COMPUTER AIDED SLOPE STABILITY ANALYSIS USING OPTIMIZATION AND PARALLEL COMPUTING TECHNIQUES

By

MOHAMMAD TABARROKI

Thesis submitted in fulfillment of the requirements
for the degree of
Master of Science

July 2013

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my supervisor, Professor Fauziah Ahmad for her kindness, guidance, and support. Her willingness to oversee this research and dedication to making me more serious is greatly appreciated.

A Special thanks to my co-supervisor, Associate Professor Ahmad Shukri Yahya who advised me on the statistical and mathematical aspects of this work.

I am also grateful to Dr. Curtis Kelln and Dr. Chris Kelln from GEO-SLOPE, for their valuable comments and discussions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AFSA	Artificial fish swarms algorithm
CEC	Congress on evolutionary computation
CoDE	Composite differential evolution
DMS-PSO-SHS	Dynamic multi-swarm particle swarm optimizer with subregional harmony search
EA	Evolutionary algorithm
FE-SRM	Strength reduction finite element methods
FOS	A function that calculates FS
FS	Factor of safety
GA	Genetic algorithm
GLE	General limit equilibrium method
HS	Harmony search
HS/PSO	Coupled particle swarm and harmony search
LEM	Limit equilibrium methods
MATLAB	Programming language
MGSA	Modified gravitational search algorithm
MHS	Modified harmony search algorithm
MPSO	Modified particle swarm optimization
NCDE	Neighborhood based crowding differential evolution
NFE	Number of function evaluations (number of trials)
NFL	No free lunch
NSDE	Neighborhood based speciation differential evolution
OMS	Ordinary method of slices
PSO	Particle swarm optimization

SLOPE/W	Commercial software for slope stability analysis
USD	United State of America Dollar
PC-SLOPE	Slope stability software
SLOPE-II	Slope stability software
GEO-SLOPE	Slope stability software
SLOPE	Slope stability software
GUI	A graphical user interface
TOL1	Tolerance of FS (high value)
TOL2	Tolerance of FS (low value)
LB	Lower bound
UB	Upper bound
maxIter	Maximum iteration

LIST OF SYMBOLS

c	Cohesion
E	Inter-slice normal force
f	The perpendicular offset of the normal force from the center of rotation
R	the radius for a circular slip surface
u	Pore-water pressure
W	Slice weight
X	The inter-slice shear force
X_L	Left side inter-slice shear force
X_R	Right side inter-slice shear force
α	Inclination of slice base
N	Normal force at the base of slice
β	The base length of each slice.
λ	Scaling factor for inter-slice function
ϕ	Friction angle
ε	Tolerance

ANALISIS KESTABILAN CERUN BERBANTU KOMPUTER MENGGUNAKAN TEKNIK PENGOPTIMUMAN DAN PENGKOMPUTERAN SELARI

ABSTRAK

Kaedah keseimbangan had (LEM) merupakan kaedah yang digunakan secara meluas untuk meramal kestabilan cerun. LEM membolehkan pengiraan faktor keselamatan (FS) untuk beberapa percubaan kegelinciran permukaan di mana FS minimum dilaporkan untuk kegelinciran permukaan kritikal. Teknik pengoptimuman global heuristik telah digunakan kerana masalah dalam pencarian kegelinciran permukaan adalah dikenalpasti sebagai *NP-hard (non-deterministic polynomial-time)*. Walaupun teknik-teknik tersebut secara umumnya menghasilkan keputusan yang baik, masalah berkaitan theorem *"No Free Lunch" (NFL)* dalam pencarian kegelinciran permukaan kritikal adalah tidak berkesudahan. Menurut theorem NFL, tiada teknik optimum heuristik dapat menyelesaikan semua masalah dengan baik. Walau bagaimanapun, kegelirinciran permukaan lain berkemungkinan besar memainkan peranan yang penting seperti kegelinciran permukaan kritikal dalam analisis praktikal. Kegelinciran permukaan memainkan peranan yang penting sekiranya ia menjauhi gelinciran permukaan kritikal; namun, sekiranya FS menghampiri FS minima atau kesan serius dari kegagalan sepanjang gelinciran permukaan. Hal ini memerlukan implementasi yang berterusan dan teknik pengoptimuman yang berlainan. Namun yang demikian, implementasi dalam teknik pengoptimuman untuk meramal kestabilan cerun tidak selalunya lancar kerana memerlukan pautan luaran ke LEM. Kajian telah dijalankan untuk merungkai isu ini dengan membangunkan algoritma *"decoupled"* yang membolehkan teknik pengoptimuman dilaksanakan dengan mudah. Kajian ini seterusnya menunjukkan keringkasan algoritma yang dibangunkan

dan mempromosikan kajian terkini dalam kestabilan cerun di mana tiga teknik pengoptimuman terkini telah diimplemetasi dan kecekapan serta keberkesanan dalam mengesan minima sejagat dan minima tunggal/berbilang telah dikaji dalam siri kajian masalah. Teknik pengoptimuman yang telah digunakan adalah *composite differential evolution* (CoDE), *neighborhood based crowding differential evolution* (NCDE), dan *neighborhood based speciation differential evolution* (NSDE). CoDE telah digunakan untuk mengesan kegelinciran permukaan kritikal tunggal *(single-modal)* manakala NCDE dan NSDE telah digunakan untuk mengesan kegelinciran permukaan kritikal berbilang *(multi-modal)*. Tambahan lagi, satu versi *multithreaded* untuk algoritma *decoupled* telah dibangunkan untuk meminimumkan masa untuk menganalisis menggunakan teknik perkomputeran selari. Analisis daripada *single-modal* menunjukkan bahawa CoDE boleh mengesan kegelinciran permukaan kritikal dengan baik. Ia juga mempunyai kecekapan dan memerlukan kurang percubaan jika dibandingkan dengan kajian yang telah dijalankan sebelum ini. Analisis daripada *multi-modal* menunjukkan bahawa prestasi NCDE adalah lebih baik berbanding NSDE dalam mengesan pelbagai kegelinciran permukaan penting. Selain itu, perkomputeran selari mampu mempercepatkan pengiraan. Namun yang demikian, jurang di antara kelajuan ideal dengan kelajuan sebenar meningkat dengan peningkatan CPU utama.

