

UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE GEOCIÊNCIAS

DESENVOLVIMENTO DE ROTINAS COMPUTACIONAIS PARA  
ANÁLISE CINEMÁTICA DE TALUDES ROCHOSOS

Mayko Aleci Silva de Oliveira

Orientador: Ginaldo Ademar da Cruz Campanha

Co-orientador: Luiz Antonio Urtiga e Silva

MONOGRAFIA DE TRABALHO DE FORMATURA

TF-14/24

São Paulo

2014

TF  
O48  
MAS.d

UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE GEOCIÊNCIAS

DEDALUS - Acervo - IGC



30900032412

**DESENVOLVIMENTO DE ROTINAS COMPUTACIONAIS PARA  
ANÁLISE CINEMÁTICA DE TALUDES ROCHOSOS**

Mayko Aleci Silva de Oliveira



Orientador: Ginaldo Ademar da Cruz Campanha

Co-orientador: Luiz Antonio Urtiga e Silva

MONOGRAFIA DE TRABALHO DE FORMATURA

TF-14/24

São Paulo

2014



TF  
048  
MAS d

UNITED STATES OF AMERICA  
DEPARTMENT OF COMMERCE

AMERICAN OVERSEAS INVESTMENT CORPORATION  
AMERICAN OVERSEAS INVESTMENT CORPORATION



AMERICAN OVERSEAS INVESTMENT CORPORATION  
AMERICAN OVERSEAS INVESTMENT CORPORATION

AMERICAN OVERSEAS INVESTMENT CORPORATION  
AMERICAN OVERSEAS INVESTMENT CORPORATION

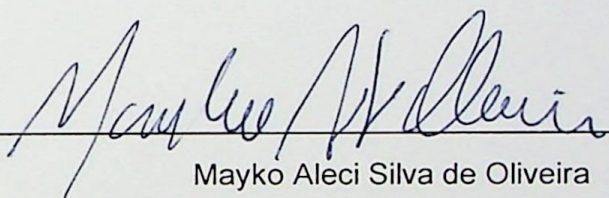
AMERICAN OVERSEAS INVESTMENT CORPORATION  
AMERICAN OVERSEAS INVESTMENT CORPORATION

UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE GEOCIÊNCIAS

**DESENVOLVIMENTO DE ROTINAS COMPUTACIONAIS PARA  
ANÁLISE CINEMÁTICA DE TALUDES ROCHOSOS**

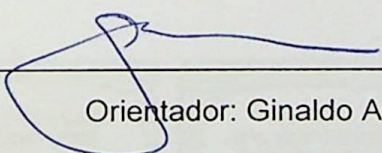
MONOGRAFIA DE TRABALHO DE FORMATURA

TF-14/24



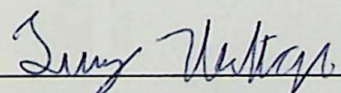
---

Mayko Aleci Silva de Oliveira



---

Orientador: Ginaldo Ademar da Cruz Campanha



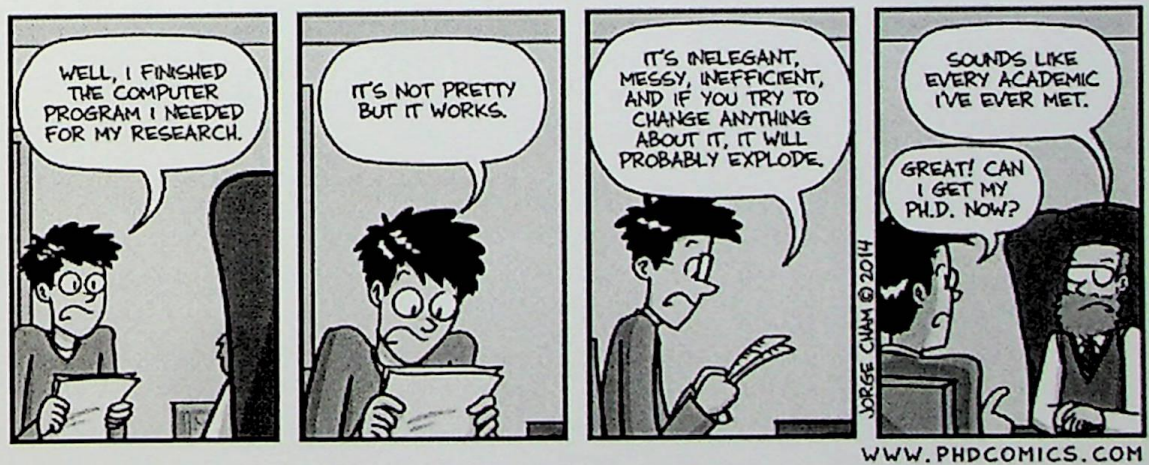
---

Co-orientador: Luiz Antônio Urtiga e Silva

São Paulo

2014





Cham, J. (2014) -



## AGREDECIMENTOS

A todos que, diretamente ou indiretamente, participaram de todo o longo processo de minha formação. A Universidade de São Paulo e ao Instituto de Geociências e toda equipe de professores e funcionários. A todos os colegas da turma 52, minha original, e também aos colegas das outras turmas a qual fiz parte das matérias.

A grandes geólogos que encontrei nas empresas em que realizei estágios, em especial ao Leonardo Lamy, Hamilton, Edmilton, Ivan, Silvia e Vinicius.

A “Aroeira” pelo banco, que proporcionou grandes momentos durante a graduação e a todos os seus membros (Zé, Cowboy, Quevara, Burns, Zigoto, Consolo, Angélica, Bolovo). Aos meus estimados amigos Toba, Raimunda, Planaria que têm grande participação nesse projeto. Pela revisão criteriosa desse texto, pelos conselhos e principalmente pelas críticas.

Ao Urtiga, pela ideia, paciência e por ser um ótimo co-orientador. Ao professor Ginaldo por me orientar, suportar e sempre estar disposto a conversas e resoluções de dúvidas.

A minha família como um todo. Minhas irmãs Fernanda, Kari e Patrícia, meus cunhados Wagner e Alex. Os “primos tortos”, Patrícia, Cleiton, Dani e Sandro. A minha mãe que amo tanto.

E por fim, ao meu pai. Sem você nada disso seria possível.



## RESUMO

A análise de estabilidade de taludes rochosos é um item primordial nos projetos de engenharia civil e de minas. Geralmente, em uma primeira análise, é feito o estudo cinemático, obtendo-se uma avaliação prévia das características de um talude. Esse estudo teve como objetivo o desenvolvimento de rotinas computacionais para avaliação de estabilidade de taludes rochosos pelo critério cinemático. Baseado na revisão bibliográfica foram desenvolvidos quatro rotinas em linguagem de programação Python, referentes aos principais tipos ruptura: planar, em cunha, tombamento direto e flexural. A partir dos dados inseridos pelo usuário, essas rotinas identificam as descontinuidades instáveis e estáveis para o talude analisado, agrupando-as em diferentes tabelas. Esses dados podem ser inseridos em um programa de análises estereográficas, permitindo uma avaliação visual dos dados. Os testes realizados mostram que as rotinas são compatíveis com as técnicas manuais de plotagem, sendo uma ferramenta potencial para a análise de estabilidade de taludes rochosos.

Palavras chave: Estabilidade de taludes rochosos, programação, .



## ABSTRACT

The rock slope analysis is a essential item in civil and mining engineering. Usualy, in a inicial analysis, the kinematic study obtain a preliminar evaluation of the characteristics of a slope. This study aimed the development of computacional routines for rock slope evaluation, by kinematic criteria. Based on the reviwed literature, four routines in Python programming language were developed for the principal styles of rock failure: planar, wedge, direct and flexural toppling. From data inserted by the user, this routines identify the discontinuities stable and unstable for the analysed slope, grouping them separate tables. This data can be viewed in a stereographic software, allowing a visual data analysis. The test shows that developed routines are compatible with manual plotting techniques, being a potencial tool in rock slope analysis.

Key words: Rock slope stability, programming.



## SUMÁRIO

|                                                        |    |
|--------------------------------------------------------|----|
| 1. INTRODUÇÃO .....                                    | 4  |
| 2. OBJETIVO.....                                       | 4  |
| 3. FUNDAMENTAÇÃO TEÓRICA .....                         | 5  |
| 3.1. Taludes e processos de movimentação de massa..... | 5  |
| 3.2. Estabilidade de Taludes Rochosos .....            | 6  |
| 3.3. Representação Espacial de Descontinuidades .....  | 6  |
| 3.4. Ângulo de Atrito .....                            | 8  |
| 3.5. Rupturas em Taludes Rochosos .....                | 8  |
| 3.5.1. <i>Rupturas Planares</i> .....                  | 9  |
| 3.5.2. <i>Rupturas em Cunha</i> .....                  | 12 |
| 3.5.3. <i>Tombamento</i> .....                         | 13 |
| 4. MATERIAIS E MÉTODOS .....                           | 20 |
| 5. RESULTADOS OBTIDOS .....                            | 20 |
| 5.1. Rupturas planares.....                            | 23 |
| 5.2. Rupturas em cunha.....                            | 23 |
| 5.3. Tombamento flexural .....                         | 24 |
| 5.4. Tombamento direto .....                           | 24 |
| 5.5. Testes Realizados.....                            | 25 |
| 6. CONCLUSÕES .....                                    | 28 |
| 7. REFERÊNCIAS BIBLIOGRÁFICAS.....                     | 28 |

# SUMÁRIO

|       |                                                          |    |
|-------|----------------------------------------------------------|----|
| 1     | INTRODUÇÃO                                               | 4  |
| 2     | OBJETIVO                                                 | 4  |
| 3     | FUNDAMENTAÇÃO TEÓRICA                                    | 5  |
| 3.1   | Técnicas e processos de desenvolvimento de projetos      | 5  |
| 3.2   | Conceitos de gerenciamento de projetos                   | 6  |
| 3.3   | Importância do conhecimento de Gerenciamento de Projetos | 6  |
| 3.4   | Áreas do conhecimento                                    | 6  |
| 3.5   | Importância do Gerenciamento de Projetos                 | 6  |
| 3.5.1 | Gerenciamento de Projetos                                | 6  |
| 3.5.2 | Gerenciamento de Recursos Humanos                        | 7  |
| 3.5.3 | Gerenciamento de Materiais                               | 7  |
| 4     | MATERIAIS E MÉTODOS                                      | 10 |
| 5     | RESULTADOS E DISCUSSÃO                                   | 10 |
| 5.1   | Resumo dos dados                                         | 11 |
| 5.2   | Resumo dos dados                                         | 11 |
| 5.3   | Resumo dos dados                                         | 11 |
| 5.4   | Resumo dos dados                                         | 11 |
| 5.5   | Resumo dos dados                                         | 11 |
| 6     | CONCLUSÃO                                                | 12 |
| 7     | REFERÊNCIAS BIBLIOGRÁFICAS                               | 13 |



## ANEXOS:

ANEXO I - EXEMPLOS PARA RUPTURAS EM CUNHA

ANEXO II - EXEMPLOS PARA RUPTURAS PLANARES

ANEXO III - EXEMPLOS PARA TOMBAMENTO DIRETO

ANEXO IV - EXEMPLOS PARA TOMBAMENTO FLEXURAL

ANEXO V - CÓDIGO FONTE DAS ROTINAS – RUPTURAS EM CUNHA

ANEXO V - CÓDIGO FONTE DAS ROTINAS – RUPTURAS PLANARES

ANEXO V - CÓDIGO FONTE DAS ROTINAS – TOMBAMENTO DIRETO

ANEXO V - CÓDIGO FONTE DAS ROTINAS – TOMBAMENTO FLEXURAL



## 1. INTRODUÇÃO

A análise de estabilidade de taludes naturais e de escavação é um item primordial nos projetos de engenharia civil e de minas e também tem extrema relevância em estudo de áreas de risco. Sua importância se dá principalmente do ponto de vista dos aspectos sociais, quando falamos de riscos de perdas humanas e econômicas, assim como quando do dimensionamento de obras e projetos de avaliação e operação de jazidas. Portanto, deve-se conhecer as características das movimentações que podem ocorrer em um talude, e quais são os fatores geradores ou limitantes que definem a estabilidade ou instabilidade de um corte.

No caso de taludes em rocha, em uma primeira análise, geralmente é feito seu estudo cinemático, no qual são estudadas as relações geométricas entre as descontinuidades individuais ou famílias e seu ângulo de atrito, com a face livre da escavação, sem fazer referência às forças responsáveis pela movimentação (Hudson & Harrison, 1997; Goodman, 1989). Dessa maneira se obtém de forma rápida uma avaliação prévia das características de um talude.

É muito comum que estas análises sejam feitas manualmente, plotando-se em estereogramas os dados, talude a talude. Quando se tem grandes quantidades de informações, este procedimento pode se tornar bastante dispendioso e arriscado, pois sempre há a possibilidade de erros de plotagem. Torna-se então importante a elaboração de métodos para a análise sistemática desses dados a fim de agilizar os estudos e diminuir as chances de erros.

Neste contexto, o presente trabalho visou a automação do procedimento de análise cinemática de taludes rochosos para os quatro principais tipos de ruptura associados, a saber: ruptura planar, em cunha e tombamento direto e flexural. Para isso, foi selecionada a linguagem de programação Python, orientada a objetos e que possui uma ampla biblioteca de funções. Com isto pretende-se auxiliar os profissionais da área a reduzir o tempo de análise dos dados, bem como o aumento da confiabilidade das informações.

## 2. OBJETIVO

O objetivo desse trabalho foi o desenvolvimento de rotinas computacionais para a análise de estabilidade de taludes rochosos. Elaborou-se quatro módulos para os principais tipos de instabilidade:

- a) Instabilidade em cunha
- b) Instabilidade planar



- c) Tombamento direto
- d) Tombamento flexura

Buscou-se elaborar um *software* independente, gratuito e de fácil utilização, que poderá ser usado pela comunidade científica e técnica afim de auxiliar nos estudos iniciais em estabilidade de talude.

### 3. FUNDAMENTAÇÃO TEÓRICA

#### 3.1. Taludes e processos de movimentação de massa

Os estudos de estabilidade de taludes e movimentos de massa têm sido foco de estudo das mais diversas disciplinas (Engenharia Civil e de Minas, Geografia, Geologia, Geotecnia, Geomorfologia, Mecânica dos Solos e das Rochas e outros) devido a sua relevância em questões teóricas, praticas, sociais e também a importância do ponto de vista econômico.

Talude pode ser definido como qualquer superfície inclinada em um terreno composto por massas de rocha, solo ou mistos. Podem ser naturais ou ainda apresentarem alguma modificação antrópica. Nesse caso inserem-se os taludes de corte e/ou aterro.

Uma definição simples, porém muito usada acerca de movimentos de massa é apresentada por Cruden (1991) e por Cruden & Varnes (1996), onde “escorregamento é a movimentação de uma massa de solo, detritos ou rocha encosta abaixo”.

São encontrados na literatura diversos sistemas classificadores dos movimentos de massa, genericamente chamados de “escorregamentos”. As classificações mais comuns para movimentos de massa são baseadas nas propriedades dos materiais e nos tipos de processos (Thiebes, 2012). Cruden & Varnes (1996) definem seis tipos de processos (Tombamento, queda, escorregamento translacional e rotacional, fluxo, espalhamento e processos complexos) para três tipos de materiais (Solo, detritos e rocha). Uma outra forma de classificação é baseada na velocidade dos movimentos podendo ser desde extremamente rápidos (da ordem de metros por segundo) até extremamente lentos (da ordem de milímetros por ano) (Cruden & Varnes, 1996). Movimentos de massa ainda podem ser classificados quanto ao seu estado físico. Cruden & Varnes (1996) definem oito grupos, sendo eles: ativos, suspensos, reativados, inativos, dormentes, abandonados, estabilizados e reliquias.

Outra classificação disponível na literatura é a realizada por Guidicini & Nieble (1984), extraído do trabalho de Freire (1965), onde os movimentos de massa são classificados em três tipos: escorregamentos, escoamentos e subsidência.



### 3.2. Estabilidade de Taludes Rochosos

A estabilidade de taludes rochosos está diretamente correlacionada a estruturas geológicas dentro do maciço rochoso. Essas estruturas podem ser de vários tipos, tais como: acamamento sedimentar, xistosidade, juntas e falhas, sendo essas estruturas chamadas genericamente de descontinuidades.

Descontinuidade, segundo Hudson & Harrison (1997), denota qualquer separação em um maciço rochoso, que apresenta baixa ou nula resistência à tração. O termo não faz distinção a respeito da idade, geometria ou conotação genética (sedimentar, diagenética, tectônica), porém geralmente são esses os principais fatores que controlam suas características (Priest, 1993).

O termo descontinuidade, em engenharia civil e geotécnica, é usado genericamente para se referir a estruturas planares. Mesmo sendo essas as mais comuns, ocorrem outros tipos, como por exemplo lineares e onduladas

. Para esse trabalho será dado enfoque nas descontinuidades do tipo planar.

A orientação das descontinuidades é um fator de grande importância nas características geomecânicas de um maciço rochoso (Jaeger & Cook, 1969), em especial em taludes, onde é um dos principais fatores que influenciam a sua estabilidade. Em Geologia, as orientações de dados tridimensionais geralmente são representadas por meio das projeções estereográficas (Fossen, 2010). Essa forma de apresentação dos dados facilita a visualização das estruturas e ainda permite a realização de muitas operações entre planos, linhas, pontos ou conjuntos desses fatores. Mais detalhes sobre projeções estereográficas podem ser facilmente localizados em diversos livros de geologia estrutural como Fossen (2010), Ragan (2009), ou ainda em alguns livros de mecânica de rochas ou engenharia, em especial Priest (1993).

### 3.3. Representação Espacial de Descontinuidades

O sistema de representação de dados estruturais mais comum em geologia estrutural é a representação de uma estrutura baseada no azimute do rumo do mergulho e o mergulho da estrutura citada (p. ex.: 125/45 indica que a estrutura mergulha 45 graus para o rumo 125 graus em relação ao norte geográfico). Esse sistema é comumente chamado de "notação *Clar*". Esse nome refere-se a um modelo de bússola em que a leitura é feita como azimute do mergulho e mergulho.

A Figura 1 mostra a representação de um vetor, que pode representar um dado estrutural (como intersecções de planos, ou a linha de máximo mergulho de um plano) no sistema cartesiano (x,y,z) sendo ilustrado em função do "*trend* (T)" e do "*plunge* (P)".



