Congress*, 2010.

[7] A. Scoble, W. Brown, and M. Zhang, "Evolutionary Spatial Auto-Correlation for Assessing Earthquake Liquefaction Potential using Parallel Linear Genetic Programming," in *Victoria University Wellington*, 2013.

[8] P. B. W. Stephenson and R. Beetham, "Microtremor based determination of shear-wave velocity of soft soils using the spac method,," in *GNS Science, Tech. Rep.*, 2011.

[9] R. B. Herrmann, "Computer programs in seismology," in *Saint Louis University, USE, 2002.*, 2011.

[10] J. R. Koza, "Genetic programming: On the programming of computers by means of natural selection,," in *Cambridge, Mass.: MIT Press,*, 1992.

[11] W. B. L. R. Poli and N. F. McPhee, "A field guide to genetic programming," in *ISBN 978-1-4092-0073-4*, 2008.

[12] C. Downey and M. Zhang, "Parallel linear genetic programming," in *in Proceedings of the 14th European Conference on Genetic Programming*, 2011.

[13] M. Z. C. Downey and J. Liu, "Parallel linear genetic programming for multi-class classification," in *Genetic Programming and Evolvable Machines*, 2012.

[14] M. Brameier and W. Banzhaf, "Evolving teams of predictors with linear genetic programming," in *in Genetic Programming and Evolvable Machines.*, 2001.

[15] M. Brameier and W. Banzhaf, "A comparison of linear genetic programming and neural networks in medical data mining," in *IEEE Transactions on Evolutionary Computation*, 2001.

[16] C. Fogelberg and M. Zhang, "Parallel linear genetic programming for multi-class classification," in *in AI 2005: Advances in Artificial Intelligence*, 2005.

[17] L. T. G. Olague, E. Romero and B. Bhanu, "Multi-class object recognition based on texture linear genetic programming," in *Applications of Evolutionary Computing*, 2007.

[18] J. A. Nelder and R. Mead, "A simplex method for functional minimization," in *The Computer Journal, vol. 7, no. 4, pp. 308-313,*, 1965.

[19] R. D. Andrus and K. H. I. Stockoe, "Liquefaction resistance of soils from shear-wave velocity," in *Geotechnical and Geoenvironmental Engineering*, 2000.

[20] F. G. L. G. R. Harik and D. E. Goldberg, "The compact genetic algorithm," in *IEEE Transactions on Evolutionary Computation*, 1999.

[21] N. Hansen and S. Kern, "Evaluating the CMA Evolution Strategy on Multimodal Test Functions," in *Swiss Federal Institute of Technology (ETH) Zurich, Switzerland*, 2004.

[22] N. Hansen, "The CMA Evolution Strategy: A Comparing Review," in *Swiss Federal Institute of Technology (ETH) Zurich, Switzerland*, 2006.

[23] P. Lanzi and R. Riolo, *A Roadmap to the Last Decade of Learning Classifier System Research (From 1989 to 1999)*. Lecture Notes in Computer Science, 2000. ISBN 978-3-540-67729-1.