COMPUTER AIDED SLOPE STABILITY ANALYSIS USING OPTIMIZATION AND PARALLEL COMPUTING TECHNIQUES

ABSTRACT

Slope stability analysis is commonly performed using limit equilibrium methods (LEM). In LEM, factor of safety (FS) is calculated for different trial slip surfaces and the one with the minimum FS is reported as the critical slip surface. Since locating the critical slip surface is believed to be an NP-hard (non-deterministic polynomial-time) problem, heuristic global optimization techniques are employed. Although these techniques have usually produced good results, “No Free Lunch” (NFL) theorems seem to have made the problem of locating the critical slip surface an endless research. According to the NFL theorems, no heuristic optimization technique can perform well for all problems. On the other hand, there may exist other slip surfaces that are as important as the critical slip surface in practical analyses. A slip surface is important, if it is located far away from the critical slip surface, but gives FS close to the minimum FS or the consequences of failure along the slip surface is serious. Therefore, there is a need to constantly implement and test different optimization techniques. However, implementation of optimization techniques in slope stability analysis is often not straightforward because it requires internal links to LEM. Firstly, the present study resolves this issue by developing a decoupled algorithm that allows for easy implementation of optimization techniques. Then, to demonstrate the simplicity of this algorithm and to promote the latest research on slope stability, three state-of-the-art optimization techniques are implemented, and their effectiveness and efficiency in detecting single/multiple global and local minima is investigated on a series of test problems. The employed optimization techniques are composite differential evolution (CoDE), neighborhood

based crowding differential evolution (NCDE), and neighborhood based speciation differential evolution (NSDE). CoDE is used for locating single critical slip surface (single-modal), while NCDE and NSDE are used for locating multiple critical slip surfaces (multi-modal). Moreover, a multithreaded version of the decoupled algorithm is developed to reduce the analysis time using parallel computing technique. The results of single-modal analyses show that CoDE can effectively locate the critical slip surface of the investigated test problems. It is also very efficient and requires less trials compared to the previous works. The results of multi-modal analyses show that NCDE performs better than NSDE in locating multiple important slip surfaces. It is also shown that parallel computing can speed up the calculations. However, the gap between the ideal speedup and the actual speedup increases with increasing the number of CPU cores.

CHAPTER 1

INTRODUCTION

1.1 Background

The first article on the use of computers in slope stability analysis appeared in 1958 (Little & Price, 1958), which described the implementation of the Bishop method (Bishop, 1955) on the English Electric DEUCE computer. This program was able to analyze 200 circles in less than ½ hour and successfully used in the design of many projects, most notably Mangla dam (the world's 5th largest earth dam), which is said that about 20,000 circular slip surfaces were analyzed (Vaughan et al., 2004). Over the years, the number of computer programs for performing slope stability analysis increased. Whitman and Bailey (1967) estimated that from 25 to 50 programs had been written until 1967 in the United States. In their famous paper, they also talked about the role of man-machine communication and how an engineer "might wish to perform slope stability analyses using a computer". Although what they had described was like a "science fiction" story (Duncan, 1996) in 1967, the advent of personal computers in 1980s turned it into a reality for most of the engineers.

Computers have also helped to develop more accurate methods for non-circular slip surfaces (e.g., Morgenstern & Price, 1965; Spencer, 1967; Janbu, 1973). This can in part be traced to the development of high-level programming languages that enable researchers to concentrate on the solution strategy and program design without having to deal directly with the internal details of a specific hardware. Moreover, the cost of computing and storage are falling steadily. For example, at the time of writing this thesis (2013), tablet computers with the price of 500 USD and weight of 650 g, can analyze thousands circular slip surface using Bishop method within 12 sec (Rickard & Sitar, 2012). This is a tremendous achievement when compared with the DEUCE computer in 1958, which with the price of 50,000 GBP (~78,600 USD) and weight of 1193 kg could analyze about 200 circular slip surface using Bishop method within 30 min (Barrett, 2004).

1.2 Problem Statement

Although slope stability analysis has become more elaborate and successful with improvements in computer technology, the issue of locating the slip surface with the minimum FS is still believed to be an NP-hard (non-deterministic polynomial-time) problem (Cheng et al., 2012). It means that there does not exist any algorithm that can find this critical slip surface among many trial slip surfaces in a polynomial time. In other words, the CPU time increases rapidly as the dimension of the search spaces increases, and it becomes impossible to explore all trial slip surfaces. For this reason, heuristic global optimization algorithms are employed to find, with no guarantee, the critical slip surface in an acceptable time (e.g., Kahatadeniya et al., 2009; Dong et al., 2010; Khajehzadeh et al., 2011b; Cheng et al., 2012). These algorithms have usually produced good results that are believed to be near to the actual critical slip surface.

However, the lack of guarantee in the quality of results and the “No Free Lunch” (NFL) theorems (Wolpert & Macready, 1997) seem to have made the problem of locating the critical slip surface an endless research. According to the NFL theorems, no optimization algorithm can perform well for all types of problems. A quick literature review shows that this statement holds in slope stability analysis. Zolfaghari et al. (2005) used genetic algorithm (GA) to find the critical slip surface of a slope with complex soil layering, and found the minimum FS=1.24. Later on, Cheng et al. (2007) employed particle swarm optimization (PSO) for the same slope and reported a more critical slip surface with FS=1.11. While the performance of PSO for this slope was promising, Cheng et al. (2008) encountered a cross section of an important dam project for which the PSO produced poor result (min. FS=3.37). They adopted the artificial fish swarms algorithm (AFSA), and minimum FS=2.67. Recently, Cheng et al. (2012) have presented another cross section of this dam project for which neither the AFSA nor the PSO can produce satisfactory results. They have shown that a coupled PSO and harmony search (HS), called HS/PSO, is robust for this problem. The minimum FS reported by PSO, AFSA, and HS/PSO are 2.18, 1.83, and 1.65, respectively.