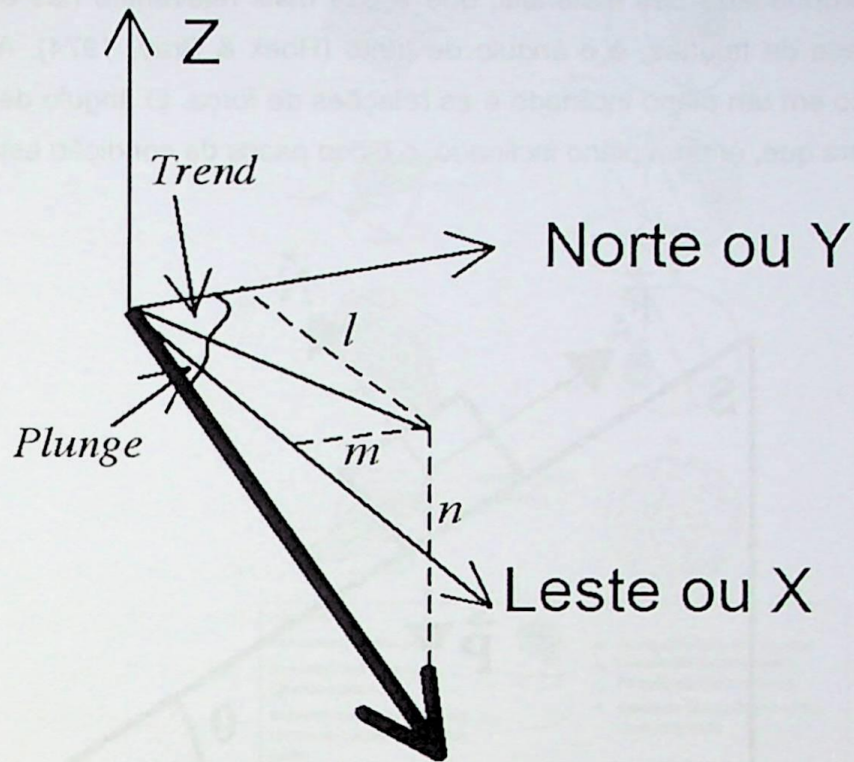


Figura 1 - Representação vetorial de uma medida estrutural (modificado de University of Alberta 2011)

Ainda na mesma figura,  $l$ ,  $m$  e  $n$  representam a magnitude do vetor nas direções X, Y e Z, que representam as direções leste, oeste e para cima, respectivamente. A magnitude (ou comprimento)  $|u|$  do vetor pode ser calculado pela fórmula:

$$|u| = \sqrt{l^2 + m^2 + n^2} \quad (1)$$

Se  $|u| = 1.0$ , trata-se de um vetor unitário. Considerando o vetor da Figura 1 como sendo um vetor unitário, então  $l$ ,  $m$  e  $n$  são seus cossenos diretores (Priest, 1993) para as direções X, Y e Z e o seu cálculo se dá pelas fórmulas abaixo:

$$l = \sin T \cos P \quad (2)$$

$$m = \cos T \cos P \quad (3)$$

$$n = -\sin P \quad (4)$$

Para análise de dados planares (como descontinuidades), é necessário determinar a atitude da reta normal ao plano e então calcular seus cossenos diretores. Sendo  $\alpha_n$  o rumo e  $\beta_n$  o mergulho do plano, são utilizadas as seguintes equações:

$$\alpha_n = T \pm 180^\circ \Rightarrow 0 \leq \alpha_n \leq 360^\circ \quad (5)$$

$$\beta_n = 90^\circ - P \quad (6)$$



### 3.4. Ângulo de Atrito

Uma propriedade dos materiais, que é das mais relevantes nas discussões sobre estabilidade de taludes, é o ângulo de atrito (Hoek & Bray, 1974). A Figura 2 mostra um bloco em um plano inclinado e as relações de força. O ângulo de atrito é o ângulo limite para que, em um plano inclinado, o bloco passe da condição estática para movimentação.

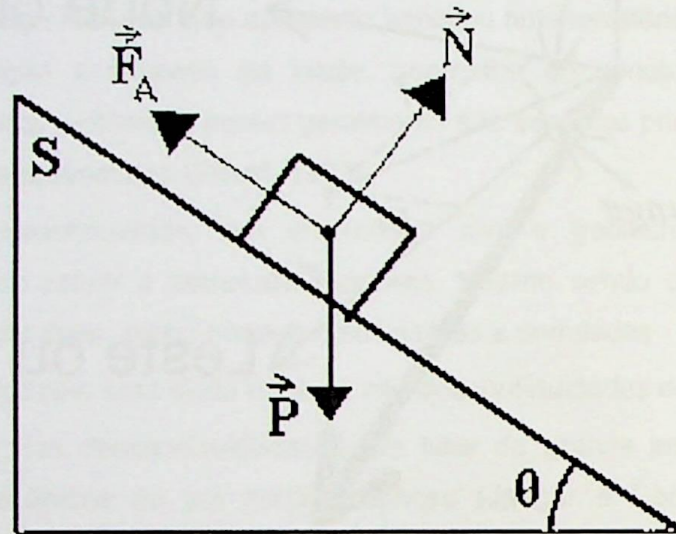


Figura 2 – Relação de forças de um bloco em um plano inclinado.

No caso tridimensional, é realizada a análise do chamado “cone de atrito”, que é o lugar geométrico das retas com valor igual ao ângulo de atrito que, em um estereograma, representa um círculo menor. Valores de ângulos de atrito podem ser medidos em laboratório, ou ainda podem ser usados valores médios disponíveis na literatura como mais comuns.

### 3.5. Rupturas em Taludes Rochosos

Diferentes tipos de rupturas são diretamente associados com as relações entre as orientações das descontinuidades e da face escavada (Hoek & Brown, 1980 ; Hoek & Bray, 1981 ; Wyllie & Mah, 2004).

Na Figura 3 são apresentados os quatro tipos de ruptura mais comuns em taludes rochosos e sua representação estereográfica. Em a) rupturas do tipo planar, em b) rupturas em cunha, em c) tombamento (ou *Toppling*) e em d) rupturas circulares. Esse último tipo não faz parte do escopo desse trabalho por sua ocorrência ser mais comum em solos ou rochas muito alteradas ou fraturadas.



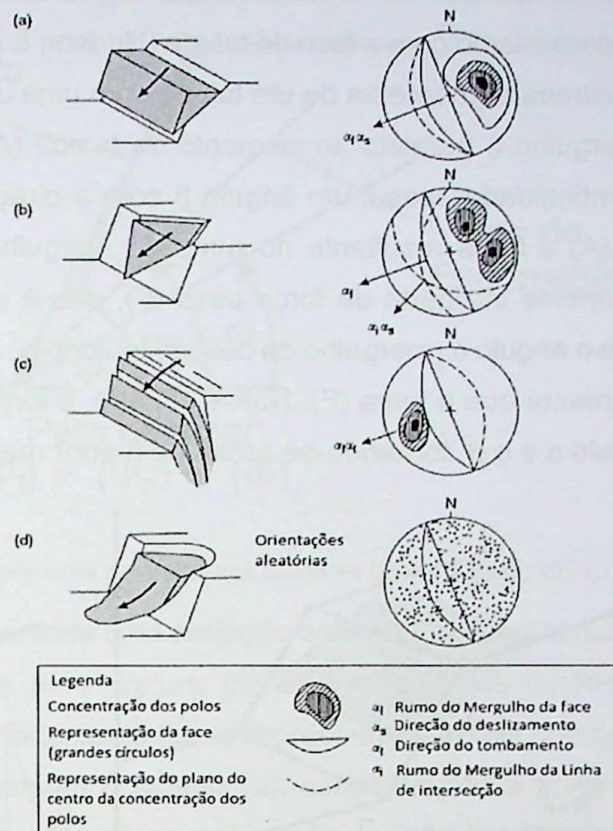


Figura 3 - Tipos de rupturas em taludes e condições estruturais para que ocorram (Wyllie & Mah, 2004).

A cinemática refere-se ao estudo do movimento, sem fazer referência às forças que o produzem (Hudson & Harrison, 1997; Goodman, 1989). Um estudo inicial de estabilidade de taludes rochosos pelo método cinemático é relevante, pois o movimento é possível para determinadas geometrias decorrentes das orientações das discontinuidades e da face do talude.

Apesar desse método não prover de valores numéricos de instabilidade, é extremamente importante em uma primeira análise, devido à facilidade e rapidez de sua interpretação.

A análise cinemática de rupturas planares, em cunha e de tombamento, serão detalhadas na sequência. O método é exposto por Hoek & Bray (1974), Goodman (1989), Hudson & Harrison (1997) e Wyllie & Mah (2004).

### 3.5.1. Rupturas Planares

Para que ocorra a instabilidade de uma ruptura planar são necessários os seguintes critérios:

- O rumo do mergulho do talude deverá ter aproximadamente  $\pm 20^\circ$  de diferença do rumo do mergulho da discontinuidade. Esse é um critério empírico onde foi observado que deslizamentos tendem a ocorrer com mais



frequência quando as descontinuidades que os controlam apresentam um certo paralelismo com a face do talude (Hudson & Harrison, 1997). Na Figura 4 é ilustrada a geometria de um talude com uma descontinuidade cujo rumo do mergulho é paralelo ao caimento do talude (A) e outro cujo direção da descontinuidade possui um ângulo  $\beta$  com a direção da face do talude. No caso (A) a força resultante no rumo do mergulho do talude (R) é igual a componente inclinada da força peso (F), que é uma relação entre a força peso e o ângulo  $\alpha$  (mergulho da descontinuidade). Já no caso (B) a resultante (R) é menor que a força (F). Nesse cenário, a força (R) é uma relação entre o ângulo  $\alpha$  e o  $\beta$ , tornando-se assim de menor magnitude.

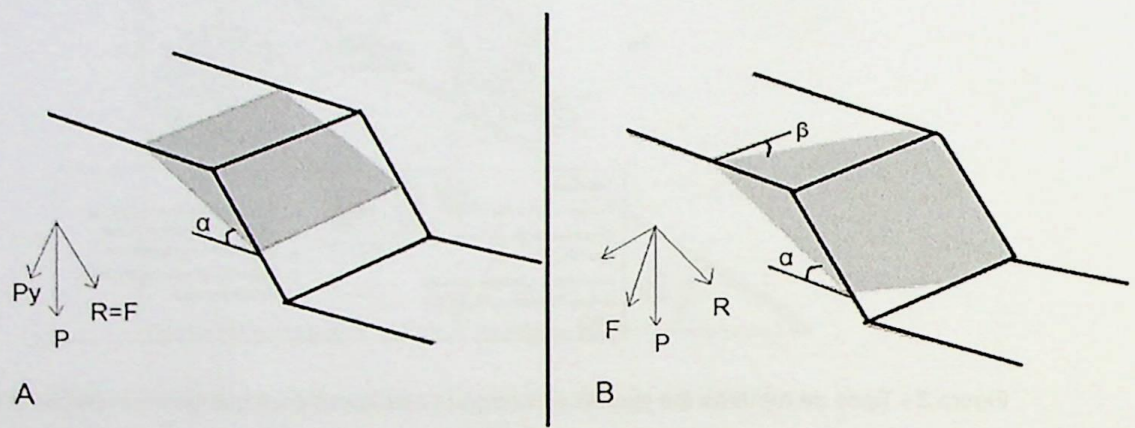


Figura 4 - Critério do  $\pm 20^\circ$  de paralelismo

- b) O mergulho do plano de descontinuidade ( $\Psi_p$ ) deverá exceder o ângulo de atrito ( $\Phi$ ).
- c) A inclinação do talude ( $\Psi_F$ ) deverá ser maior que o mergulho do plano de descontinuidade ( $\Psi_p$ ) para que blocos de rocha instável possam deslizar.
- d) O plano de descontinuidade deve aflorar ("daylight") no plano do talude, isso é necessário para que os blocos formados a partir do critério (c) possam ter liberdade de movimento

Os parâmetros acima são ilustrados na Figura 5.



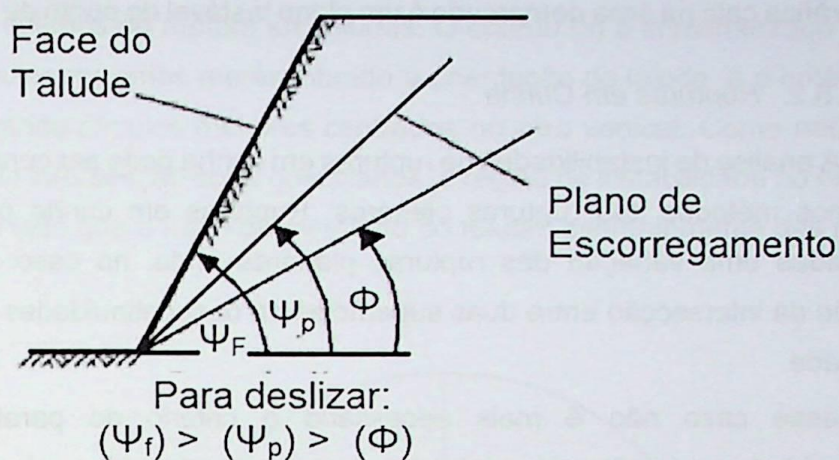


Figura 5 - Geometria para rupturas planares (Hoek & Bray, 1974).

Na Figura 6 é apresentada uma projeção estereográfica genérica de uma análise de estabilidade de taludes para ruptura planar baseada nos critérios apresentados acima. Cada linha apresentada nesta figura representa uma das condições listadas. As duas linhas radiais representam o critério (a), concentrando a zona de instabilidade sendo  $\pm 20^\circ$  da direção do talude. Já a combinação da linha concêntrica marcada como "Atrito" e a linha marcada como "Talude" representam os critérios (b) e (c). Foi usado um ângulo de atrito de  $30^\circ$  e um talude com mergulho de  $75^\circ$ .

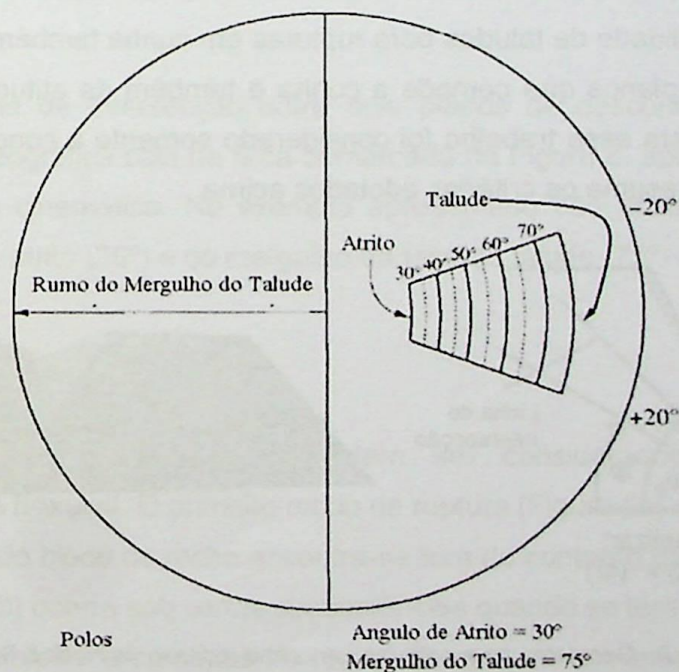


Figura 6 - Análise de rupturas planares (Hudson & Harrison, 1997).



Qualquer plano de descontinuidade cujo polo de sua representação estereográfica caia na área demarcada é um plano instável do ponto de vista cinemático.

### 3.5.2. Rupturas em Cunha

A análise de instabilidades de rupturas em cunha pode ser construída utilizando os mesmos métodos das rupturas planares. Rupturas em cunha podem ainda ser consideradas uma variação das rupturas planares onde, no caso mais simples, a orientação da intersecção entre duas superfícies de descontinuidades que determina a instabilidade.

Nesse caso não é mais necessário o critério do paralelismo entre a descontinuidade e o talude, pois rupturas em cunha são definidas unicamente pela linha de intersecção entre os planos de fraqueza. Os critérios para instabilidade, portanto, são:

- O mergulho do talude deve ser maior que o caimento (*plunge*) da linha de intersecção entre dois planos de descontinuidades potenciais, para que possam ser formadas cunhas instáveis. Esse critério é similar ao apresentado no item (c) das rupturas planares.
- A linha de intersecção entre os dois planos de descontinuidades deve aflorar na face do talude, para que os blocos tenham liberdade de movimento.
- O caimento da linha de intersecção entre os dois planos potenciais deverá exceder o ângulo de atrito da rocha em estudo.

A estabilidade de taludes para rupturas em cunha também é influenciada pelo ângulo entre os planos que compõe a cunha e também da atitude da bissetriz entre esses planos. Para esse trabalho foi considerado somente a condição simplificada. A Figura 7 abaixo resume os critérios adotados acima.

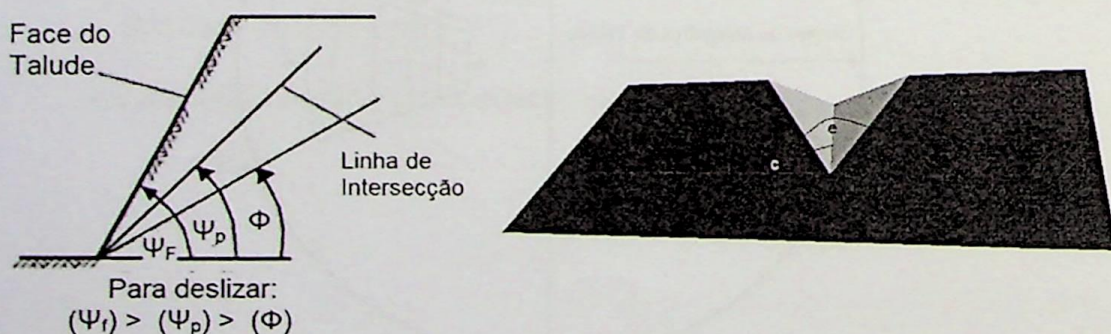


Figura 7 - Geometria para rupturas em cunha (editado de Hoek & Bray, 1974).



A Figura 8 é análoga à apresentada no item anterior sendo a representação genérica para os critérios de ruptura em taludes. O critério (a) é implementado usando uma serie de círculos máximos representando a orientação do talude, e o critério (c) é implementado usando círculos menores centrados no eixo vertical. Como neste caso utiliza-se a linha de intersecção entre dois planos, a região de instabilidade no diagrama ocorre do mesmo lado que o rumo do mergulho do talude, diferentemente das rupturas planares.

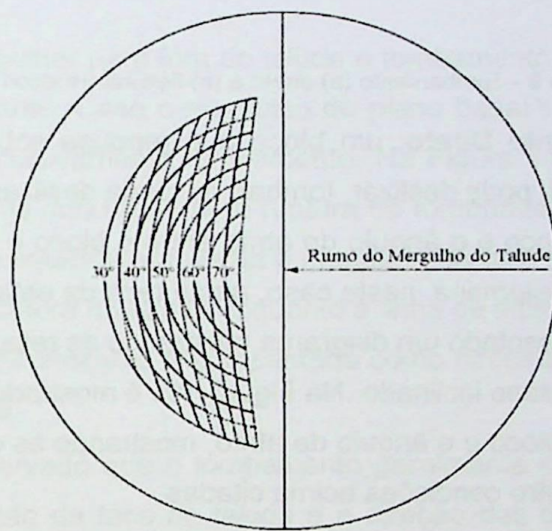


Figura 8 - Análise de rupturas em cunha (Hudson & Harrison, 1997).

Qualquer linha de intersecção entre dois planos de descontinuidades, cuja representação estereográfica caia na área demarcada na Figura 8, aponta uma cunha instável pelo critério cinemático. No exemplo apresentado são usados os mesmos valores de ângulo de atrito ( $30^\circ$ ) e do mergulho da face do talude ( $75^\circ$ ) que no exemplo de rupturas planares.

### 3.5.3. Tombamento

Para rupturas por tombamento podem ser considerados dois modos: tombamento **direto** e **flexural**. O primeiro modo de ruptura (Figura 9a) ocorre quando o centro de gravidade do bloco de rocha encontra-se fora do contorno da base do bloco. O segundo (Figura 9b) ocorre sob certas circunstâncias quando se tem uma rocha com descontinuidades paralelas e camadas de rocha afloram em um talude rochoso, e a tensão cisalhante nessas descontinuidades induz o seu deslizamento, causando assim tombamento dos blocos.