While research on locating the critical slip surface should remain active, there may exist other slip surfaces that are as important as the critical slip surface in practical analyses. A slip surface is important, if it is located far away from the critical slip surface, but gives FS close to the minimum FS or the consequences of failure along the slip surface is serious. Since the input parameters in slope stability analysis (e.g., shear strengths and unit weights) are often defined with limited accuracy, such slip surfaces may have the same likelihood of failure as the critical slip surface. Therefore, practical designers must consider them, in addition to the critical one, for possible slope stabilization works. Currently, these slip surfaces are detected by running the optimization algorithm several times, each time restricted to a different search space. This approach requires a strong engineering experience to divide the search space so that there exist only one important slip surface in each run of the algorithm, otherwise some of them will be missed. Therefore, in addition to the regular heuristic algorithms, which aim to find a single global optimum, this thesis will also study the usefulness of multi-modal heuristic algorithms in capturing several global and local optima in a single run.

Two previous paragraphs suggest that there is a need to constantly implement and test different optimization algorithms, and employ the suitable one for the target slope. However, implementation of optimization algorithms in slope stability analysis is often not straightforward because it requires internal links to the LEM. For example, upper and lower bounds for parameters that define a trial slip surface in LEM must be known to the optimization algorithm, and more importantly, these bounds may change in each iteration step. Therefore, it is desirable to develop a decoupled algorithm composed of LEM and optimization sub-algorithms so that they communicate through an interface and the internal details of sub-algorithms are encapsulated and hidden from each other. This decoupling strategy allows for easy implementation of optimization algorithms in slope stability analysis.

In addition to the decoupling strategy, utilizing parallel computing can also be of great interest in computer aided slope stability analysis. This is particularly important for very complicated slopes where optimization algorithms require many trials. The need for parallelization became more evident around 2005 when CPU manufacturers began to produce processors with frequencies higher than 4 GHz, and encountered some physical limits such as heat, power consumption, and leakage current. These problems led to a radical change in CPU technology from single-core architectures to multi/many-core architectures, which marked the end of “free ride” era for computer programs. Prior to this, there was no need to redesign the structure of programs and they automatically ran faster with each new generation of CPUs. However, it is no longer possible to rely on this “free ride” and redesigning is necessary to exploit the full performance of multi-core CPUs.

1.3 Objectives of the Study

Considering the issues discussed in the previous section, three objectives are defined for this thesis:

i. To develop a decoupled algorithm for slope stability analysis composed of LEM and optimization sub-algorithms so that they communicate through an interface.

ii. To implement three state-of-the-art optimization techniques in this decoupled algorithm and study their effectiveness and efficiency in detecting single/multiple global and local minima.

iii. To develop a multithreaded version of this decoupled algorithm and study the performance benefits of using parallel computing technique.

1.4 Scope of the Research

Although slope failure mechanisms are three-dimensional in nature, this research adopts two-dimensional modeling for slope stability analysis. As shown by other studies, this simplification is usually on the safe side, i.e., the minimum FS calculated in two-dimensional is lower than the minimum FS calculated in three-dimensional. Moreover, the LEM sub-algorithm of this research considers only weight of soil and piezometric condition as the driving forces that contribute to slope instability, and on the other hand, considers only soil shear strengths (frictional and cohesive forces) as the resisting forces that help to maintain a slope stable. It, however, should be noted that extending the LEM sub-algorithm to consider more parameters such as surcharge loads, seismic forces, and soil nails does not require re-designing the decoupled algorithm developed in this research. Finally, this study performs parallel computing only on multi-core CPUs, and does not utilize the parallel structure of GPGPU (general-purpose computing on graphics processing units).

1.5 Structure of the Thesis

This thesis consists of five chapters. A necessary background of the study, followed by the problem statement and objectives are given in chapter 1 (current chapter). Chapter 2 discusses previous research studies related to limit equilibrium methods and global optimization algorithms. A brief background on parallel computing is also presented. The research methodology that have been employed to approach the thesis objectives are described in Chapter 3. Chapter 4 presents the results and discussions obtained from analyzing a number of test problems. Finally, conclusions of this study and some suggestions for future research are given in Chapter 5.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Evaluating stability of the slopes is a big challenge for geotechnical engineers, even after decades of experience, which unfortunately often has been gained from slope failures. The most common methods for slope stability analyses are limit equilibrium methods (LEM) and finite element methods (FEM). This study focuses on the LEM and its associated part, which is optimization. An optimization run is successful if minimum FS is reached with the given precision within defined number of iterations. To reduce the time of computations, parallel computing is introduced in the last section of present chapter and implementation of it is brought in the next chapter.

2.2 Limit Equilibrium Method

For evaluating stability of slopes by LEM, failure modes are approximated by slip surfaces and factor of safety (FS) of them is calculated. Slip surface in any shape including circular (Figure 2.1) or non-circular (Figure 2.2) that produces the minimum factor of safety is called critical slip surface and its corresponding factor of safety is taken for design purposes.

Only three control variables are needed to represent a circular slip surface: horizontal position of entrance, horizontal position of exist of slip surface on the ground and radius of the circle.

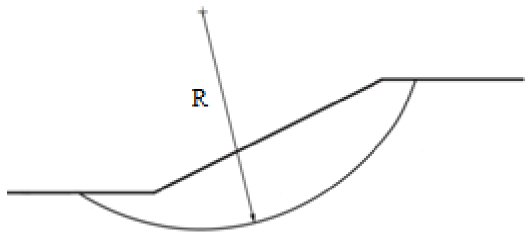

Figure 2.1: Circular slip surface with radius R.

For the non-circular slip surfaces, a set of connected lines with n vertices whose coordinates $\{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\}$ are the unknowns of the slope stability problem, approximate the slip surface. First and last vertices stay on the ground surface of the slope while the others are below it. The vertices are well distanced, so that mutual interferences are avoided (Greco, 1996).

Figure 2.2: Non-circular slip surface

Beside more number of vertices represents the critical slip surface better, by increasing the number of vertices, the degree of approximation of a slip surface increases and optimization algorithms require more trials to find the global minimum. Li and White (1987) proposed a method to generate slip surface with enough large number of vertices without the problem of interferences among vertices. In their method, first search is performed with a few vertices. Then, at the midpoint of each line that joins adjacent vertices, a new vertex is introduced. As the second step, the slip surface of the previous search with added vertices is assumed as the trial slip surface. For combining the speed and robustness of using large number of vertices by the purposed method by Li and White (1987), Greco (1996) used three different numbers of vertices. In the first step, Greco (1996) used four vertices, and then he extended it to 7 and 13 vertices at the third search. He found no improvement in his results after using 13 vertices and commented on sufficiency of 13 or even 7 number of vertices for most practical purposes.

There are other constraints for trial slip surfaces that suggest generating concave upward slip surfaces (Basudhar et al., 1988), however, in the layered soils, due to stratigraphic conditions, part of the slope could be convex and upward concavity constraint can lead to error (Greco, 1996).

To compute the factor of safety, soil mass bounded beneath by the assumed slip surface and above by the ground surface is divided into a finite number of slices , which can be vertical, horizontal, or oblique; however, the vertical slip surfaces are more common (Figure 2.3). Duncan and Wright (2005) discussed the required number of slices for best approximation of FS.

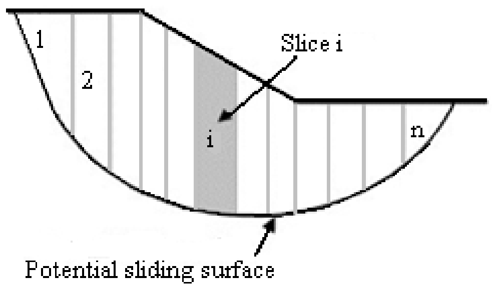

Figure 2.3: Slicing the slip surface vertically in LEM.

An important aspect of slicing is to ensure that only one soil type exists at the bottom of each slice. It can be achieved by using so many slices, which of course is inefficient. Commercial software of SLOPE/W (Krahn, 2004), has simple rules for slicing that are described in some steps here. According to Figure 2.4, the first section starts on where the slip surface enters the ground surface and last section ends where the slip surface exits the ground surface. Other sections occur where:

1) The slip surface crosses the piezometric line,

(2) The slip surface crosses a stratigraphic boundary,

(3) Wherever there is a region point, and

(4) Where the piezometric line crosses a soil boundary.

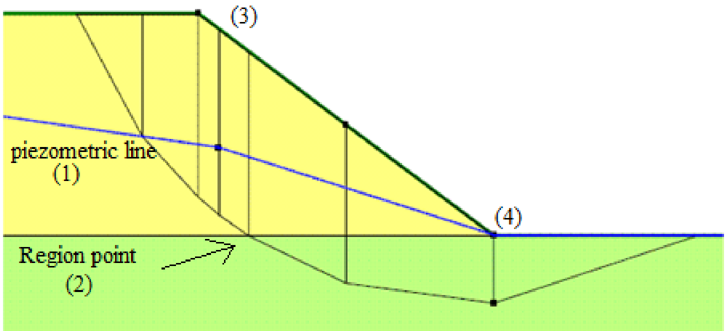

Figure 2.4: Sections in the slice discretization process (After Krahn, 2004)

In the next step, SLOPE/W finds the horizontal distance from slip surface entrance to exit and divides this distance by the number of desired slices specified by the user (the default is 30). In this way, an average slice width is determined. The last step is to compute how many slices of equal width can fit into each section. If very thin slices are generated in the slope, this may introduce some numerical problems in the code and very thin slices must be deleted (Krahn, 2004).

After slicing the assumed slip surfaces, depending on the LEM method that is used, one or both of below static equilibrium equations are written and solved for slices:

1) Equilibrium of forces in the vertical and horizontal direction,

2) Equilibrium of moments about any point.

Duncan and Wright (2005) explained different procedures of LEM and determined the type of the equilibrium equations that are satisfied and assumptions that are made in each method. Some of these LEM methods are listed in Table 2.1. Duncan and Wright (2005) presented a table in which conditions of practical usefulness for each LEM procedure is mentioned.

Table 2.1: LEM methods and equations of statics that are satisfied for each method

Method	**Force Equilibrium**		**Moment Equilibrium**
	X direction	Y direction	
Ordinary method of slices (Fellenius, 1936)	No	No	Yes
Bishop (Bishop, 1955)	No	Yes	Yes
Janbu's simplified (Janbu et al., 1956)	Yes	Yes	No
Spencer (Spencer, 1967)	Yes	Yes	Yes
Morgenstern-Price (Morgenstern & Price, 1965)	Yes	Yes	Yes

2.2.1 General limit equilibrium method (GLE)

A general limit equilibrium method by Fredlund and Krahn (1977) and Fredlund et al. (1981) encompasses all the method in Table 2.1 and works for a range of inter slice forces assumptions. In the GLE, relation between inter-slice forces, which are depicted in Figure 2.5 as E and X, are handled with an equation proposed by Morgenstern and Price (1965) as Equation 2.1 that can be rewritten separately for the right and left forces in Equation 2.2.

$$X = E\,\lambda\, f(x) \qquad (2.1)$$

$$(X_R - X_L) = (E_R - E_L)\,\lambda\, f(x) \qquad (2.2)$$

where:

f(x) = inter-slice force function representing the relative direction of the resultant inter-slice force.

λ = the percentage (in decimal form) of the function used,

E = the inter-slice normal force, and

X = the inter-slice shear force.

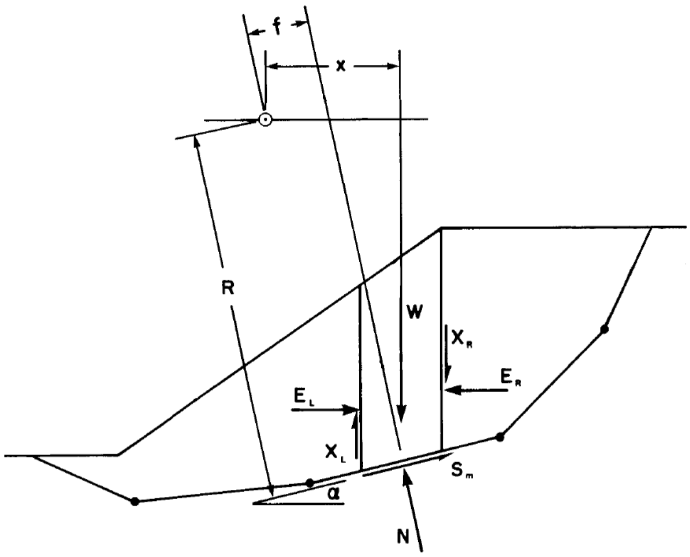

Figure 2.5: Forces acting on a slice through a sliding mass defined by a non-circular slip surface.