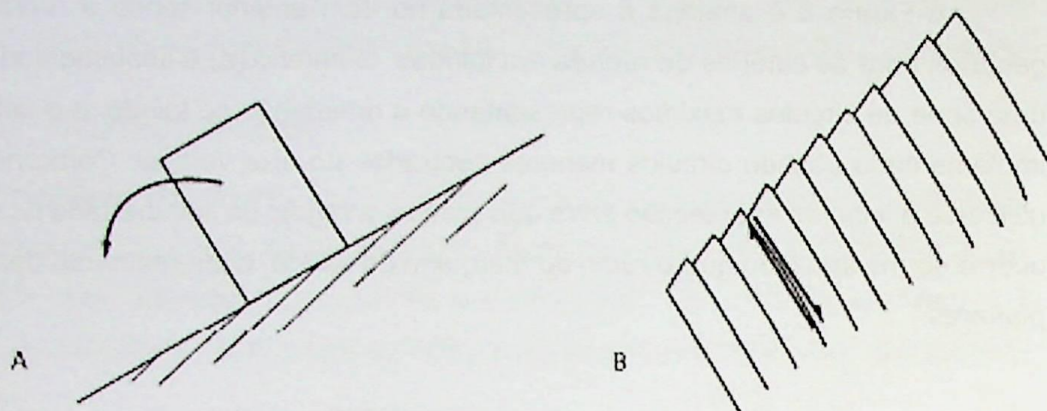


Figura 9 - Tombamento (a) direto e (b) flexural (Hudson & Harrison, 1997).

**Tombamento Direto:** um bloco que repousa sobre uma superfície inclinada pode estar estável, pode deslizar, tombar, ou ainda deslizar e tombar simultaneamente. A geometria do bloco e o ângulo de atrito entre o bloco e a superfície na qual o bloco repousa é o que determina, neste caso, a natureza da estabilidade ou instabilidade. Na Figura 10a é apresentado um diagrama mostrando as relações geométricas e força em um bloco em um plano inclinado. Na Figura 10b é mostrado um gráfico que considera a geometria desse bloco e o ângulo de atrito, mostrando as condições para que ocorram cada uma das quatro condições acima citadas.

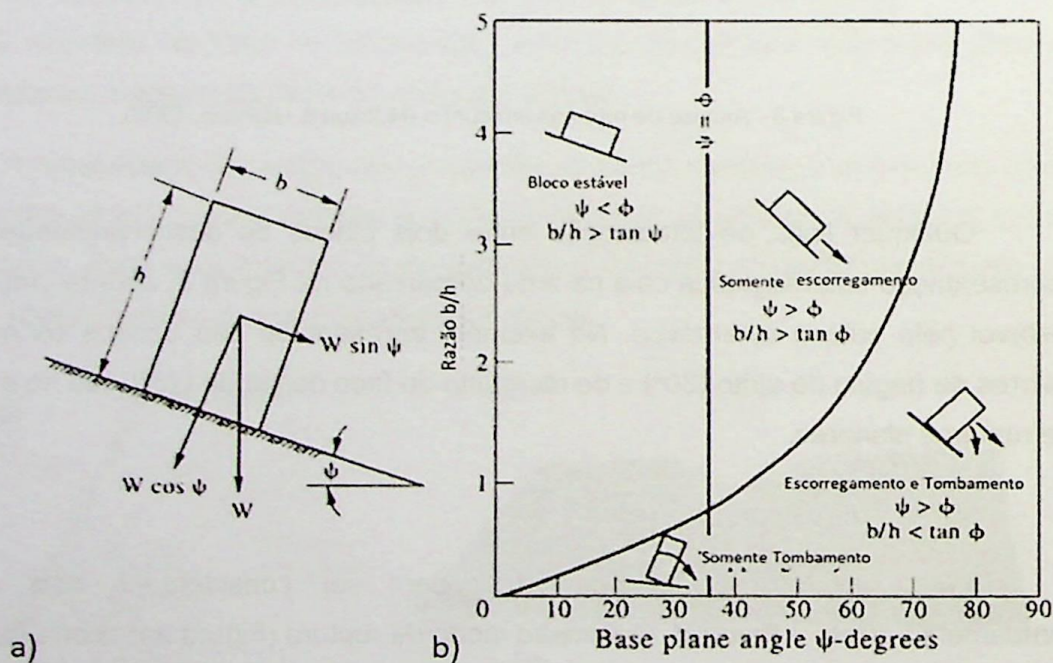


Figura 10 - Instabilidade de Tombamento e escorregamento para um bloco em plano inclinado (Hoek & Bray, 1974).



No caso de tombamento direto, o critério cinemático é relacionado somente à geometria do bloco rochoso, porém o ângulo de atrito pode ser usado para definir o limite entre o tombamento, o escorregamento e o tombamento com escorregamento, como mostrado na Figura 10. Os critérios são:

- a) Devem existir dois conjuntos de planos de descontinuidades cujas intersecções mergulhem para dentro do talude, para serem formados blocos de rocha.
- b) Deve existir uma família de descontinuidade para formar a base para o tombamento dos blocos.

Se o plano basal mergulhar para fora do talude o tombamento é mais provável, mas tal condição não é limitante. Caso o mergulho do plano basal seja menor que o ângulo de atrito ocorrerá exclusivamente tombamento. Na Figura 11 é mostrado um estereograma genérico com as condições para ruptura de tombamento direto. A linha com a seta apontada para a esquerda apresenta o mergulho da face do talude. Devido ao plano basal mergulhar para fora do talude enquanto a linha de intersecção mergulha para dentro do talude, a região limitante é apresentada como círculos concêntricos do mesmo lado do estereograma.

Empiricamente é observado que o tombamento geralmente ocorre quando há um paralelismo entre a direção da face do talude e a direção das descontinuidades. Hoek & Bray (1974) sugeriram a utilização do limite  $\pm 15^\circ$ . Goodman (1989), revisando os critérios adotados por Hoek & Bray (1974), sugere o uso de  $\pm 30^\circ$ . Hudson & Harrison (1997) e Wyllie & Mah (2004) usam como parâmetro de paralelismo  $\pm 20^\circ$ . Em caso de taludes muito íngremes esse parâmetro pode ser aumentado (Hudson & Harrison, 1997). Tal parâmetro é mostrado no estereograma da Figura 11 como duas linhas radiais.



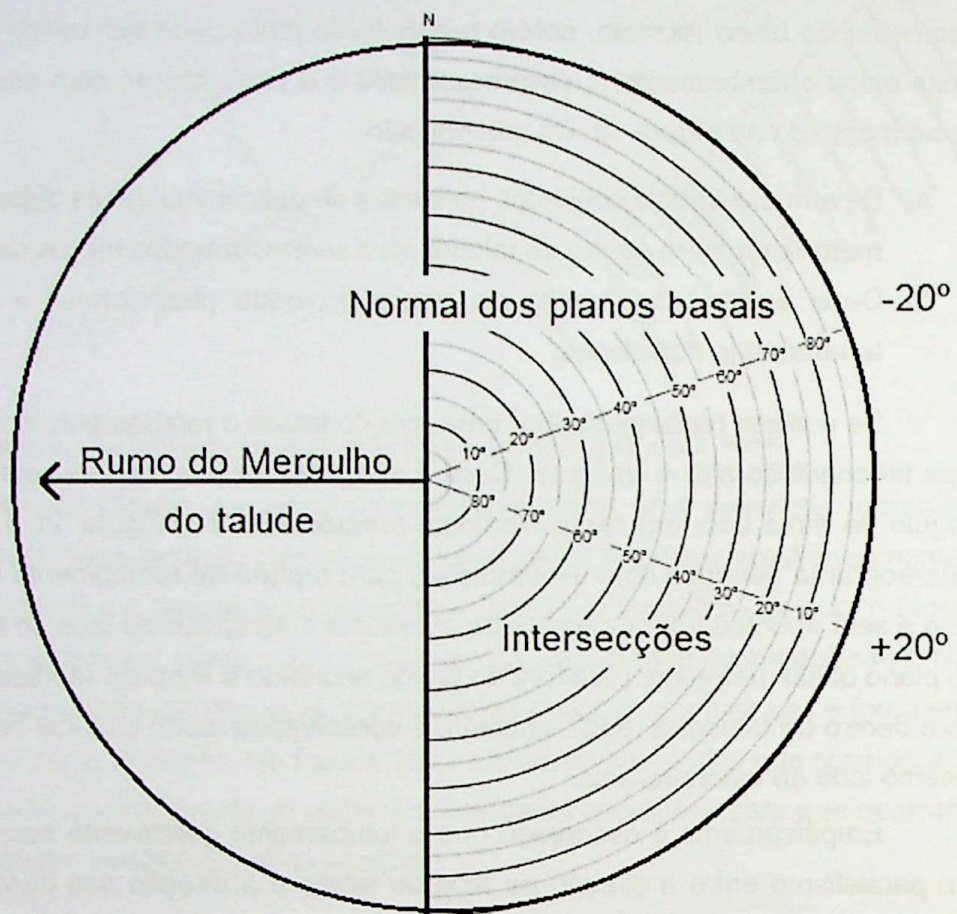


Figura 11 - Análise de rupturas de tombamento direto (Hudson & Harrison, 1997).

Na Figura 12 são mostradas as diversas regiões de instabilidade associadas ao tombamento direto. Um aspecto observado é que o rumo do mergulho do talude é usado como critério de paralelismo para os polos e intersecções, e não sua inclinação.



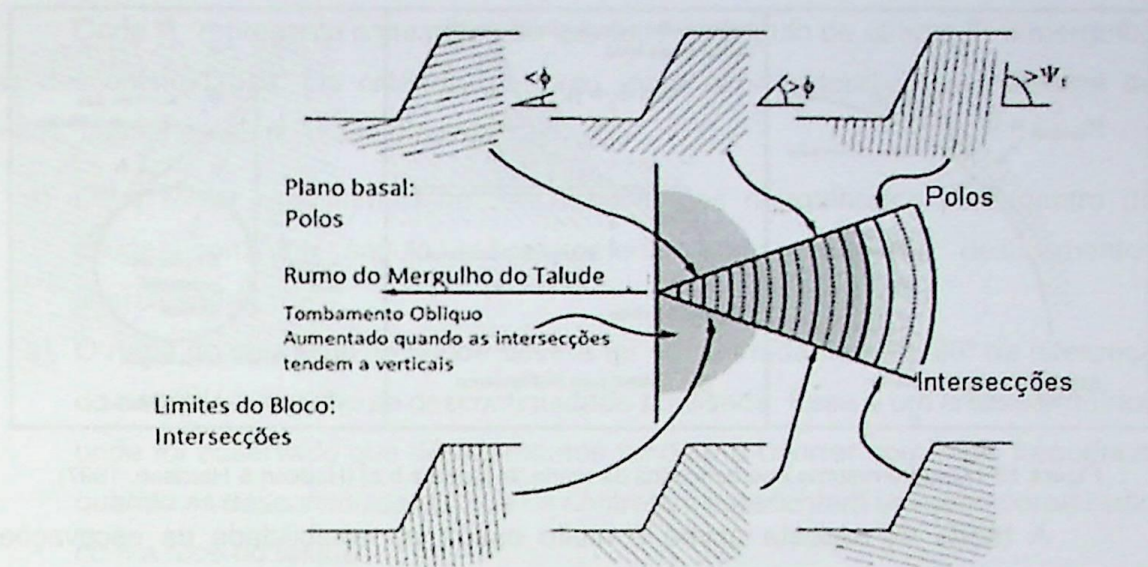


Figura 12 - Zoom do estereograma mostrando as relações entre o plano basal e as intersecções nas análises de tombamento direto (Hudson & Harrison, 1997)

**Tombamento Flexural:** Quando camadas de rochas deslizam umas sobre as outras, as forças inter-camadas ficam inclinadas  $\Phi_j$  graus da normal de cada camada (Goodman, 1989). O parâmetro  $\Phi_j$  é o ângulo de atrito entre dois planos de descontinuidades em uma rocha (Hudson & Harrison, 1997).

Os conceitos básicos para o deslizamento intercamadas resumida por Hudson & Harrison (1997) são mostrados abaixo:

- Quando uma rocha é escavada, as tensões principais tornam-se perpendiculares e/ou paralelas à superfície escavada livre; a superfície de escavação torna-se o principal plano de tensão, ou seja, um plano com tensão cisalhante zero (Figura 13a);
- O deslizamento sobre uma descontinuidade oblíqua à face do talude pode ocorrer quando o ângulo entre a tensão principal aplicada e a normal da descontinuidade for maior que o ângulo  $\Phi_j$  (Figura 13b);
- A construção de uma geometria favorável utilizando os critérios (a) e (b) pode ser usada para estender os limites para potenciais deslizamentos intercamadas (ou interdescontinuidades) (Figura 13c).



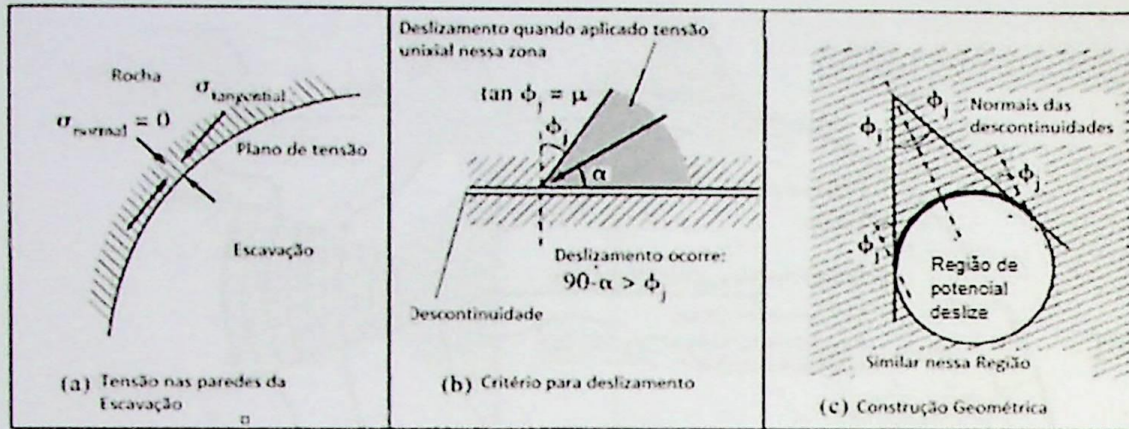


Figura 13 - Quadro resumo dos conceitos da teoria Φj (itens a,b,c) (Hudson & Harrison, 1997).

A teoria Φj exposta acima é muito usada na estabilidade de escavações subterrâneas, mas uma forma análoga pode ser usada para estabilidade de taludes, em especial às rupturas por tombamento flexural. Pode ser considerado que, similar ao caso de escavações, a tensão principal passa de paralela a perpendicular à face de escavação (que pode ser considerada a face do talude, nesse caso).

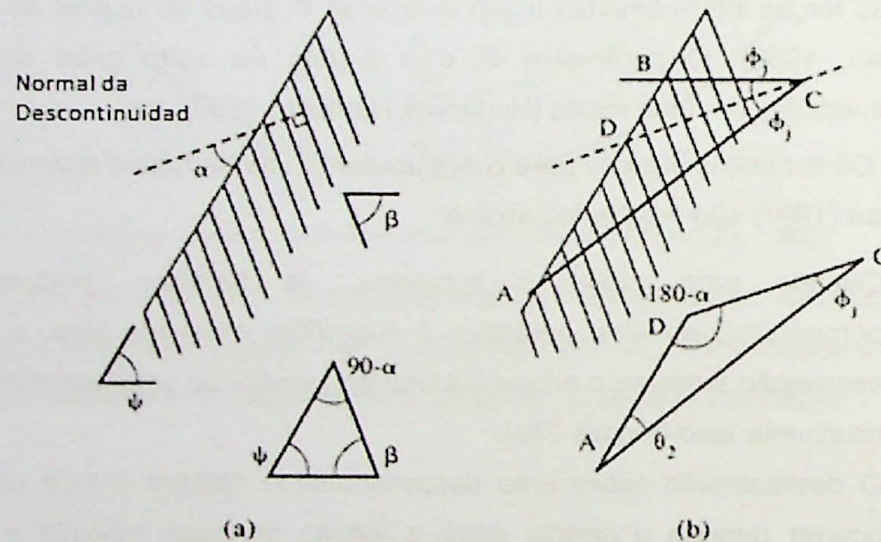


Figura 14 - Tombamento flexural (a) geometria e (b) análises Qj (Hudson & Harrison, 1997)

O potencial para deslizamentos intercamadas apresenta uma geometria ilustrada na Figura 14a. A Figura 14b mostra a geometria do sistema em um triângulo (ACD), onde pode-se obter a seguinte expressão:

$$\psi \geq 90 + \phi_j - \beta \quad (7)$$

Ou ainda,

$$\beta \geq \phi_j + (90 - \psi) \quad (8)$$



Onde  $\psi$  representa o mergulho do talude,  $\phi$  o ângulo de atrito e  $\beta$  o mergulho das descontinuidades. Os critérios, portanto, para ocorrer tombamento flexural de blocos, baseando-se no modelo acima, são:

- Deve existir uma família de descontinuidades mergulhando para dentro do talude, com um ângulo suficientemente alto para gerar deslizamentos intercamadas.
- O rumo do mergulho do talude deverá ter aproximadamente  $\pm 20^\circ$  de diferença do rumo do mergulho da descontinuidade analisada. Esse é um critério empírico onde foi observado que deslizamentos tendem a ocorrer com mais frequência quando as descontinuidades que os controlam apresentam um certo paralelismo com a face do talude.

A Figura 15 ilustra um estereograma genérico para análise de tombamento flexural (é usado  $\phi = 30^\circ$  e  $\psi = 75^\circ$ ) resumindo os critérios apresentados acima. O critério (a) é construído a partir dos grandes círculos. Já o critério (b) é construído a partir de linhas radiais.

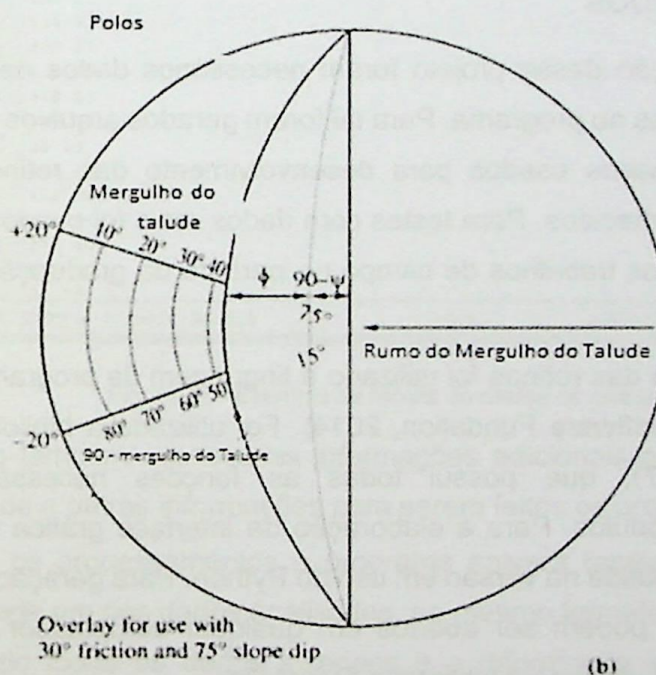


Figura 15 - Análise de rupturas de tombamento flexural (Hudson & Harrison, 1997).

Na Figura 16 é mostrado um resumo dos critérios para rupturas de tombamento flexural.