The GLE factor of safety equations with respect to moment equilibrium, FS_m, (Equation 2.3) and with respect to horizontal force equilibrium, FS_f, (Equation 2.4) are as bellow. All the summations are from one to the number of slices.

$$FS_m = \frac{\sum (c'\beta R + (N - u\beta) R \tan\varphi')}{\sum Wx - \sum Nf} \tag{2.3}$$

$$FS_f = \frac{\sum (c'\beta \cos\alpha + (N - u\beta) \tan\varphi' \cos\alpha)}{\sum N \sin\alpha} \tag{2.4}$$

where:

c' = effective cohesion

β = the base length of each slice.

R = the radius for a circular slip surface or the moment arm associated with the mobilized shear force for any shape of slip surface.

ϕ' = effective angle of friction

u = pore-water pressure

W = slice weight

x = the horizontal distance from the centerline of each slice to the center of rotation or to the center of moments.

f = the perpendicular offset of the normal force from the center of rotation or from the center of moments. It is assumed that *f* distances on the right side of the center of rotation of a negative slope (i.e., a right-facing slope) are negative and those on the left side of the center of rotation are positive. For positive slopes, the sign convention is reversed.

α = inclination of base slice

N = normal force at the base slice that is calculated by:

$$N = \frac{W - (X_R - X_L) - \dfrac{(c'\beta \sin\alpha + u\beta \sin\alpha \tan\varphi')}{FS}}{\cos\alpha + \dfrac{\sin\alpha \tan\varphi'}{FS}} \qquad (2.5)$$

FS in Equation 2.5 is FS_m or FS_f, depending on N is going to be used in Equation 2.3 or Equation 2.4, respectively. E_R-E_L that is appeared in Equation 2.2 is calculated as follows:

$$(E_R - E_L) = -\frac{(c'\beta - u\beta\tan\varphi')\cos\alpha}{FS} + N(-\frac{\tan\varphi'\cos\alpha}{FS} + \sin\alpha) \qquad (2.6)$$

In the first step of GLE, Lambda (λ) is set to zero for simplicity, so the inter slice shear forces and obviously (X_R-X_L) are zero. Hence, "N = W cosα" and Equation 2.3 will be reduced to Equation 2.6.

$$FS_m = \frac{\sum(c'\beta R + (W\cos\alpha - u\beta)R\tan\varphi')}{\sum W\sin\alpha R} \qquad (2.7)$$

FS_m in Equation 2.7 is calculated directly without any iteration and is equivalent to the Ordinary Method of Slices (OMS). In case of circular slip surfaces, R is constant and it will be removed from Equation 2.7. Since the only unknown parameter in N (Equation 2.5) is FS, FS=1, or FS=OMS, or FS=OMS×1.2, etc are used as guessed FS and FS_m is obtained from Equation 2.3. Then, If $|FS_m - guessed\ FS| <$ tolerance , the solution is converged. Otherwise, obtained FS_m is used as a new guess for FS in Equation 2.5 and the procedure will be repeated until convergence is achieved or failure to converge happens. The same procedure should be done for FS_f using Equations (2.3) and (2.4). In this stage, the obtained FS_m corresponds to Bishop's simplified method and the obtained FS_f corresponds to Janbu's simplified method without any empirical correction. The last stage is to find a λ for which FS_m and FS_f are equal within a defined tolerance.

2.3 Location of the critical slip surface

The process of locating the critical slip surface, which has the lowest FS, can be viewed as the problem of finding the global minimum of a function, called “objective function” in mathematical optimization terminology. Result of optimization algorithms are global and local optima (if there existed any local optima), whether minimum or maximum. Absolute or global minimum is the smallest value that a function takes at a point within entirely function domain (Point G in the Figure 2.6). Local minimums refer to points, which have minimum values of objective function within some neighborhood (Points A, B, C and D in Figure 2.6).

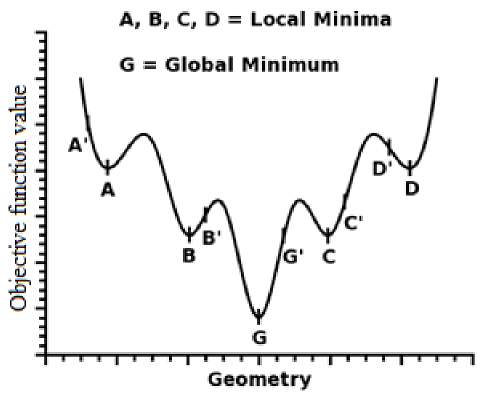

Figure 2.6: Illustration for global and local minimums

Input of global optimization methods for the slope stability problems are trial slip surfaces that try to converge to minimum FS. Inputs of non-circular cases are [x,y] coordinates of a slip surface and for circular slip surfaces are [R,(O_x,O_y)], where R is radius of the circle, and (O_x,O_y) is the center of circle. The optimization process predict a new set of input based on the output of each step, until a set of [x,y] or [R,(O_x,O_y)] is found for which FS is minimized. For circular slip surfaces a wide range of [R, (O_x,O_y)] can be tried, however it is not a good practice to give any [x,y] coordinates as slip surface to the objective function in slope stability, because they may form ill-shaped slip surface (Figure 2.7). Hence, some approaches are proposed by researchers (Cheng, 2003) to generate smooth and concave-upward slip surfaces.