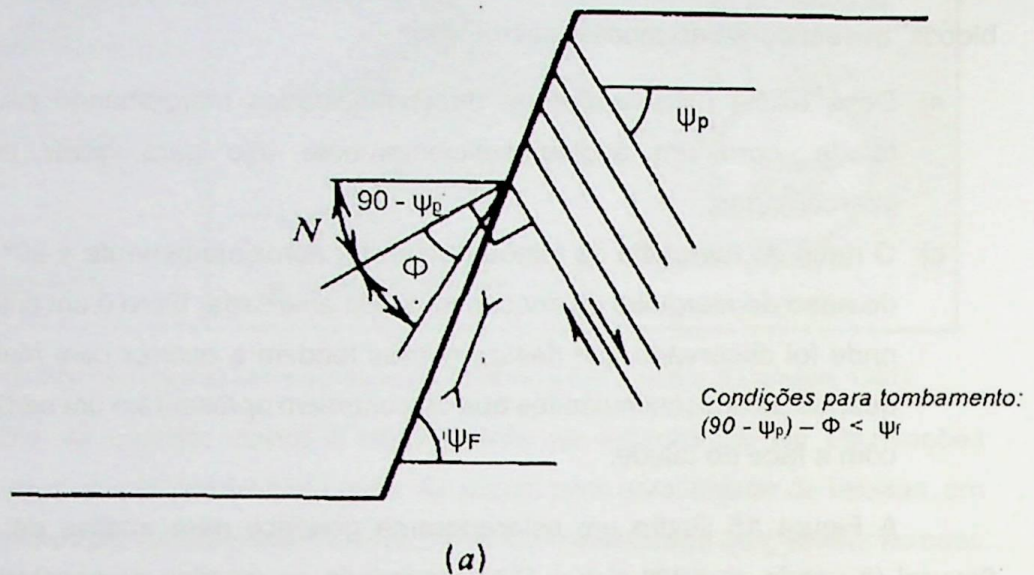


Figura 16 - Geometria para rupturas por tombamento (editado de Goodman, 1989).

#### 4. MATERIAIS E MÉTODOS

Para a elaboração desse projeto foram necessários dados de teste para a validação e busca de erros no programa. Para tal foram gerados arquivos aleatórios por meio dos próprios *softwares* usados para desenvolvimento das rotinas, e usados arquivos de dados já conhecidos. Para testes com dados reais foi usado um banco de dados coletado durante os trabalhos de campo no período da graduação( Anexos I a IV).

Para elaboração das rotinas foi utilizado a linguagem de programação Python na versão 2.7 (Python Software Foundation, 2014). Foi utilizada a biblioteca científica *Numpy* (Oliphant, 2007), que possui todas as funções necessárias para o desenvolvimento dos módulos. Para a elaboração da interface gráfica foi utilizado a biblioteca *Tkinter*, já embutida na versão em uso do Python. Para geração dos arquivos executáveis (.exe), que podem ser abertos em qualquer computador com sistema operacional Windows, foi utilizado a biblioteca *Pyinstaller*.

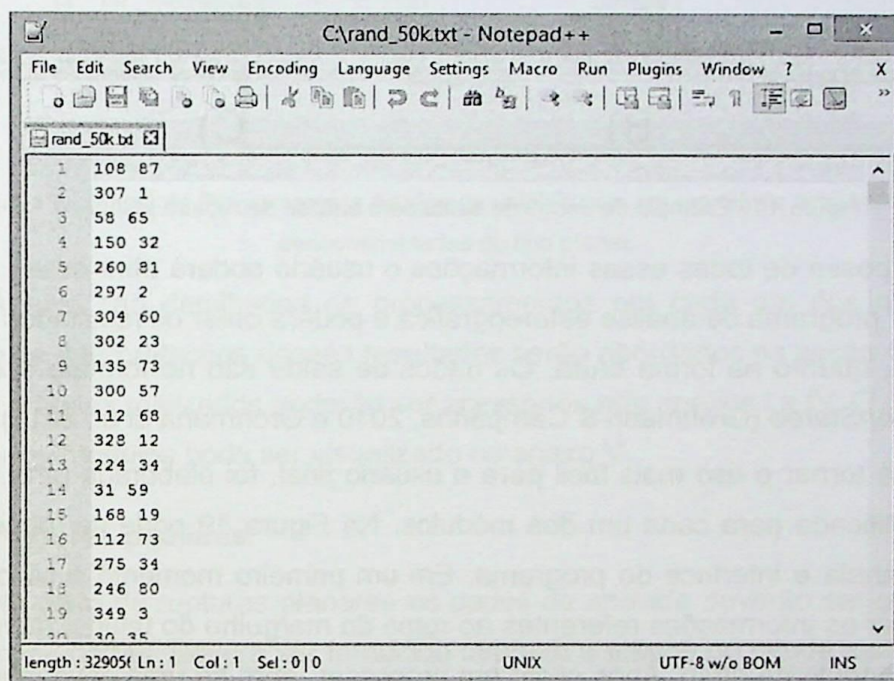
#### 5. RESULTADOS OBTIDOS

Foram desenvolvidas quatro rotinas computacionais independentes que realizam as análises cinemáticas para os principais tipos de rupturas em taludes rochosos: planares, em cunha, tombamento direto e tombamento flexural. Os dados de entrada em todas as rotinas são na forma de dados estruturais (rumo do mergulho e



mergulho) e os dados de saída são arquivos de texto, na forma de tabelas de dados, separadas por situações estáveis e instáveis, pelo critério geométrico.

É necessária a confecção de uma tabela contendo os dados estruturais no formato especificado e ser salva como ".txt" ou ".csv", que são extensões comuns e usuais para trabalhos com bancos de dados em geologia estrutural, além de ser o formato padrão do *software OpenStereo* (Grohmann & Campanha, 2010 e Grohmann et al, 2011). Na Figura 17 é exemplificado como devem ser apresentados os dados de entrada (das descontinuidades a serem analisadas). A primeira coluna refere-se ao rumo do mergulho e a segunda ao mergulho da estrutura.



|    |     |    |
|----|-----|----|
| 1  | 108 | 87 |
| 2  | 307 | 1  |
| 3  | 58  | 65 |
| 4  | 150 | 32 |
| 5  | 260 | 81 |
| 6  | 297 | 2  |
| 7  | 304 | 60 |
| 8  | 302 | 23 |
| 9  | 135 | 73 |
| 10 | 300 | 57 |
| 11 | 112 | 68 |
| 12 | 328 | 12 |
| 13 | 224 | 34 |
| 14 | 31  | 59 |
| 15 | 168 | 19 |
| 16 | 112 | 73 |
| 17 | 275 | 34 |
| 18 | 246 | 80 |
| 19 | 349 | 3  |
| 20 | 30  | 35 |

Figura 17 - Exemplo de tabela de dados de entrada.

Serão também necessárias informações adicionais como o ângulo de atrito, atitude do talude e outras informações para serem feitos os processamentos.

Após os processamentos o programa salvará tabelas com as informações referentes a cada um dos dados analisados, no mesmo formato acima. Será salvo uma tabela contendo todos os dados inseridos e a informação sobre a estabilidade ou instabilidade de cada um deles. Adicionalmente será salvo uma tabela somente com os dados considerados estáveis e outra com os dados considerados instáveis, para os critérios analisados. Será também gerado um arquivo com as condições limites. Na Figura 18 é mostrado um exemplo de análise de ruptura planar usando os dados do arquivo mostrado como exemplo acima. A tabela ilustrada em a) refere-se somente às descontinuidades **estáveis**, diferentemente de b) que mostra as **instáveis**. A tabela ilustrada em c) é composta por todos os dados analisados. Quando na terceira coluna



é apresentado o número 0 a descontinuidade em questão é **estável**, quando o número 1, **instável**.

The figure shows three screenshots of Notepad++ windows, each displaying a table of numerical data. The windows are titled 'C:\planarEST.txt - Notepad++', 'C:\planarINST.txt - Notepad++', and 'C:\testeplanar.txt - Notepad++'. Each table has three columns and multiple rows of data.

| Line | Col:1 | Col:2 | Col:3 |
|------|-------|-------|-------|
| 1712 | 52.0  | 56.0  |       |
| 1713 |       |       |       |
| 1714 | 52.0  | 57.0  |       |
| 1715 |       |       |       |
| 1716 | 52.0  | 58.0  |       |
| 1717 |       |       |       |
| 1718 | 52.0  | 59.0  |       |
| 1719 |       |       |       |
| 1720 | 52.0  | 60.0  |       |
| 1721 |       |       |       |
| 1722 | 52.0  | 61.0  |       |
| 1723 |       |       |       |
| 1724 | 52.0  | 62.0  |       |
| 1725 |       |       |       |
| 1726 | 52.0  | 63.0  |       |
| 1727 |       |       |       |
| 1728 | 52.0  | 64.0  |       |
| 1729 |       |       |       |
| 1730 | 52.0  | 65.0  |       |

| Line | Col:1 | Col:2 | Col:3 |
|------|-------|-------|-------|
| 2072 |       |       |       |
| 2073 | 354.0 | 58.0  |       |
| 2074 |       |       |       |
| 2075 | 12.0  | 39.0  |       |
| 2076 |       |       |       |
| 2077 | 349.0 | 53.0  |       |
| 2078 |       |       |       |
| 2079 | 345.0 | 36.0  |       |
| 2080 |       |       |       |
| 2081 | 359.0 | 46.0  |       |
| 2082 |       |       |       |
| 2083 | 340.0 | 46.0  |       |
| 2084 |       |       |       |
| 2085 | 346.0 | 34.0  |       |
| 2086 |       |       |       |
| 2087 | 349.0 | 65.0  |       |
| 2088 |       |       |       |
| 2089 | 358.0 | 36.0  |       |
| 2090 |       |       |       |

| Line | Col:1 | Col:2 | Col:3 | Col:4 |
|------|-------|-------|-------|-------|
| 1667 | 185.0 | 11.0  | 0.0   |       |
| 1668 |       |       |       |       |
| 1669 | 342.0 | 55.0  | 1.0   |       |
| 1670 |       |       |       |       |
| 1671 | 305.0 | 52.0  | 0.0   |       |
| 1672 |       |       |       |       |
| 1673 | 23.0  | 54.0  | 0.0   |       |
| 1674 |       |       |       |       |
| 1675 | 312.0 | 19.0  | 0.0   |       |
| 1676 |       |       |       |       |
| 1677 | 262.0 | 82.0  | 0.0   |       |
| 1678 |       |       |       |       |
| 1679 | 38.0  | 17.0  | 0.0   |       |
| 1680 |       |       |       |       |
| 1681 | 12.0  | 85.0  | 0.0   |       |
| 1682 |       |       |       |       |
| 1683 | 238.0 | 74.0  | 0.0   |       |
| 1684 |       |       |       |       |
| 1685 | 353.0 | 55.0  | 1.0   |       |

A)

B)

C)

Figura 18 - Exemplo de dados de saída para análise de ruptura planar.

De posse de todas essas informações o usuário poderá abrir esses arquivos em qualquer programa de análise estereográfica e poderá obter os resultados tanto na forma visual, quanto na forma bruta. Os dados de saída são no formato aceito pelo *software OpenStereo* (Grohmann & Campanha, 2010 e Grohmann et al, 2011).

Para tornar o uso mais fácil para o usuário final, foi elaborada uma interface gráfica simplificada para cada um dos módulos. Na Figura 19 pode ser observado o padrão de janela e interface do programa. Em um primeiro momento é solicitado ao usuário digitar as informações referentes ao rumo do mergulho do talude, mergulho do talude e ao ângulo de atrito. Após clicar em processar, abre-se uma nova janela onde deverá ser buscado o(s) arquivo(s) de texto a ser(em) lido(s). Depois de processado o programa salvará na mesma pasta os arquivos pós processados.



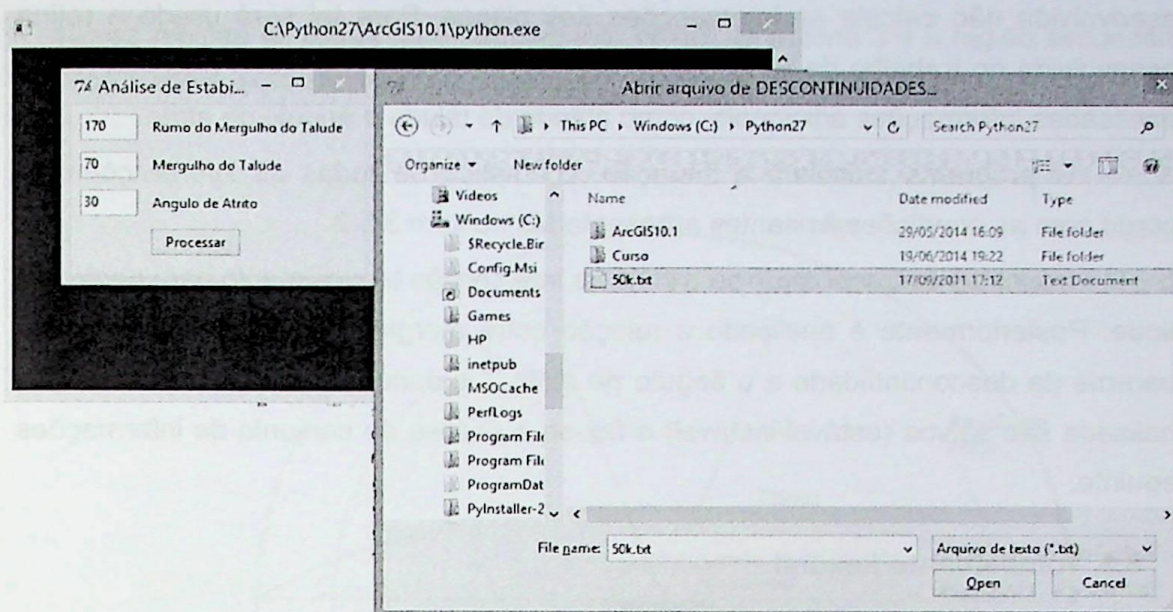


Figura 19 - Janela do Programa para Análise de estabilidade em um talude hipotético, para descontinuidades do tipo planar.

Abaixo são detalhados os processamentos em cada um dos módulos. As discussões e interpretações desses resultados serão abordados na seção 5.5. Demais exemplos e testes realizados poderão ser acessados nos anexos I a IV. O código fonte completo dos módulos pode ser visualizado no anexo V.

### 5.1. Rupturas planares

No caso de rupturas planares os dados de entrada deverão ser como planos de descontinuidades. Deverá ser fornecido também a atitude do talude a ser analisado e o ângulo de atrito considerado.

O programa calculará para cada uma das descontinuidades do banco de dados sua situação cinemática para os dados do talude e ângulo de atrito apresentados. As condições limites foram apresentadas no item **Erro! Fonte de referência não encontrada..**

É calculado a diferença entre o rumo do mergulho da descontinuidade e o rumo do mergulho do talude (nesse caso é usado o critério adotado por Hudson & Harrison (1997) de  $\pm 20^\circ$ ). Posteriormente é comparado o mergulho aparente da descontinuidade, com o ângulo de atrito, e o mergulho do talude. Feita essas análises o programa guarda os dados (instável/estável) e analisa a descontinuidade seguinte.

### 5.2. Rupturas em cunha

No caso de rupturas em cunha os dados de entrada deverão ser as intersecções entre todos os planos de descontinuidades analisados. A rotina



desenvolvida não calcula as intersecções dos planos. Para tal será usado a rotina desenvolvida no trabalho de Silva (2012). Similar ao caso de rupturas planares serão necessárias informações adicionais, como atitude do talude e ângulo de atrito.

O programa calculará a situação cinemática de todas as intersecções de acordo com as condições limitantes apresentadas no item 3.5.2.

Inicialmente é analisado se a linha de intersecção tem caimento para dentro do talude. Posteriormente é analisado a relação entre mergulho do talude, o mergulho aparente da descontinuidade e o ângulo de atrito. Os dados referentes a intersecção analisada são salvos (estável/instável) e faz-se a análise do conjunto de informações seguinte.

### **5.3. Tombamento flexural**

Para tombamento flexural os dados de entrada deverão ser como planos de descontinuidades. Deverá ser fornecido também a atitude do talude a ser analisado e o ângulo de atrito considerado.

O programa calculará para cada uma das descontinuidades a situação cinemática para os dados do talude e ângulo de atrito apresentados. As condições limites foram apresentadas no item 3.5.3.

Faz-se a análise se o plano de descontinuidade analisado tem caimento para fora do talude, e se o mesmo respeita o critério de paralelismo de  $\pm 20^\circ$  (Hudson & Harrison, 1997). Posteriormente é calculado a relação entre o mergulho do talude, o mergulho da descontinuidade e o ângulo de atrito.

### **5.4. Tombamento direto**

Diferentemente dos outros casos, para as análises de tombamento direto serão necessárias duas tabelas como dados de entrada. Uma com as informações dos planos de descontinuidades e outra com as intersecções entre elas.

O programa inicialmente fará a verificação em busca de planos de descontinuidade para formar a base para o tombamento de blocos. Se existirem, posteriormente será feita a análise de todas as intersecções dentro das condições limites apresentadas no item 3.5.3.

No caso de tombamento direto, em descontinuidades muito íngremes, o critério de paralelismo pode ser ampliado (Hudson & Harrison, 1997). Para esse caso específico expandiu-se o critério de  $\pm 20^\circ$  para  $\pm 45^\circ$  (dessa forma todas as descontinuidades que mergulham para dentro do talude serão consideradas). A partir desse critério são geradas três regiões de instabilidade que são ilustradas na Figura 20. As zonas 1 e 2



são as regiões primárias para tombamento direto. Já a zona 3 é a região secundária, onde é considerado tombamento oblíquo.

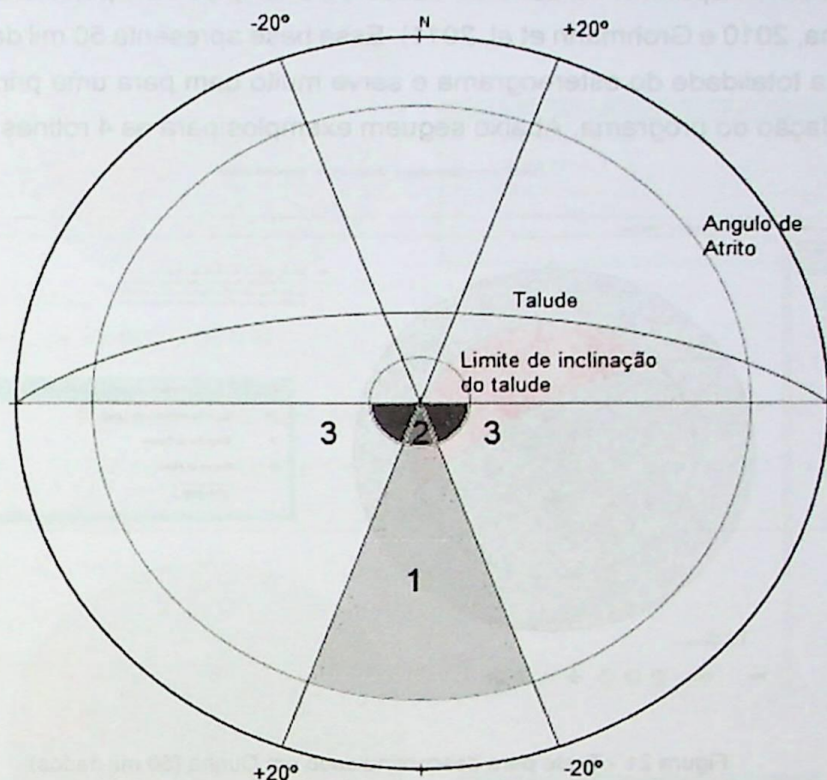


Figura 20 - Regiões de instabilidade para tombamento direto: 1 e 2 – regiões primárias. 3 – Região de tombamento oblíquo.

Deverá ser fornecido, além das tabelas de descontinuidades, atitude do talude e ângulo de atrito, a informação referente ao limite da inclinação do talude, ou seja, o quão íngreme o talude deverá ser, para que o critério de paralelismo entre o talude e a descontinuidade possa ser ampliado.

Os dados de saída serão similares aos dos outros módulos apresentados, porém serão geradas, além da tabela geral e da tabela com as descontinuidades estáveis, outras três tabelas referentes a cada uma das três áreas demarcadas no estereograma da Figura 20.

### 5.5. Testes Realizados

O programa gerou resultados esperados, de acordo com critérios adotados e comparados com os métodos manuais. Inicialmente foram feitos testes com o arquivo



"rand\_50k.txt", disponível na base de dados de exemplos do *OpenStereo* (Grohmann & Campanha, 2010 e Grohmann et al, 2011). Essa base apresenta 50 mil dados aleatórios cobrindo a totalidade do estereograma e serve muito bem para uma primeira análise e para validação do programa. Abaixo seguem exemplos para as 4 rotinas apresentadas.

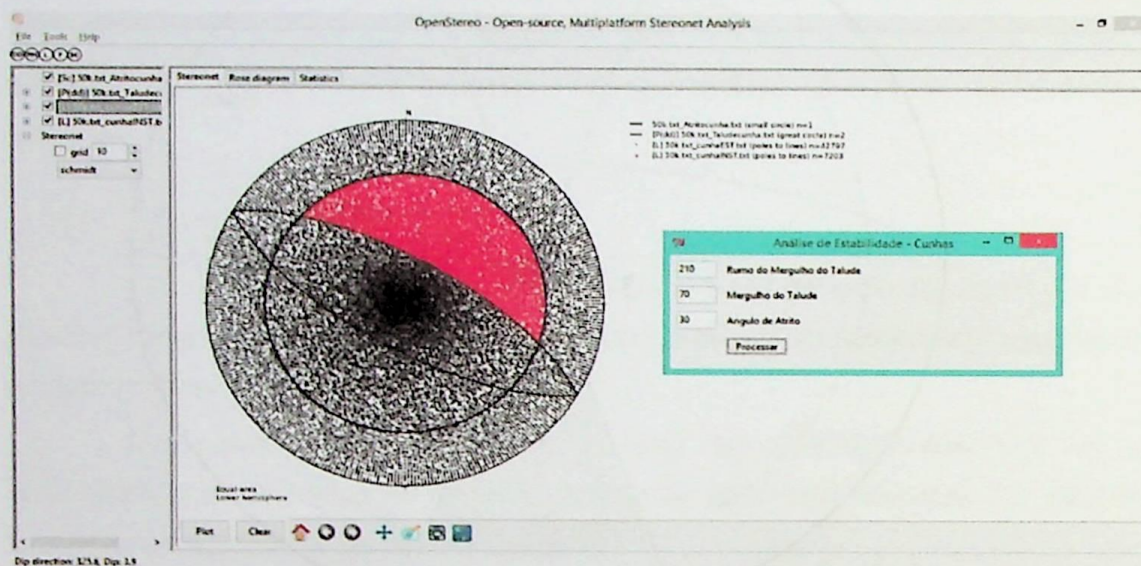


Figura 21 - Teste para descontinuidade em Cunha (50 mil dados)

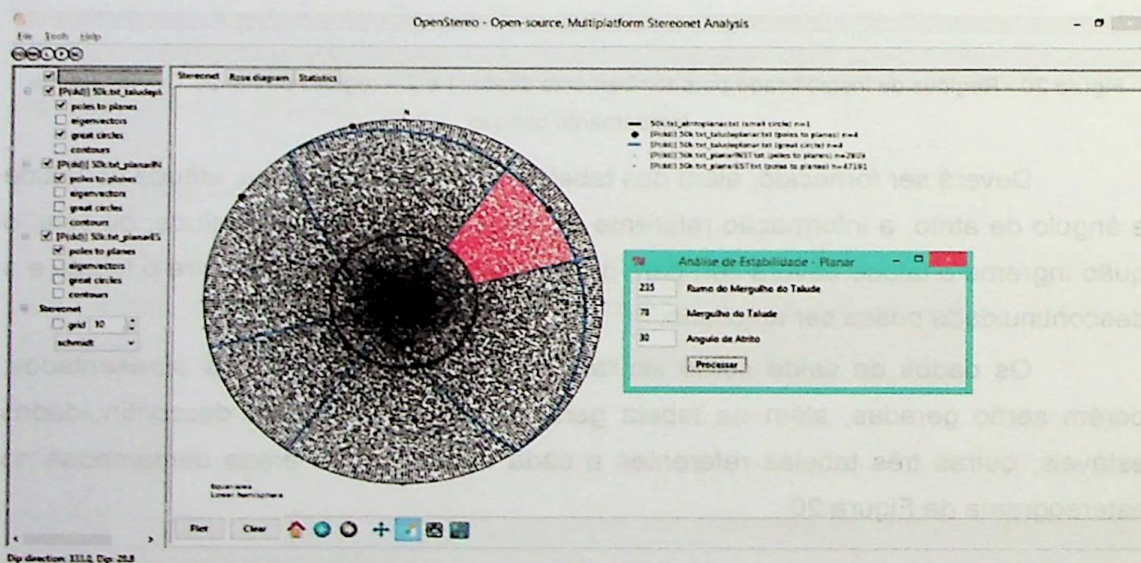


Figura 22 - Teste para descontinuidade do tipo planar (50 mil dados)



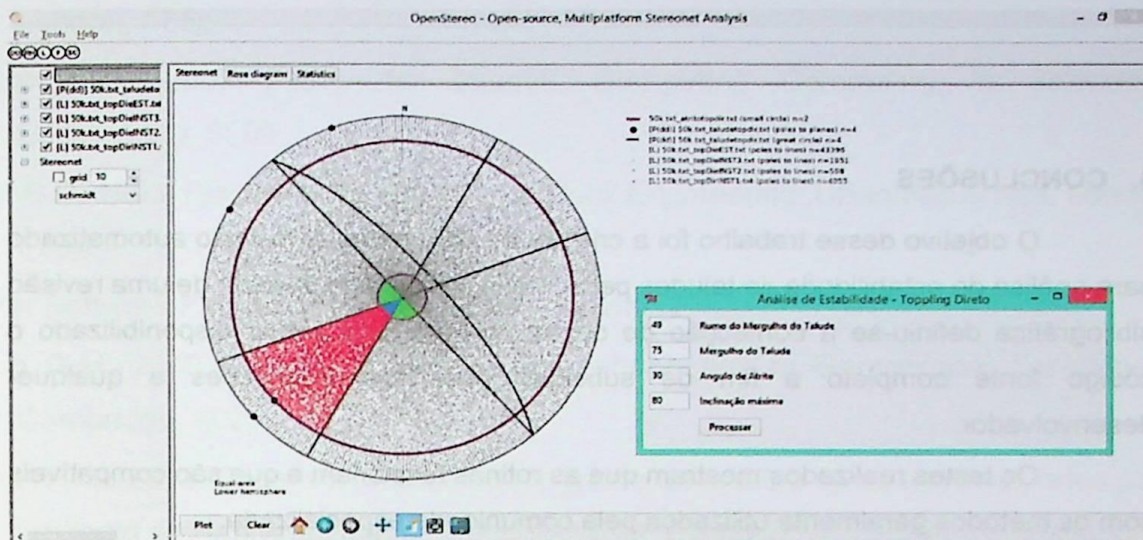


Figura 23 - Teste para tombamento direto (50 mil dados).

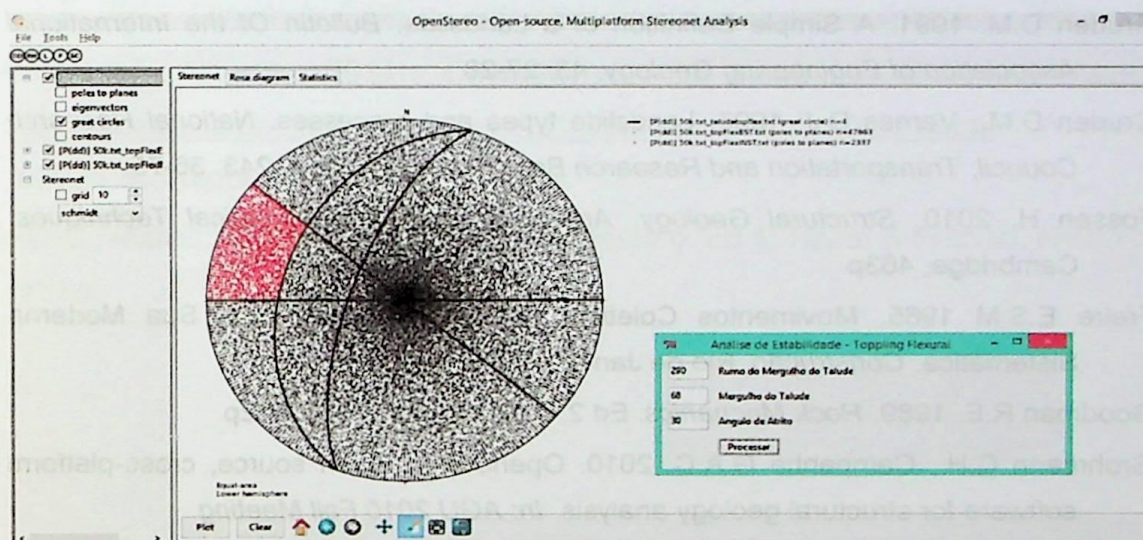


Figura 24 - Teste para tombamento flexural (50 mil dados).

Como pode ser observado nas figuras acima, o resultado se deu conforme o esperado nos taludes testados. As áreas com cores em vermelho indicam regiões de instabilidade pelo critério geométrico (no caso de tombamento direto, as regiões instáveis são divididas em três – vermelho, azul e verde). Ocorre correlação entre os cálculos efetuados pelo programa e as áreas esperadas. Outros exemplos podem ser visualizados nos anexos de I a IV.



## 6. CONCLUSÕES

O objetivo desse trabalho foi a criação de uma rotina de cálculo automatizado para análise de estabilidade de taludes pelo critério geométrico. A partir de uma revisão bibliográfica definiu-se a confecção de quatro rotinas. Está sendo disponibilizado o código fonte completo a fim de subsidiar possíveis adaptações a qualquer desenvolvedor.

Os testes realizados mostram que as rotinas funcionam e que são compatíveis com os métodos geralmente utilizados pela comunidade especializada.

## 7. REFERÊNCIAS BIBLIOGRÁFICAS

- Cruden D.M. 1991. A Simple Definition of a Landslide. *Bulletin Of the International Association of Engineering Geology*, 43: 27-28.
- Cruden D.M., Varnes D.J. 1996. Landslide types and processes. *National Research Council, Transportation and Research Board Special Report*, 243: 36-75.
- Fossen H. 2010. *Structural Geology. An Introduction to Geometrical Techniques*. Cambridge, 463p
- Freire E.S.M 1965. Movimentos Coletivos de Solos e Rochas e Sua Moderna Sistemática. *Construção*, Rio de Janeiro, 8, p.: 10-18.
- Goodman R.E. 1989. *Rock Mechanics*. Ed 2, John Wiley & Sons, 562p
- Grohmann C.H., Campanha G.A.C. 2010. OpenStereo: open source, cross-platform software for structural geology analysis. *In: AGU 2010 Fall Meeting*
- Grohmann C.H., Campanha G.A.C., Soares Junior, A.V. 2011. OpenStereo: um programa Livre e multiplataforma para análise de dados estruturais. *In: XIII Simpósio Nacional de Estudos Tectônicos*
- Guidicini G, Nieble C.M. 1984. *Estabilidade de Taludes Naturais e de Escavação*. Editora Blucher, São Paulo, 194p
- Hoek E., Bray J. 1974. *Rock Slope Engineering*. Ed 1, Inst. Of Mining and Metallurgy, London, 309p
- Hoek E., Bray J. 1981. *Rock Slope Engineering*. Ed 3, Inst. Of Mining and Metallurgy, London, 358p
- Hoek E., Brown 1980. *Underground excavation in rock*. Inst. Of Mining and Metallurgy, London, 527p
- Hudson J.A., Harrison J.P. 1997. *Engineering Rock Mechanics - An introduction to the Principles*. Elsevier Science Ltd., Oxford - UK, 444p.

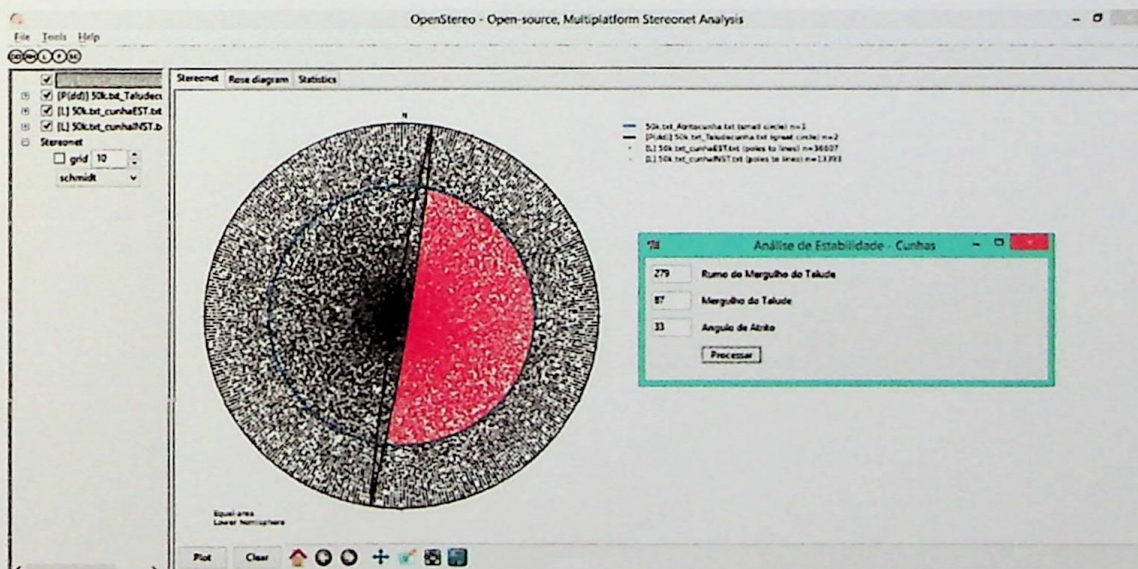
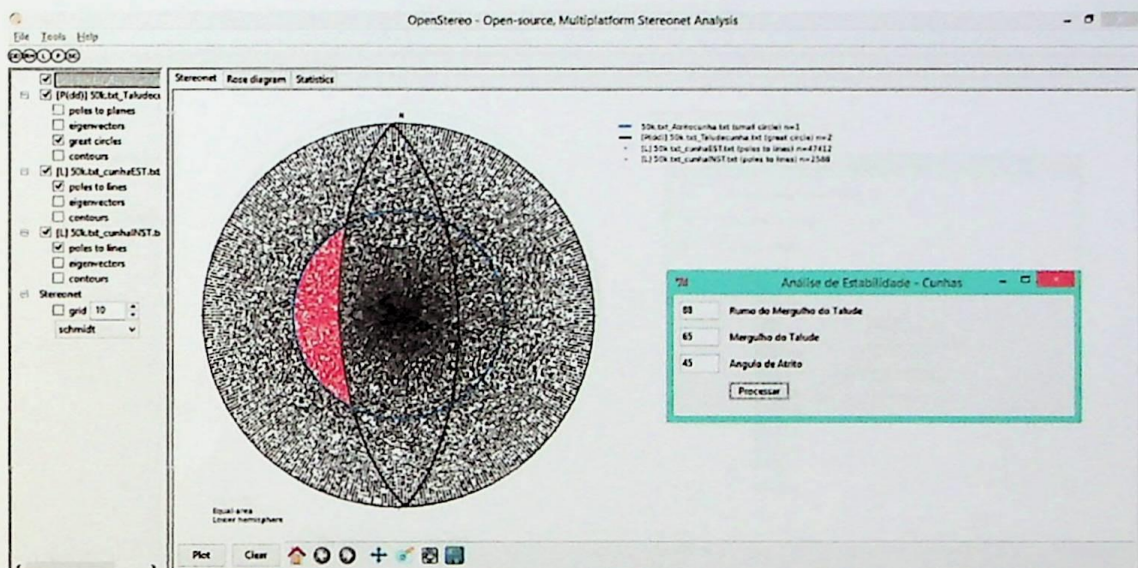
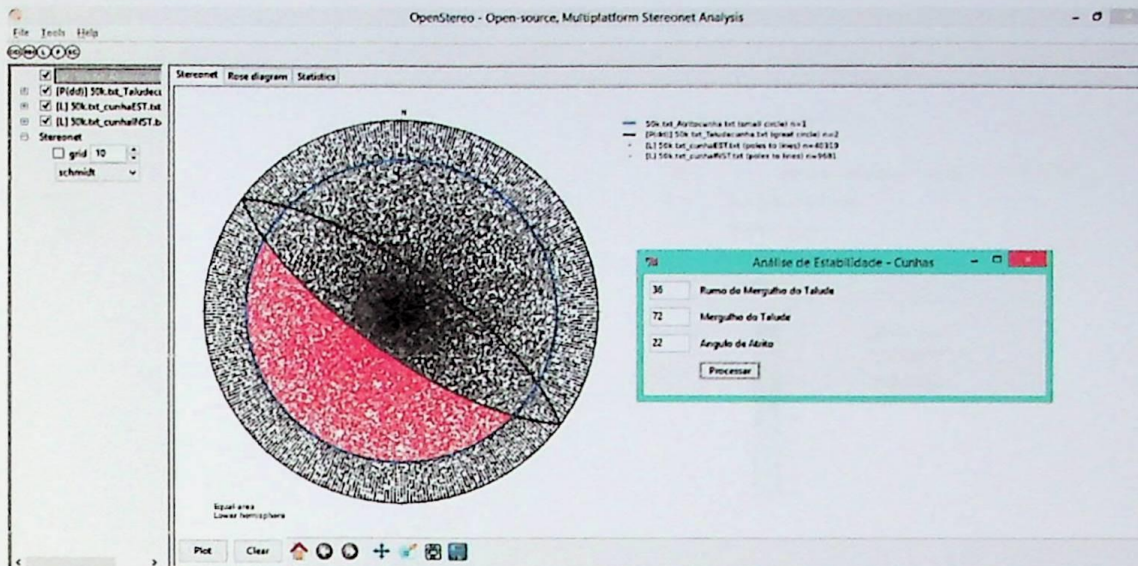


- Jaeger J.C., Cook N.G.W. 1969. *Fundamentals of Rock Mechanics*. Methuen & Co Ltda, London, 585p
- Oliphant, T.E. 2007. Python for Scientific Computing. Computing. *In: Science & Engineering*, 9: 90
- Priest S.D. 1993. *Discontinuity Analysis for Rock Engineering*. Chapman & Hall, London, 473p.
- Python Software Foundation 2014. Python - Version 2.7.2.
- Ragan D.M. 2009. *Structural Geology. An Introduction to Geometrical Techniques*. Cambridge, 602p
- Silva L. A. U. 2012. Desenvolvimento de programa de análise de descontinuidades para geologia estrutural aplicada. Monografia de Trabalho de Formatura. Instituto de Geociências – Universidade de São Paulo, p. 23.
- Thiebes B. 2012. *Landslide Analysis and Early Warning Systems*. Springer, 266p
- University of Alberta. 2011. *Orientation data analysis*. Disponível em : <https://courses.eas.ualberta.ca/eas421/lecturepages/orientation.html>. Acessado em 23 de outubro de 2014
- Wyllie D.C., Mah C.W. 2004. *Rock Slope Engineering*. Ed 4, Taylor & Francis, New York, 431p