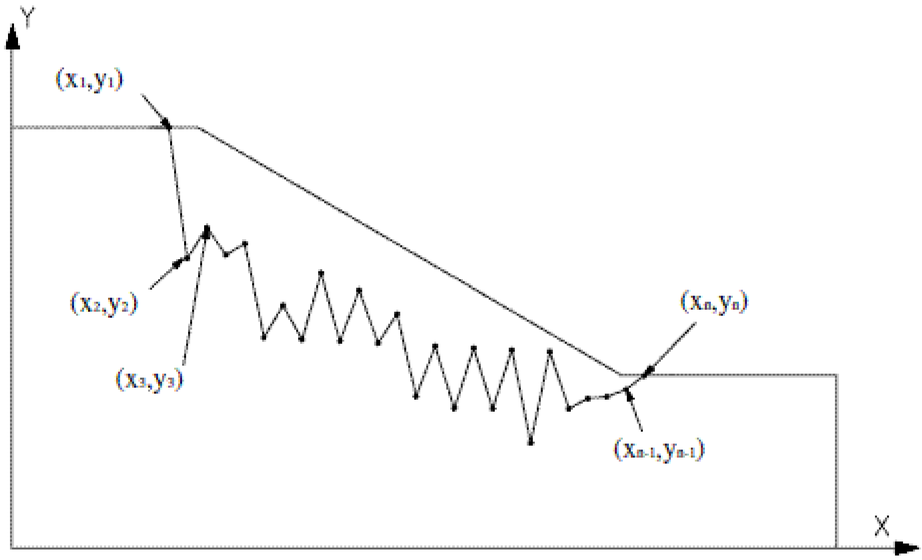

Figure 2.7: A fluctuation slip surface (after Zolfaghari et al. 2005)

In practice, it is usually difficult to have a good initial trial failure surface that is valid for general conditions. Even for the acceptable trial slip surfaces, failure to converge may happen, hence the objective function may not be continuous over the whole solution domain (Cheng et al., 2008). Cheng et al. (2008) developed five procedures for generating slip surfaces and commented on the applicability of each method for a case study with a thin soft layer of soil. Slopes that have irregular shape or soft layers are referred as complex problems are also called "slopes with weak seams". Cheng et al. (2008) noted that factor of safety of such case is very sensitive to the precise location of the critical solution and the differences between the results of the optimization methods for them are large. Figure 2.8 illustrates a slope containing an inclined thin layer which Ching et al. (2009) used to highlight the difficulties of LEM in slopes with weak seams.

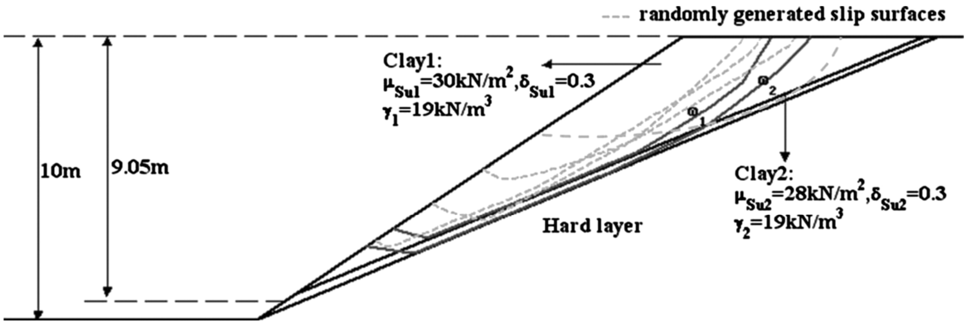

Figure 2.8: Slope with inclined thin clay layer (After Ching et al., 2009)

In different fields of study, so many comparisons between optimization algorithms are made that are varying in their operators and working principles but they are confusing and limited to the test problems used in their study. Some algorithms are complementary to each other and an algorithm, which works satisfactory for one study, may not work in other problems. However, improvement in an optimization method may improve other methods such as the termination criterion that is an important factor in efficiency of optimization algorithms. For optimization algorithms, a suitable number of trials must be defined. While there are no guidelines, researchers have to find optimum number of trials by a trial and error process. If a large number of trials are tried out, a minimum may be found at early stages and the rest of trial and error process does not improve the result (Cheng et al., 2008). The process cannot terminate at early stages because optimization algorithms proceed until a specified number of slip surfaces have been explored. Cheng et al. (2008) introduced a termination criterion for an optimization method called original harmony search (HM) algorithm and modified HM to Modified harmony search algorithm (MHS). In Cheng's et al. (2008) termination criterion, the algorithm is said to have reached quasi- convergence if the difference between the found FS value of the best and the worst is lower than a pre-specified convergence criterion. Modified harmony search by Cheng's et al. (2008) termination criterion is found to be very efficient compared with other global optimization techniques (Cheng et al. 2008). However, when there are more than 25 variables, both original and modified harmony search algorithms require a large number of trials to find a good solution

To begin the literature review from simple algorithms, a good example is application of the Monte Carlo technique for locating the critical slip surface by means of iterative procedures, purposed by Greco (1996) as an easily programmable technique. The benefit of these techniques toward the previous works is its simplicity, as it does not require derivatives of FS. Generation of slip surfaces in Monte Carlo method is based on random numbers and moving to the next step or substitution the coordinates of the vertices toward less FS is random (random walk). Then calculated FS are compared with the best solution of previous step for improvement. Monte Carlo based optimization methods may be divided into two classes: random walk and random jump (Malkawi et al., 2001). The difference between random jump and random walk is that every trial solution in random walk is constructive and it use information of previous trials, but trial slip surfaces in random jump are generated without considering previous achievements.

The drawback of using Monte Carlo techniques in locating critical slip surface is discussed by Greco (1996). In his viewpoint, the method must be used only for finding local minimums especially in layered slopes. For the global minimum, this method must be used iteratively to evaluate all possible local minima and take the lowest as global minima.

More recently, Taha et al. (2010) presented a survey of the literatures on various optimization methods and grouped the optimization methods into two categories: heuristic and second methods that are based on gradient information of objective function and constraints. Following the works of Taha et al. (2010), a list of conventional and modern methods applied to slope stability analyses in the literatures are reviewed in Table 2.2 with a short description of each method. In the following, some of the algorithms that have been demonstrated to be computationally superior to the other optimization algorithms are described in detail.