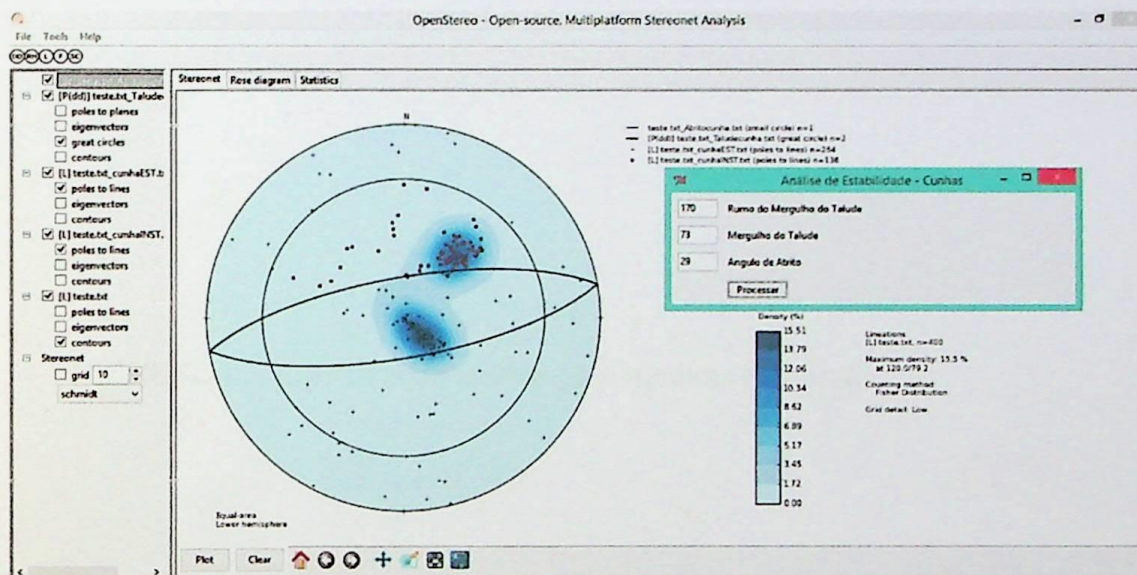
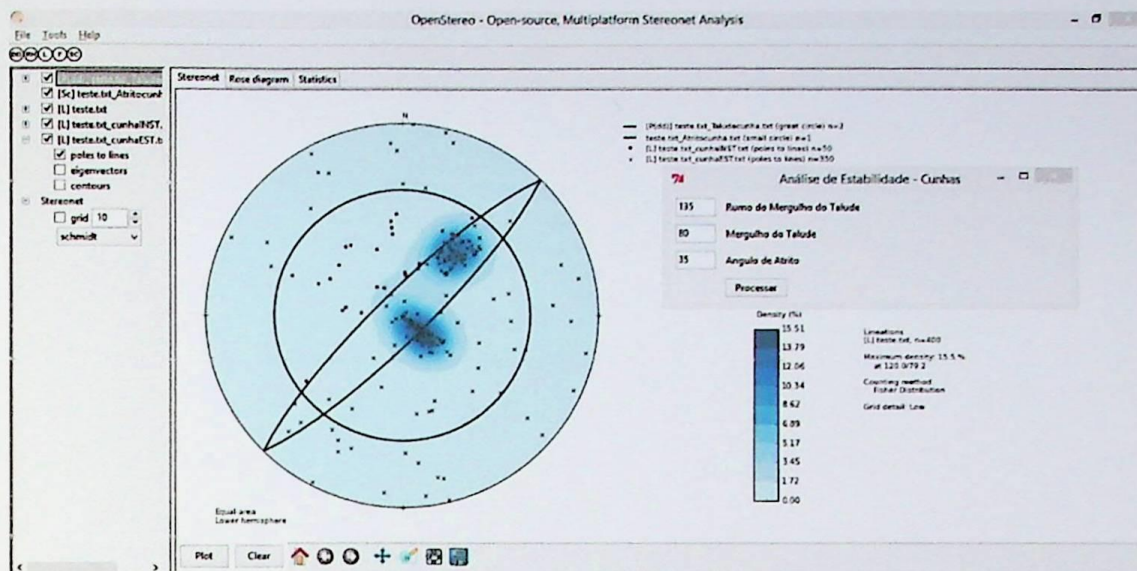


## ANEXO I – Exemplos de análise para rupturas em cunha





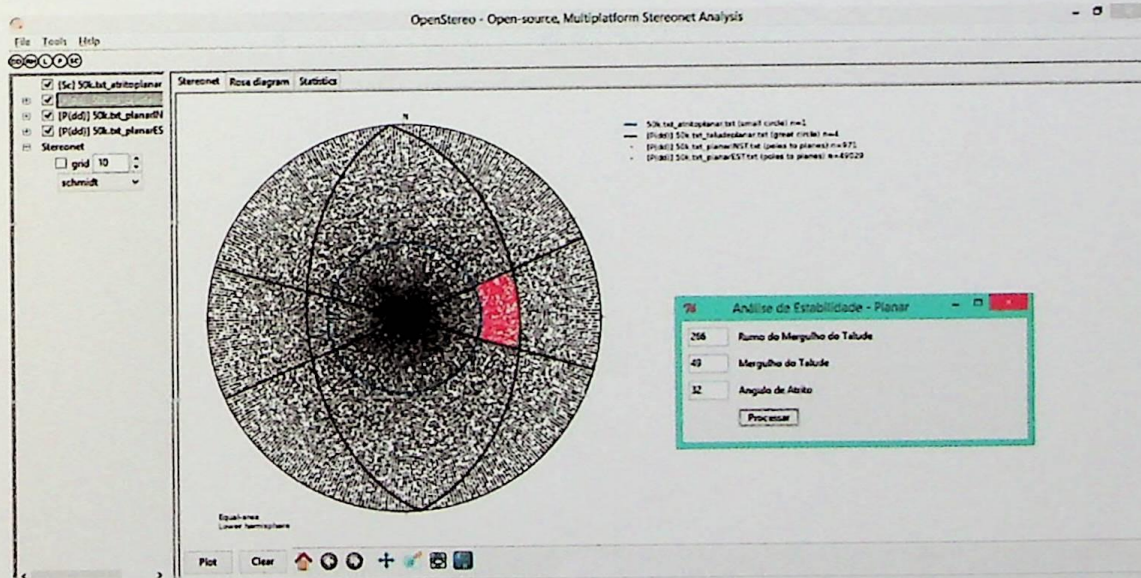
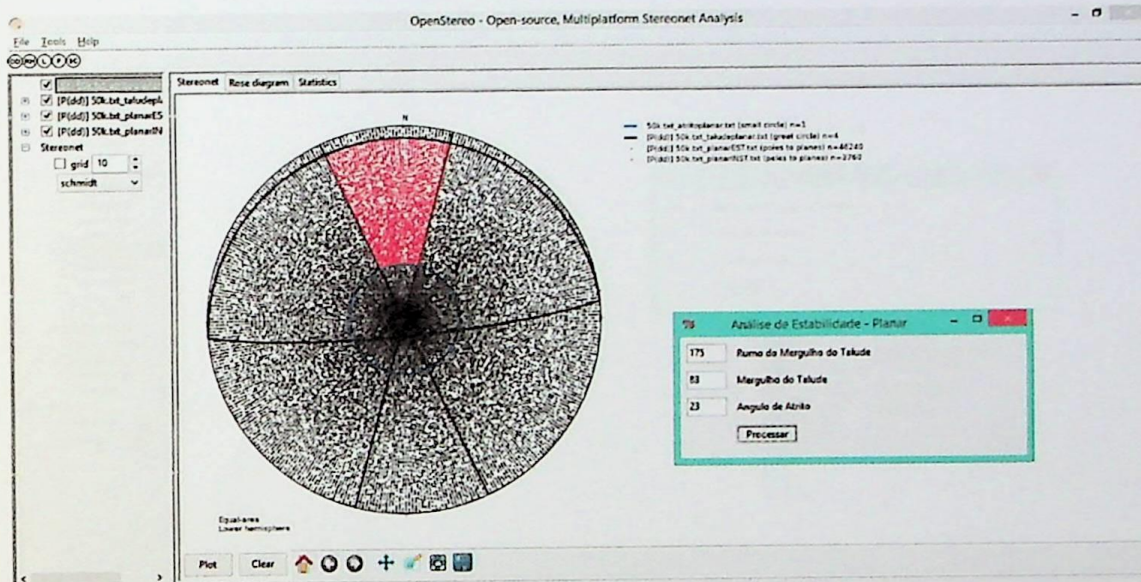
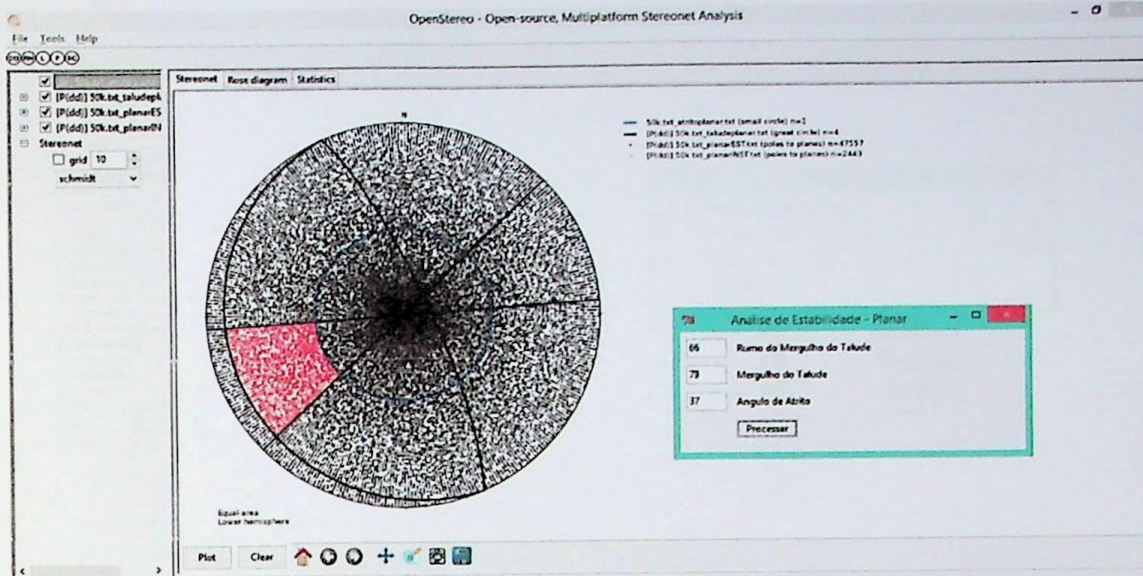




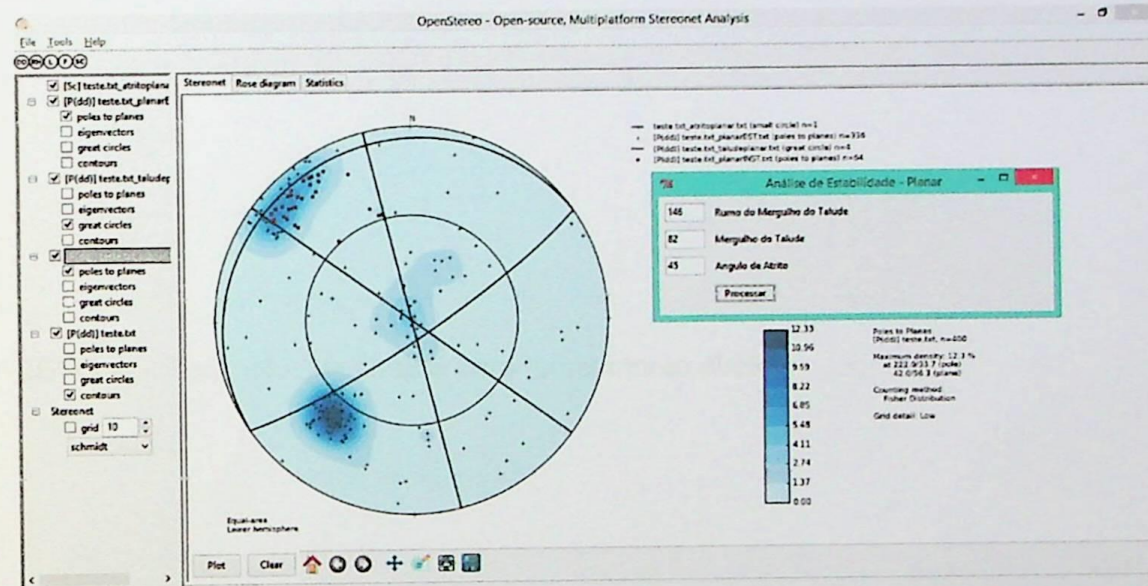
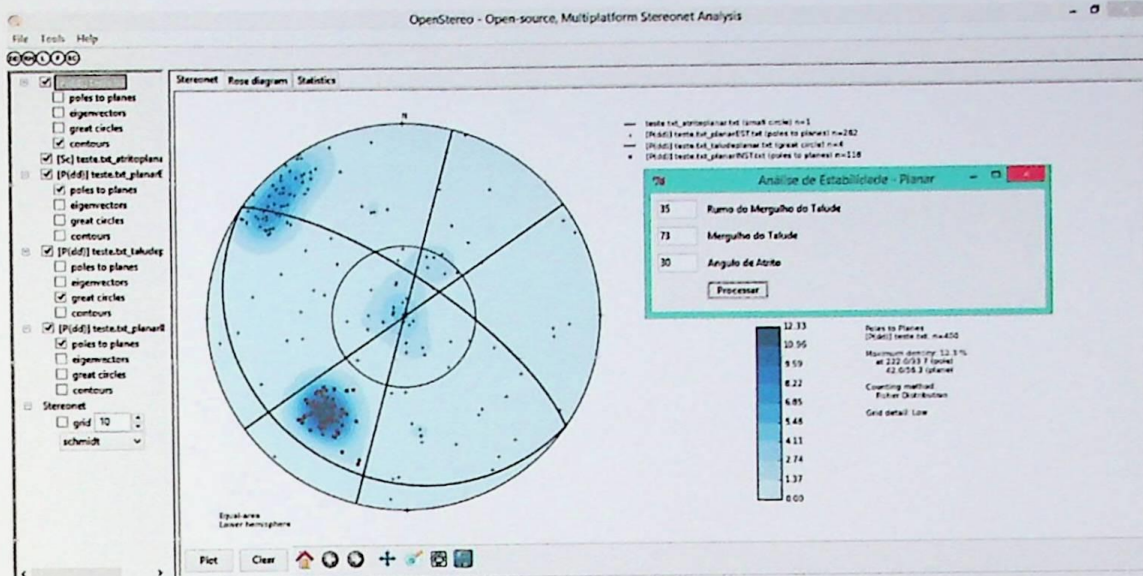


## ANEXO II – Exemplos de análise para rupturas planares





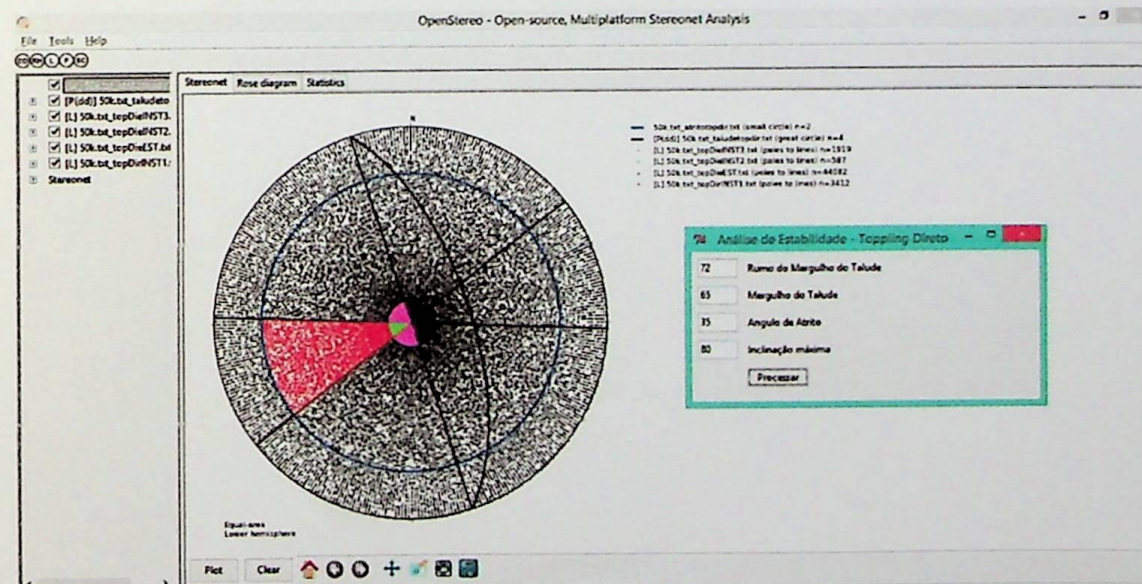
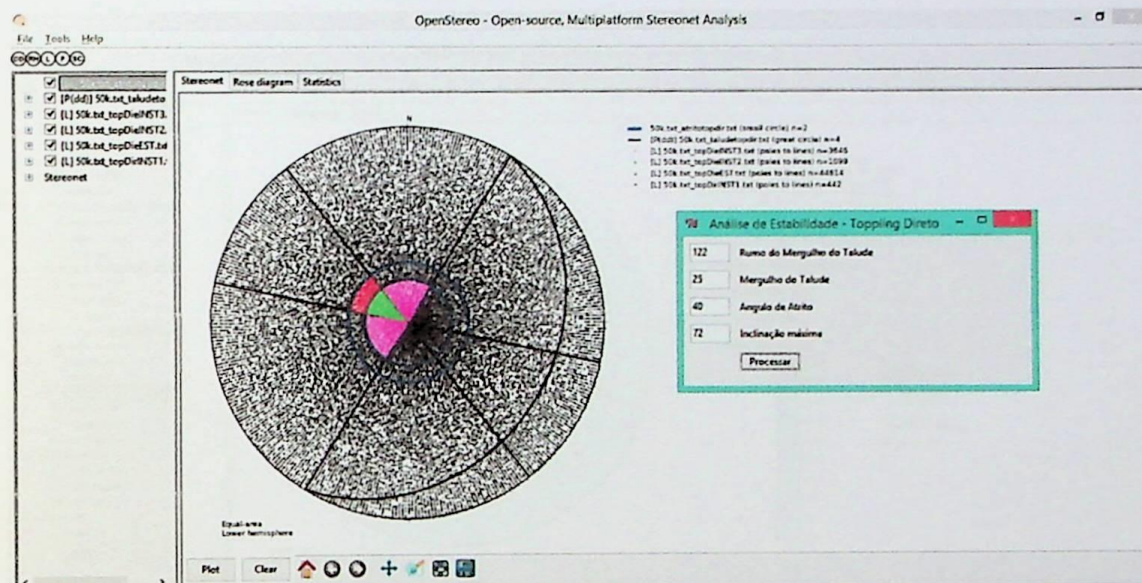
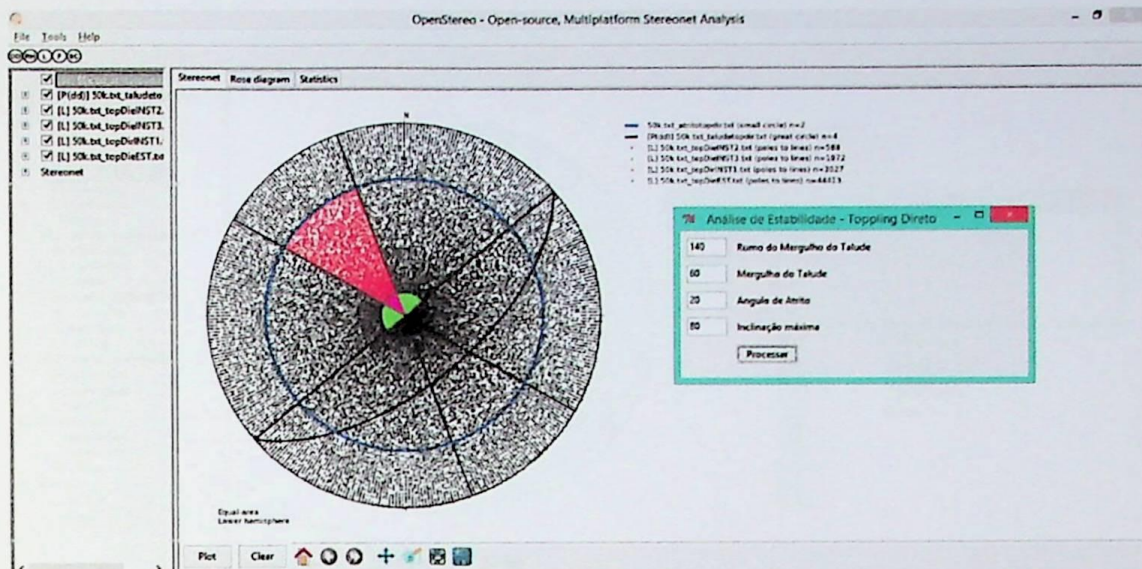




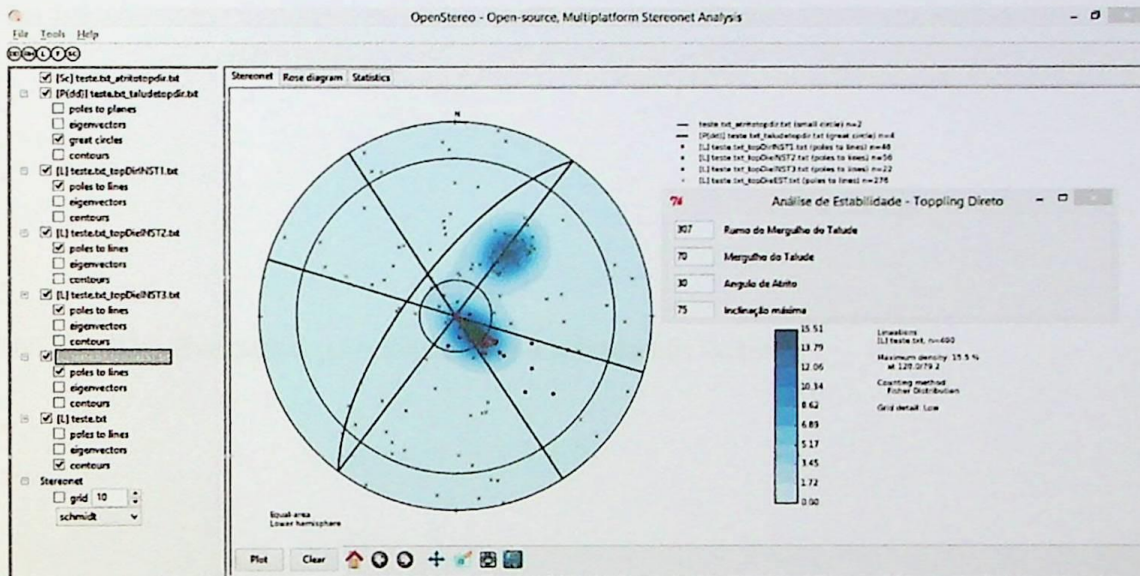
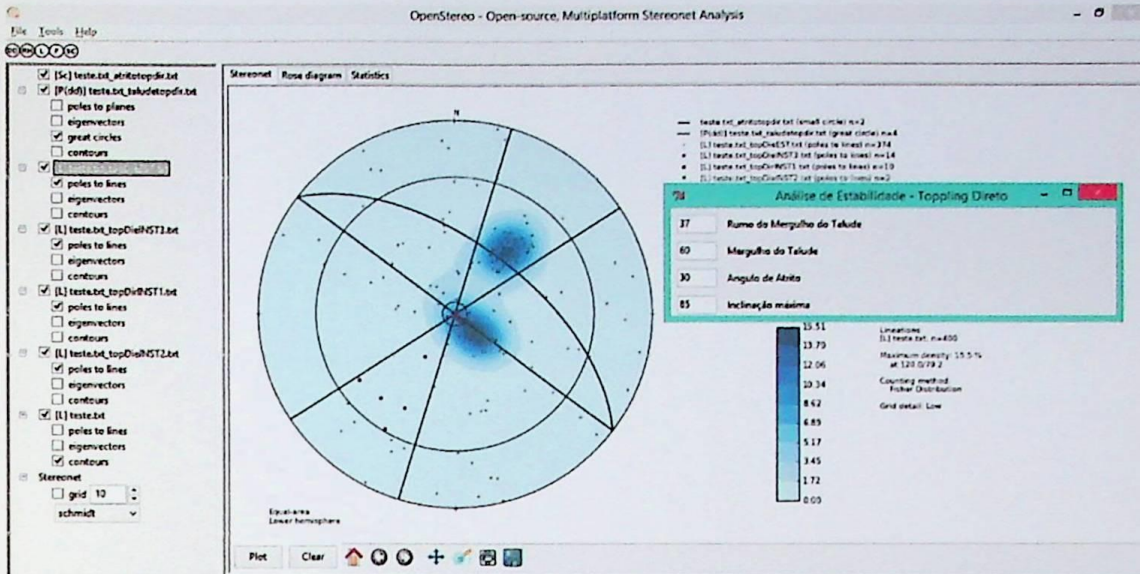


### ANEXO III - Exemplos de análise para tombamento direto





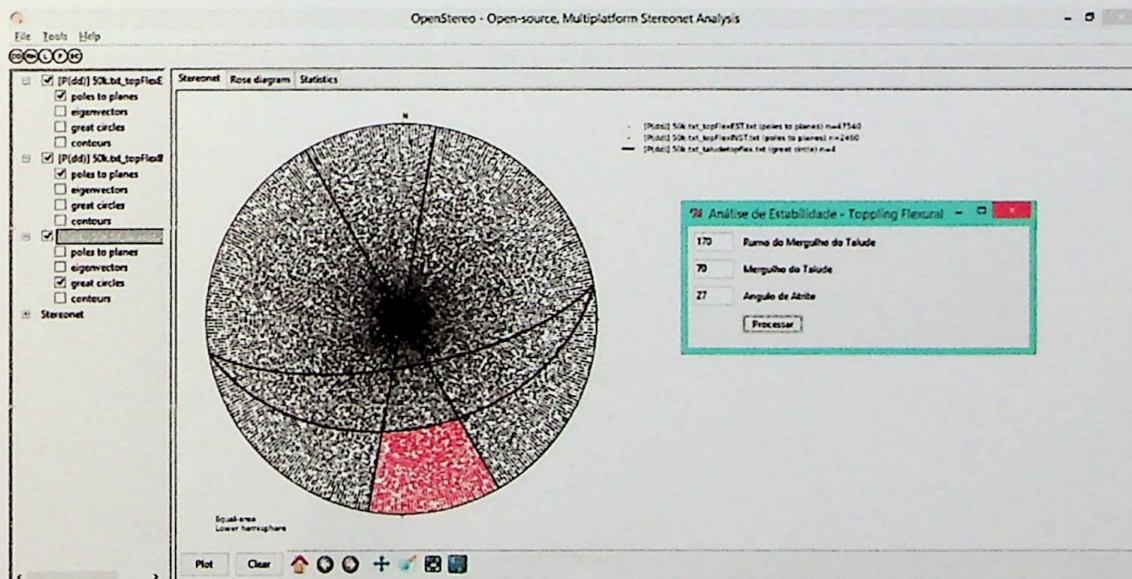
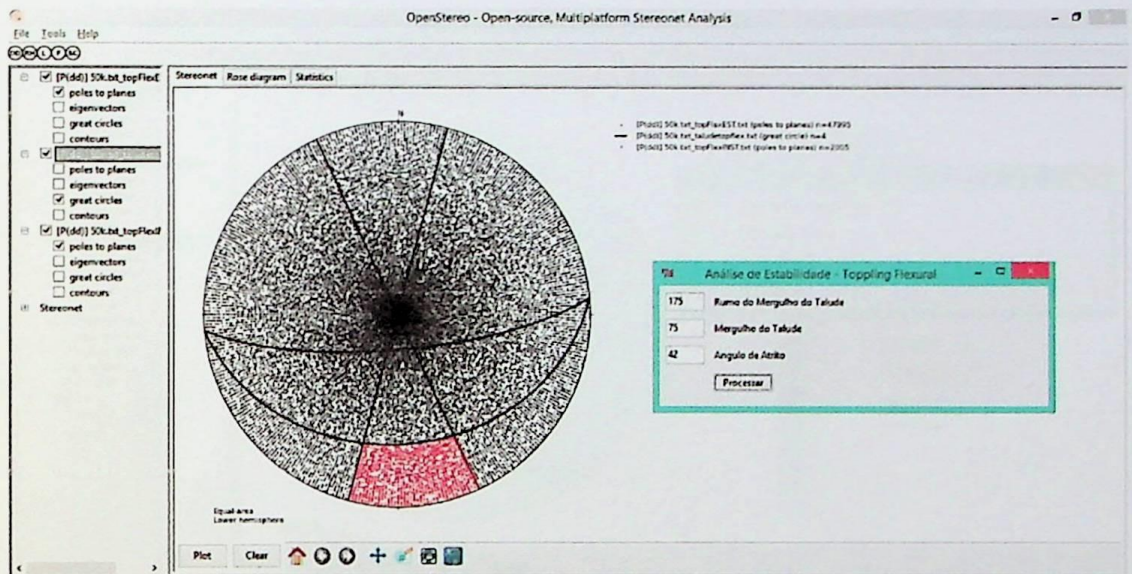
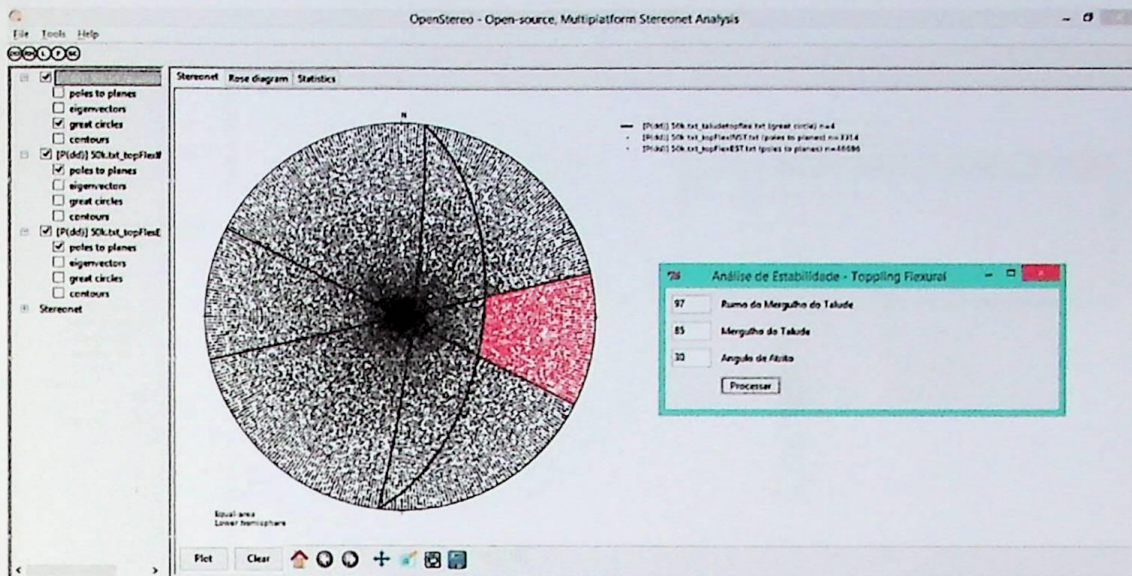




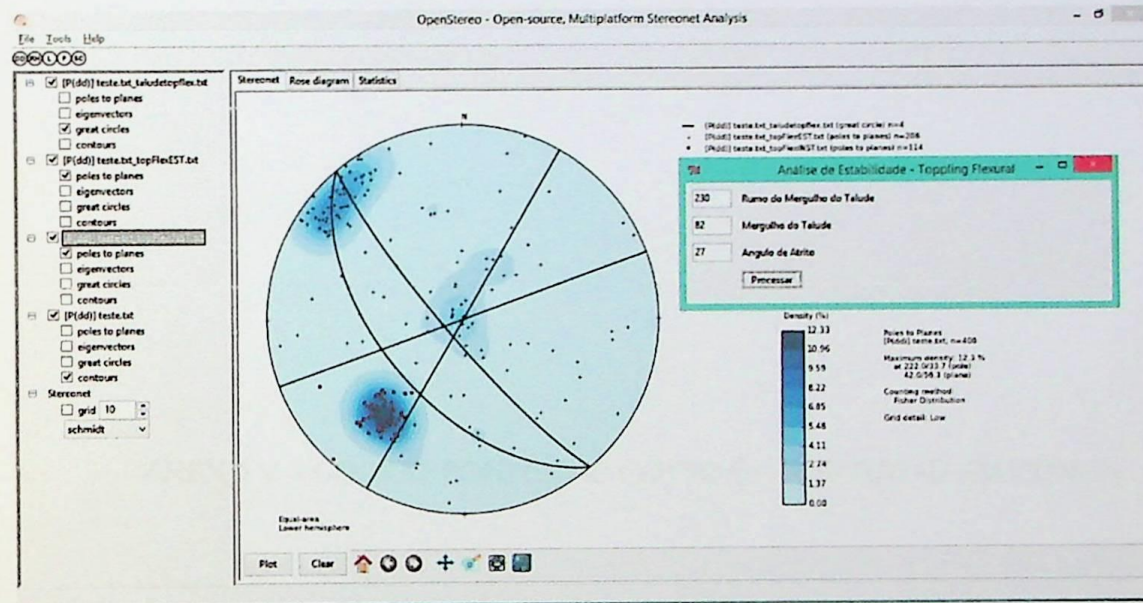
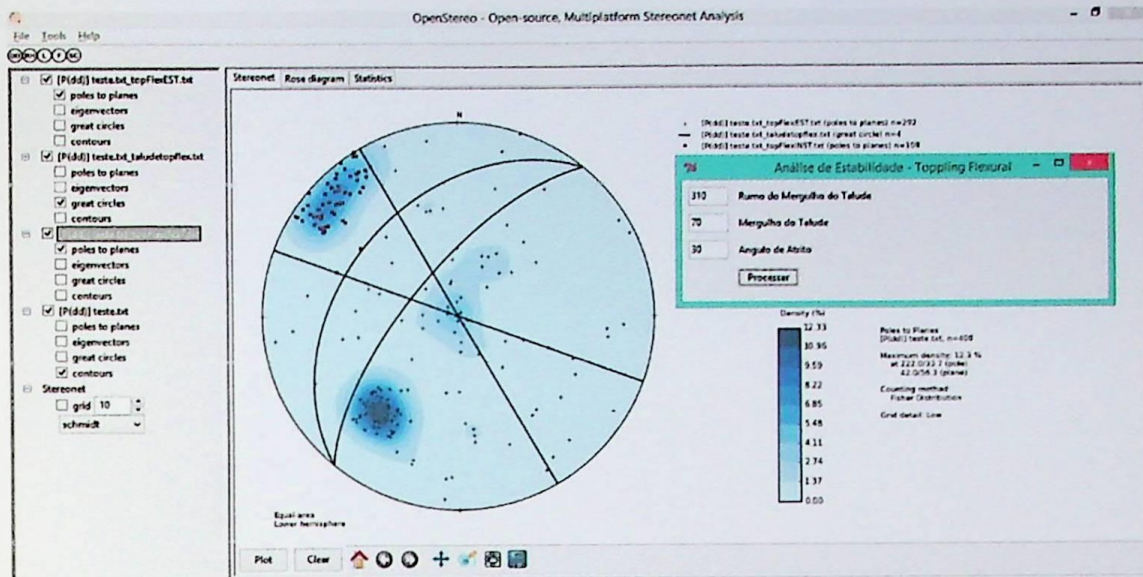


ANEXO IV - Exemplos de análise para tombamento flexural















```

# -*- coding: utf-8 -*-
import numpy as np
from numpy import arccos, sin, cos
import csv

def merg_ap(dipdir1,dip1,dipdir2):
#Rumo do mergulho e mergulho

tg=np.fabs(np.degrees(np.arctan(np.tan(np.radians(dip1))*np.cos(np.fabs(np.radians(dipdir2-dipdir1))))))
    return tg

def risco_cunha(data,dipdir,dip,at):
#data - tabela com a linha de interseccao [dipdir , dip] entre dois planos.
n=data.shape[0]
tab=np.zeros((n,3))
for i in range(0,n):
    tab[i]=(data[i,0],data[i,1],0)
for i in range(0,n):
    if tab[i,0]>180:
        res= merg_ap(dipdir,dip,tab[i,0]-180)
    else:
        res=merg_ap(dipdir,dip,tab[i,0]+180)
    dif = ((arccos((sin(np.pi*tab[i,0]/180.)*sin(dipdir*(np.pi/180.)) +
cos(tab[i,0]*(np.pi/180.))*cos(dipdir*(np.pi/180.)))))*180./np.pi)
    if dif >90:
        if tab[i,1]>at:
            if tab[i,1]<res:
                #considera-se 1 sendo instavel e 0 estável
                tab[i,2]=1
    return tab

def processar(entrada,dipdir,dip,at):
    data = np.loadtxt(entrada)

    tabela=risco_cunha(data,dipdir,dip,at)

    #imprimindo a tabela com TODAS descontinuidades
    write=csv.writer(open(entrada+"_saidatotacunha.txt" , 'w'),dialect='excel-
tab')
    write.writerows(tabela)

    #separando em duas tabelas (instavel e estável):

    c=0
    for i in range (0,tabela.shape[0]):
        if tabela[i,2]==1:
            c=c+1

    tabela_estavel=np.zeros(((tabela.shape[0]-c),2))
    tabela_instavel=np.zeros((c,2))
    a=0
    b=0
    for j in range(0,tabela.shape[0]):
        if tabela[j,2]==1:
            tabela_instavel[a]=(tabela[j,0],tabela[j,1])
            a=a+1
        else:
            tabela_estavel[b]=(tabela[j,0],tabela[j,1])
            b=b+1

    #imprimindo os critérios limites
    talude=np.zeros((2,2))
    atrito=np.zeros((1,3))
    talude[0]=(dipdir,dip)
    if dipdir<180:
        talude[1]=(dipdir+180,dip)
    else:

```



```

        talude[1]=(dipdir-180,dip)
        atrito[0]=(0,90,90-at)

        #Salvando os arquivos
        write2=csv.writer(open(entrada+'_cunhaINST.txt' , 'w'),dialect='excel-tab')
        write2.writerows(tabela_instavel)
        write3=csv.writer(open(entrada+'_cunhaEST.txt' , 'w'),dialect='excel-tab')
        write3.writerows(tabela_estavel)
        write4=csv.writer(open(entrada+'_Taludecunha.txt' , 'w'),dialect='excel-
tab')
        write4.writerows(talude)
        write4=csv.writer(open(entrada+'_Atritocunha.txt' , 'w'),dialect='excel-
tab')
        write4.writerows(atrito)

if __name__ == "__main__":

    #Interface gráfica:

    from Tkinter import *
    import ttk
    from tkFileDialog import askopenfilename

    def calculate(*args):
        try:
            dipdir_ = float(dipdir.get())
            dip_ = float(dip.get())
            at_ = float(at.get())
            entrada = askopenfilename(filetypes=[('Arquivo de texto', '.txt')],
title='Abrir arquivo de INTERSECCOES...').encode(sys.getfilesystemencoding())
            if entrada:

                processar(entrada,dipdir_,dip_,at_)

        except ValueError:
            pass

    root = Tk()
    root.title("Análise de Estabilidade - Cunhas")

    mainframe = ttk.Frame(root, padding="3 3 12 12")
    mainframe.grid(column=0, row=0, sticky=(N, W, E, S))
    mainframe.columnconfigure(0, weight=1)
    mainframe.rowconfigure(0, weight=1)

    dipdir = StringVar()
    dip = StringVar()
    at = StringVar()
    #status = StringVar()

    dipdir_entry = ttk.Entry(mainframe, width=7, textvariable=dipdir)
    dipdir_entry.grid(column=2, row=1, sticky=(W, E))
    ttk.Label(mainframe, text="Rumo do Mergulho do Talude").grid(column=3,
row=1, sticky=W)

    dip_entry = ttk.Entry(mainframe, width=7, textvariable=dip)
    dip_entry.grid(column=2, row=2, sticky=(W, E))
    ttk.Label(mainframe, text="Mergulho do Talude").grid(column=3, row=2,
sticky=W)

    at_entry = ttk.Entry(mainframe, width=7, textvariable=at)
    at_entry.grid(column=2, row=3, sticky=(W, E))
    ttk.Label(mainframe, text="Angulo de Atrito").grid(column=3, row=3,
sticky=W)

    #ttk.Label(mainframe, textvariable=status).grid(column=2, row=4, sticky=(W,
E))