Table 2.2: Traditional and modern optimization methods to locate the critical slip surface in slope stability

Methods	Authors	Descriptions
Calculus of variations	Baker & Gaber (1978)	Too complicated for complex problems, which do not have regular geometry.
Dynamic programming	Yamagami & Jiang (1997) & Baker (1980)	
Calculus of variations	Celestino & Duncan (1981)	Alternating variable methods to locate the critical non-circular slip surface.
Combination method	Chen and Shao (1983)	Steepest descent, and the Davidson–Fletcher–Powell method in conjunction with a grid search solution
Simplex method	Nguyen (1985)	It has been successful for relatively simple problems
Genetic algorithm	Kirkpatrick et al. (1983)	
Downhill simplex	Bardet& Kapuskar(1989)	Downhill simplex algorithm
RST-2 algorithm	Jade& Shanker (1995)	This optimization is used for Janbu method
Monte Carlo	Greco (1996)	Monte Carlo technique to search for the critical slip surface
Monte Carlo	Malkawi *et al.* (2001)	They adopted the Monte Carlo technique to search for the critical slip surface
Simple Genetic algorithm	McCombie& Wilkinson (2002)	For the circular Bishop method
Simulated annealing algorithm	Cheng (2003)	Transformed the various constraints & the requirement for kinematically acceptable mechanisms on the slip surface to the evaluation of upper and lower bounds of the control variables
leap-frog	Bolton *et al.* (2003)	For the simplified Janbu and Spencer methods
Combination methods	Zolfaghari *et al.* (2005)	They combined the simple genetic algorithm with the Morgenstern–Price method
Improved genetic angorithm	Wan et al.(2005)	Higher efficiency & shorter time than simple genetic anlorithm
Particle Swarm Optimization (PSO)	Kennedy & Eberhart, (1995), Cheng et al. (2007a)	
Harmony search	Cheng et al. (2008)	An improved harmony search more efficient than original harmony search (Geen et al. 2001)
Artificial fish swarm	Cheng et al. (2008)	Artificial fish swarm coupled with the slip surface generation method by Cheng et al. (2007a)
Ant colony algorithm	Kahatadeniya et al. (2009)	By using Morgenstern–Price method

Heuristic global optimization methods, by speeding up the process of finding a satisfactory solution, have attracted the attention of many geotechnical researchers in the recent years. Although heuristic algorithms do not guarantee that the best answer will be found, they are used in slope stability for dealing with optimization of difficult non-continuous, non-convex functions. Heuristic methods imitate natural phenomena like genetics in genetic algorithm by Holland (1975), simulated annealing method by human memory in the Tabu search (Glover, 1989), musical process of searching for a perfect state of harmony in the harmony search algorithm (Geem et al. 2001) and Particle Swarm Optimization (PSO) algorithm that imitates birds' searching food principle.

In May 2005, a technical report for the 2005 IEEE Congress on Evolutionary Computation (CEC 2005) released by Nanyang Technological University of Singapore contained 25 benchmark functions under thoroughly defined experimental and recording conditions. Some optimization algorithms in CEC 2005 by specifying a common termination criterion, initialization scheme, size of the problems, etc were evaluated systematically and the results were published in 13 papers. Scalability studies of CEC 2005 contests demonstrate how the running time/evaluations increase with an increase in the problem size. Some real world problems were also included in their standard test that suite with codes. These benchmark functions were categories in Pseudo-Real Problems and basic, Expanded, unimodal, Multimodal and hybrid Composition functions.

After Cheng et al. (2007a) have carried out detailed comparisons between six major types of heuristic global optimization methods and the sensitivity of these methods under different optimization parameters, they commented that there is no particular optimization method which is superior under all cases.

2.3.1 Composite Differential Evolution (CoDE)

Recently, Wang et al. (2011) have proposed composite differential evolution, CoDE. This algorithm has been tested on all CEC2005 contest instances, and experimental results show that it is very competitive optimization algorithm. Figure 2.9 presents the pseudocode of CoDE algorithm. The inputs of this algorithm are population size (NP), maximum number of iterations (Max_FES), three different strategy candidate pools, and three different parameter candidate pools. At first, CoDE randomly generates NP individuals as a population, and evaluates their objective function. Then, for each individual, three trial vectors are generated by combining three strategies with three control parameters settings in a random way. The best trial is chosen for the next generation, and the process repeats until Max_FES is reached.

Input: *NP*: the number of individuals at each generation, i.e., the population size.
Max_FES: maximum number of function evaluations.
the strategy candidate pool: "rand/1/bin", "rand/2/bin", and "current-to-rand/1".
the parameter candidate pool: [F=1.0, C_r=0.1], [F=1.0, C_r=0.9], and [F=0.8, C_r=0.2].
(1) G=0;
(2) Generate an initial population $P_0 = \{\vec{x}_{1,0}, \ldots, \vec{x}_{NP,0}\}$ by uniformly and randomly sampling from the feasible solution
(3) Evaluate the objective function values $f(\vec{x}_{1,0}), \ldots, f(\vec{x}_{NP,0})$;
(4) $FES=NP$;
(5) **while** $FES<Max\_FES$ **do**
(6) $P_{G+1} = \emptyset$;
(7) **for** i=1:NP **do**
(8) Use the three trial vector generation strategies, each with a control parameter setting randomly selected from t
candidate pool, to generate three trial
vectors $\vec{u}_{i\_1,G}$, $\vec{u}_{i\_2,G}$, and $\vec{u}_{i\_3,G}$ for the target vector $\vec{x}_{i,G}$;
(9) Evaluate the objective function values of the three trial vectors $\vec{u}_{i\_1,G}$, $\vec{u}_{i\_2,G}$, and $\vec{u}_{i\_3,G}$;
(10) Choose the best trial vector (denoted as $\vec{u}^*_{i,G}$) from the three trial vectors $\vec{u}_{i\_1,G}$, $\vec{u}_{i\_2,G}$, and $\vec{u}_{i\_3,G}$;
(11) $P_{G+1} = P_{G+1} \cup select(\vec{x}_{i,G}, \vec{u}^*_{i,G})$;
(12) $FES=FES+3$;
(13) **end for**
(14) $G=G+1$;
(15) **end while**
Output: the individual with the smallest objective function value in the population.

Figure 2.9: Pseudocode of CoDE (After Wang et al., 2011)

2.4 Multimodal Optimization

Beside many researchers have adopted different methods to search for the global minimum, multiple local minimums exist and many solution methods can be trapped by the existence of a local minimum easily. Although it is often desirable to escape from local minima, sometimes the detection of all local minima becomes important. For example, there are cases that FS of local minima are close to global minimum while their corresponding locations are far from global minimum. In such cases, locations of local minima are important for stabilization purposes and optimization problems may not capture them (Duncan and Wright, 2005). Furthermore, there are some local minima (slip surface) that are so much deeper than global minima and if failure happens in their locations, damage is more vital than shallow slip surface which has lowest FS (Duncan and Wright, 2005). Optimization methods may not be intelligent enough to report such situations. For example, the shallow slip surface in the Figure 2.10 has the minimum factor of safety (1.15) and is global minimum; however, there is a deeper circle with higher FS (FS=1.21) as a local minimum.