```



```
    ttk.Button(mainframe, text="Processar", command=calculate).grid(column=3,
row=4, sticky=W)

    for child in mainframe.winfo_children(): child.grid_configure(padx=5,
pady=5)

    dipdir_entry.focus()
    root.bind('<Return>', calculate)

    root.mainloop()
```

ANEXO VI - COORDENAÇÃO E CRIAÇÃO DE MATERIAIS EDUCACIONAIS



## ANEXO VI – CÓDIGO FONTE DAS ROTINAS – RUPTURAS PLANARES



```

# -*- coding: utf-8 -*-

import numpy as np
from numpy import arccos, sin, cos
import csv

def merg_ap(dipdir1,dip1,dipdir2):
#rumo do mergulho e mergulho

tg=np.fabs(np.degrees(np.arctan(np.tan(np.radians(dip1))*np.cos(np.radians(dipdir2-dipdir1))))))
return tg

def risco_planar(data,dipdir,dip,at):
#data é uma tabela com [dipdir , dip] das descontinuidades
n=data.shape[0]
tab=np.zeros((n,3))
for i in range (0,n):
    tab[i]=(data[i,0],data[i,1],0)
for i in range(0,n):
    res = merg_ap(dipdir,dip,tab[i,0])
    dif = ((arccos((sin(np.pi*tab[i,0]/180.)*sin(dipdir*(np.pi/180.)) +
cos(tab[i,0]*(np.pi/180.))*cos(dipdir*(np.pi/180.)))))*180./np.pi)
    if dif < 20:
        if tab[i,1]<res:
            if tab[i,1]>at:
                tab[i,2]=1
return tab

def processar(entrada,dipdir,dip,at):
data = np.loadtxt(entrada)

tabela=risco_planar(data,dipdir,dip,at)

#imprimindo a tabela com TODAS descontinuidades

write=csv.writer(open(entrada+"_saidatotalplanar.txt"
'w'),dialect='excel-tab')
write.writerows(tabela)

#separando em duas tabelas (instavel e estável):

c=0
for i in range (0,tabela.shape[0]):
    if tabela[i,2]>0:
        c=c+1

tabela_estavel=np.zeros(((tabela.shape[0]-c),2))
tabela_instavel=np.zeros((c,2))
a=0
b=0
for j in range(0,tabela.shape[0]):
    if tabela[j,2]>0:
        tabela_instavel[a]=(tabela[j,0],tabela[j,1])
        a=a+1
    else:
        tabela_estavel[b]=(tabela[j,0],tabela[j,1])
        b=b+1
#imprimindo os critérios limites
talude=np.zeros((4,2))
atrato=np.zeros((1,3))
talude[0]=(dipdir,dip)
if dipdir<180:
    talude[1]=(dipdir+180,90-dip)
    talude[2]=(dipdir+70,90)
    talude[3]=(dipdir+110,90)
else:
    talude[1]=(dipdir-180,90-dip)

```



```

        talude[2]=(dipdir-70,90)
        talude[3]=(dipdir-110,90)
        atrito[0]=(0,90,at)

    #Salvando os arquivos
    write2=csv.writer(open(entrada+'_planarINST.txt' , 'w'),dialect='excel-
tab')
    write2.writerows(tabela_instavel)
    write3=csv.writer(open(entrada+'_planarEST.txt' , 'w'),dialect='excel-tab')
    write3.writerows(tabela_estavel)
    write4=csv.writer(open(entrada+'_taludeplanar.txt' , 'w'),dialect='excel-
tab')
    write4.writerows(talude)
    write5=csv.writer(open(entrada+'_atritoplanar.txt' , 'w'),dialect='excel-
tab')
    write5.writerows(atrito)

if __name__ == "__main__":

    from Tkinter import *
    import ttk
    from tkFileDialog import askopenfilename

    def calculate(*args):
        try:
            dipdir_ = float(dipdir.get())
            dip_ = float(dip.get())
            at_ = float(at.get())
            entrada = askopenfilename(filetypes=[('Arquivo de texto', '.txt')],
title='Abrir                                arquivo                                de
DESCONTINUIDADES...').encode(sys.getfilesystemencoding())
            if entrada:

                processar(entrada,dipdir_,dip_,at_)

        except ValueError:
            pass

    root = Tk()
    root.title("Análise de Estabilidade - Planar")

    mainframe = ttk.Frame(root, padding="3 3 12 12")
    mainframe.grid(column=0, row=0, sticky=(N, W, E, S))
    mainframe.columnconfigure(0, weight=1)
    mainframe.rowconfigure(0, weight=1)

    dipdir = StringVar()
    dip = StringVar()
    at = StringVar()
    #status = StringVar()

    dipdir_entry = ttk.Entry(mainframe, width=7, textvariable=dipdir)
    dipdir_entry.grid(column=2, row=1, sticky=(W, E))
    ttk.Label(mainframe, text="Rumo do Mergulho do Talude").grid(column=3,
row=1, sticky=W)

    dip_entry = ttk.Entry(mainframe, width=7, textvariable=dip)
    dip_entry.grid(column=2, row=2, sticky=(W, E))
    ttk.Label(mainframe, text="Mergulho do Talude").grid(column=3, row=2,
sticky=W)

    at_entry = ttk.Entry(mainframe, width=7, textvariable=at)
    at_entry.grid(column=2, row=3, sticky=(W, E))
    ttk.Label(mainframe, text="Angulo de Atrito").grid(column=3, row=3,
sticky=W)

    #ttk.Label(mainframe, textvariable=status).grid(column=2, row=4, sticky=(W,
E))

```



```
    ttk.Button(mainframe, text="Processar", command=calculate).grid(column=3,  
row=4, sticky=W)
```

```
    for child in mainframe.winfo_children(): child.grid_configure(padx=5,  
pady=5)
```

```
    dipdir_entry.focus()  
    root.bind('<Return>', calculate)
```

```
    root.mainloop()
```







```

# -*- coding: utf-8 -*-
import numpy as np
from numpy import arccos, sin, cos
import csv

def merg_ap(dipdir1,dip1,dipdir2):
#rumo do mergulho e mergulho para dois planos
#1 o real e 2 o perfil

tg=np.fabs(np.degrees(np.arctan(np.tan(np.radians(dip1))*np.cos(np.fabs(np.radians(dipdir2-dipdir1))))))
return tg

def risco_toppling_direto(data1,data2,dipdir,dip,at,inclina_lim):
#data1 sÃo descontinuidades
#data2 sÃo as intersecÃoes
tab1=np.zeros((data1.shape[0],3))
tab2=np.zeros((data2.shape[0],3))
c=0
for i in range(0,data1.shape[0]):
    tab1[i]=(data1[i,0],data1[i,1],0)
for i in range(0,data2.shape[0]):
    tab2[i]=(data2[i,0],data2[i,1],0)
for i in range(0,tab1.shape[0]):
    res= merg_ap(dipdir,dip,tab1[i,0])
    if
((arccos((sin(np.pi*tab1[i,0]/180.)*sin(dipdir*(np.pi/180.)) +
cos(tab1[i,0]*(np.pi/180.))*cos(dipdir*(np.pi/180.))))*180./np.pi)<20:
        tab1[i,2]=1
        c=c+1
    if c>0:
        for i in range(0,tab2.shape[0]):
            if tab2[i,0]>180:
                res= merg_ap(dipdir,dip,tab2[i,0]-180)
            else:
                res=merg_ap(dipdir,dip,tab2[i,0]+180)

            if tab2[i,0]>180:
                dif = (((arccos((((sin(np.pi*(tab2[i,0]-
180)/180.)*sin(dipdir*(np.pi/180.)))))) + cos((tab2[i,0]-
180)*(np.pi/180.))*cos(dipdir*(np.pi/180.))))*180./np.pi)
            else:
                dif =
(((arccos((((sin(np.pi*(tab2[i,0]+180)/180.)*sin(dipdir*(np.pi/180.)))))) +
cos((tab2[i,0]+180)*(np.pi/180.))*cos(dipdir*(np.pi/180.))))*180./np.pi)
            if tab2[i,1]>(90-dip):
                if dif<90:
                    if tab2[i,1]>inclina_lim:
                        if dif<20:
                            tab2[i,2]=2
                        else:
                            tab2[i,2]=3
                    else:
                        if dif<20:
                            tab2[i,2]=1

return tab2

def processar(entrada1,entrada2,dipdir,dip,at,inclina_lim):

data1 = np.loadtxt(entrada1)
data2 = np.loadtxt(entrada2)

tabela=risco_toppling_direto(data1,data2,dipdir,dip,at,inclina_lim)

#imprimindo a tabela com TODAS descontinuidades
write=csv.writer(open(entrada1+"_saidatotaltopdir.txt" ,
'w'),dialect='excel-tab')
write.writerows(tabela)

```



```

#separando em duas tabelas (instavel e estável):

c=d=e=0
for i in range (0,tabela.shape[0]):
    if tabela[i,2]==1:
        c=c+1
    if tabela[i,2]==2:
        d=d+1
    if tabela[i,2]==3:
        e=e+1

tabela_estavel=np.zeros(((tabela.shape[0]-(c+d+e)),2))
tabela_instavel1=np.zeros((c,2))
tabela_instavel2=np.zeros((d,2))
tabela_instavel3=np.zeros((e,2))

k=l=m=n=0
for j in range(0,tabela.shape[0]):
    if tabela[j,2]==1:
        tabela_instavel1[k]=(tabela[j,0],tabela[j,1])
        k=k+1
    if tabela[j,2]==2:
        tabela_instavel2[l]=(tabela[j,0],tabela[j,1])
        l=l+1
    if tabela[j,2]==3:
        tabela_instavel3[m]=(tabela[j,0],tabela[j,1])
        m=m+1
    if tabela[j,2]==0:
        tabela_estavel[n]=(tabela[j,0],tabela[j,1])
        n=n+1

#imprimindo os critérios limite
talude=np.zeros((4,2))
atruto=np.zeros((2,3))
talude[0]=(dipdir,dip)
if dipdir<180:
    talude[1]=(dipdir,90)
    talude[2]=(dipdir+70,90)
    talude[3]=(dipdir+110,90)
else:
    talude[1]=(dipdir,90)
    talude[2]=(dipdir-70,90)
    talude[3]=(dipdir-110,90)
atruto[0]=(0,90,dip)
atruto[1]=(0,90,90-inclina_lim)

write2=csv.writer(open(entradal+'_topDirINST1.txt' , 'w'),dialect='excel-tab')
write2.writerows(tabela_instavel1)
write3=csv.writer(open(entradal+'_topDieINST2.txt' , 'w'),dialect='excel-tab')
write3.writerows(tabela_instavel2)
write4=csv.writer(open(entradal+'_topDieINST3.txt' , 'w'),dialect='excel-tab')
write4.writerows(tabela_instavel3)
write5=csv.writer(open(entradal+'_topDieEST.txt' , 'w'),dialect='excel-tab')
write5.writerows(tabela_estavel)
write6=csv.writer(open(entradal+'_taludetopdir.txt' , 'w'),dialect='excel-tab')
write6.writerows(talude)
write7=csv.writer(open(entradal+'_atrutotopdir.txt' , 'w'),dialect='excel-tab')
write7.writerows(atruto)

if __name__ == "__main__":

    from Tkinter import *
    import ttk
    from tkFileDialog import askopenfilename

    def calculate(*args):
        try:
            dipdir_ = float(dipdir.get())

```



```

        dip_ = float(dip.get())
        at_ = float(at.get())
        inclina_lim = float(inclina_lim.get())
        entrada1 = askopenfilename(filetypes=[('Arquivo de texto',
'.txt')], title='Abrir arquivo de
DESCONTINUIDADES...').encode(sys.getfilesystemencoding())
        entrada2 = askopenfilename(filetypes=[('Arquivo de texto',
'.txt')], title='Abrir arquivo de
INTERSECCOES...').encode(sys.getfilesystemencoding())
        if entrada2:

            processar(entrada1, entrada2, dipdir_, dip_, at_, inclina_lim_)

    except ValueError:
        pass

root = Tk()
root.title("Análise de Estabilidade - Toppling Direto")

mainframe = ttk.Frame(root, padding="3 3 12 12")
mainframe.grid(column=0, row=0, sticky=(N, W, E, S))
mainframe.columnconfigure(0, weight=1)
mainframe.rowconfigure(0, weight=1)

dipdir = StringVar()
dip = StringVar()
at = StringVar()
inclina_lim = StringVar()
#status = StringVar()

dipdir_entry = ttk.Entry(mainframe, width=7, textvariable=dipdir)
dipdir_entry.grid(column=2, row=1, sticky=(W, E))
ttk.Label(mainframe, text="Rumo do Mergulho do Talude").grid(column=3,
row=1, sticky=W)

dip_entry = ttk.Entry(mainframe, width=7, textvariable=dip)
dip_entry.grid(column=2, row=2, sticky=(W, E))
ttk.Label(mainframe, text="Mergulho do Talude").grid(column=3, row=2,
sticky=W)

at_entry = ttk.Entry(mainframe, width=7, textvariable=at)
at_entry.grid(column=2, row=3, sticky=(W, E))
ttk.Label(mainframe, text="Angulo de Atrito").grid(column=3, row=3,
sticky=W)

inclina_lim_entry = ttk.Entry(mainframe, width=7,
textvariable=inclina_lim)
inclina_lim_entry.grid(column=2, row=4, sticky=(W, E))
ttk.Label(mainframe, text="Inclinação máxima").grid(column=3, row=4,
sticky=W)

#ttk.Label(mainframe, textvariable=status).grid(column=2, row=4,
sticky=(W, E))
ttk.Button(mainframe, text="Processar", command=calculate).grid(column=3,
row=5, sticky=W)

for child in mainframe.winfo_children(): child.grid_configure(padx=5,
pady=5)

dipdir_entry.focus()
root.bind('<Return>', calculate)

root.mainloop()

```



## ANEXO VIII – CÓDIGO FONTE DAS ROTINAS – TOMBAMENTO FLEXURAL



```

# -*- coding: utf-8 -*-
import numpy as np
from numpy import arccos, sin, cos
import csv

def merg_ap(dipdir1,dip1,dipdir2):
#rumo do mergulho e mergulho para dois planos
#1 o real e 2 o perfil

tg=np.fabs(np.degrees(np.arctan(np.tan(np.radians(dip1))*np.cos(np.fabs(np.radians(dipdir2-dipdir1))))))
    return tg

def risco_toppling_flexural(data,dipdir,dip,at):
    n=data.shape[0]
    tab=np.zeros((n,3))
    for i in range(0,n):
        tab[i]=(data[i,0],data[i,1],0)
    for i in range(0,n):
        res= merg_ap(dipdir,tab[i,1],tab[i,0])
        dif =
        ((arccos((((sin(np.pi*tab[i,0]/180.)*sin(dipdir*(np.pi/180.)))) +
        ((cos(tab[i,0]*(np.pi/180.))*cos(dipdir*(np.pi/180.)))))))*180./np.pi)
        if res>(at+90-dip):
            if 160<dif<200:
                tab[i,2]=1
    return tab

def processar(entrada,dipdir,dip,at):
    data = np.loadtxt(entrada)

    tabela=risco_toppling_flexural(data,dipdir,dip,at)

    #imprimindo a tabela com TODAS descontinuidades
    write=csv.writer(open(entrada+"_saidatotaltopflex.txt" , 'w'),dialect='excel-tab')
    write.writerows(tabela)

    #separando em duas tabelas (instavel e estável):
    c=0
    for i in range (0,tabela.shape[0]):
        if tabela[i,2]==1:
            c=c+1

    tabela_estavel=np.zeros(((tabela.shape[0]-c),2))
    tabela_instavel=np.zeros((c,2))
    a=0
    b=0
    for j in range(0,tabela.shape[0]):
        if tabela[j,2]==1:
            tabela_instavel[a]=(tabela[j,0],tabela[j,1])
            a=a+1
        else:
            tabela_estavel[b]=(tabela[j,0],tabela[j,1])
            b=b+1

    #imprimindo os critérios limite
    talude=np.zeros((4,2))
    talude[0]=(dipdir,dip)
    talude[3]=(dipdir,90-(at+90-dip))
    if dipdir<180:
        talude[1]=(dipdir+70,90)
        talude[2]=(dipdir+110,90)
    else:
        talude[1]=(dipdir-70,90)
        talude[2]=(dipdir-110,90)

    write2=csv.writer(open(entrada+'_topFlexINST.txt' , 'w'),dialect='excel-tab')
    write2.writerows(tabela_instavel)

```



```

write3=csv.writer(open(entrada+'_topFlexEST.txt' , 'w'),dialect='excel-tab')
write3.writerows(tabela_estavel)
write4=csv.writer(open(entrada+'_taludetopflex.txt' , 'w'),dialect='excel-tab')
write4.writerows(talude)

if __name__ == "__main__":

    from Tkinter import *
    import ttk
    from tkFileDialog import askopenfilename

    def calculate(*args):
        try:
            dipdir_ = float(dipdir.get())
            dip_ = float(dip.get())
            at_ = float(at.get())
            entrada = askopenfilename(filetypes=[('Arquivo de texto',
            '.txt')], title='Abrir arquivo de
DESCONTINUIDADES...').encode(sys.getfilesystemencoding())
            if entrada:

                processar(entrada,dipdir_,dip_,at_)

        except ValueError:
            pass

    root = Tk()
    root.title("Análise de Estabilidade - Toppling Flexural")

    mainframe = ttk.Frame(root, padding="3 3 12 12")
    mainframe.grid(column=0, row=0, sticky=(N, W, E, S))
    mainframe.columnconfigure(0, weight=1)
    mainframe.rowconfigure(0, weight=1)

    dipdir = StringVar()
    dip = StringVar()
    at = StringVar()
    #status = StringVar()

    dipdir_entry = ttk.Entry(mainframe, width=7, textvariable=dipdir)
    dipdir_entry.grid(column=2, row=1, sticky=(W, E))
    ttk.Label(mainframe, text="Rumo do Mergulho do Talude").grid(column=3,
row=1, sticky=W)

    dip_entry = ttk.Entry(mainframe, width=7, textvariable=dip)
    dip_entry.grid(column=2, row=2, sticky=(W, E))
    ttk.Label(mainframe, text="Mergulho do Talude").grid(column=3, row=2,
sticky=W)

    at_entry = ttk.Entry(mainframe, width=7, textvariable=at)
    at_entry.grid(column=2, row=3, sticky=(W, E))
    ttk.Label(mainframe, text="Angulo de Atrito").grid(column=3, row=3,
sticky=W)

    #ttk.Label(mainframe, textvariable=status).grid(column=2, row=4,
sticky=(W, E))
    ttk.Button(mainframe, text="Processar", command=calculate).grid(column=3,
row=4, sticky=W)

    for child in mainframe.winfo_children(): child.grid_configure(padx=5,
pady=5)

    dipdir_entry.focus()
    root.bind('<Return>', calculate)

    root.mainloop()

```



DOAÇÃO

IGC-USP

Data 25/02/15