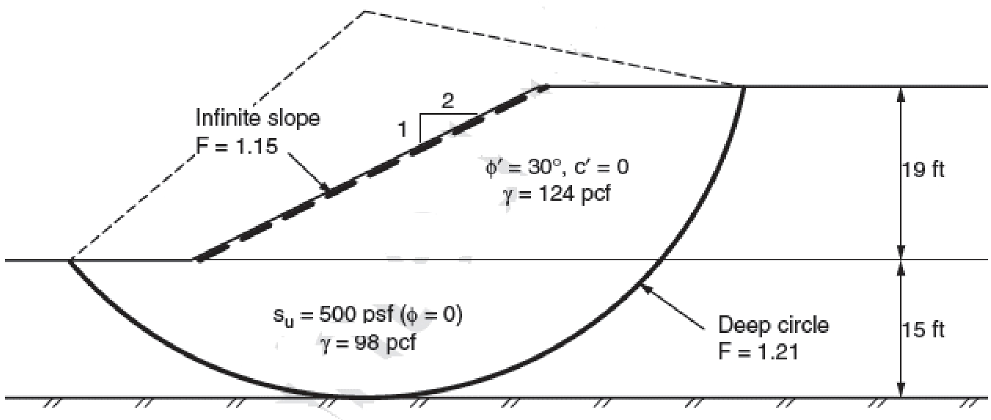

Figure 2.10: Slope with shallow global slip surface and deeper, locally critical circle (After Duncan and Wright, 2005)

In addition to the problem of local minima, in some slopes there are more than one slip surface with the same FS, which all are critical. Griffiths et al. (2012) presented an example slope, in which three failure modes with the same FS are found by FEM (Figure 2.11). They commented on inability of many LEM approaches to locate the multiple failure modes, as LEM require pre-assumption of failure mechanism.

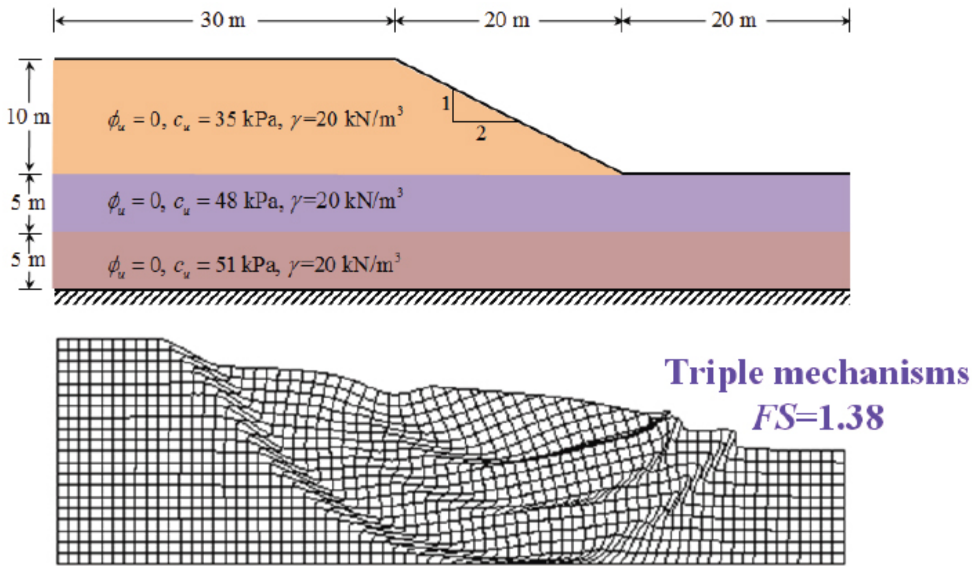

Figure 2.11: Multiple failure mechanisms of an undrained slope (after Griffiths et al., 2012)

In such cases, population-based approaches like evolutionary algorithms (EAs) can detect multi modes of failure in slopes within a single run (Das et al., 2011). Optimization techniques that find multiple local and global optima simultaneously are referred to as multimodal optimization methods. Standard evolutionary optimization algorithms can be promoted to incorporate niching techniques for locating multiple good solutions rather than only the best solution. A state of art review of such techniques is available in Das et al. (2011) with detailed explanations on how these techniques can be implemented form basics to more professional clustering techniques. Another comprehensive study is done by Singh and Deb (2006). They implemented seven multi-modal optimizations and compared the results for three problems.

2.6 Parallel Computing

Parallel computing is dividing a problem into discrete parts and application of multiple computer resources simultaneously (for example multiple CPUs) to solve each part of the problem. The aim of parallel processing in computer science is to reduce the computation time and increase the efficiency of complex algorithms toward a single computer resource. A schematic view of parallel computing is presented in Figure 2.12 for illustration.

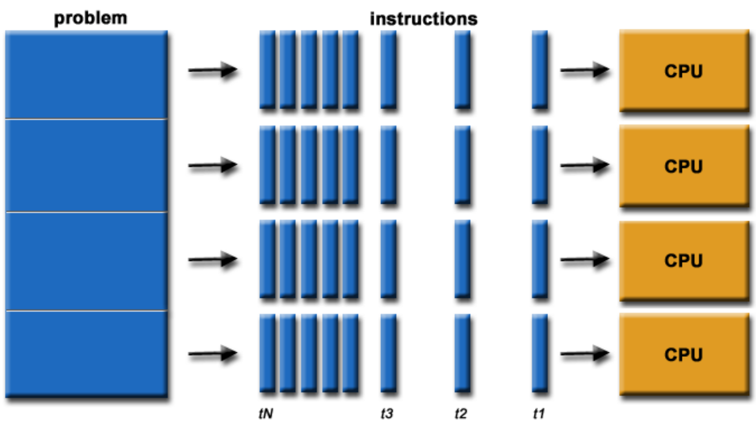

Figure 2.12: Parallel computing

Although by development of the computers, parallel processing is achievable more easily than past decades; some algorithms inherently cannot be parallelized or only some parts of the code can be modified for parallel computing. Therefore, it is necessary to identify independent set of instructions that can be run in parallel.

Using shared-memory parallel computers or multiprocessor computers in geotechnical engineering to speed up the programs is not very common. There are