



**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
UNIDADE ARAXÁ
DEPARTAMENTO DE MINAS E CONSTRUÇÃO CIVIL
GRADUAÇÃO EM ENGENHARIA DE MINAS**

**CRIAÇÃO DE UMA SOLUÇÃO DE *SOFTWARE* PARA O CÁLCULO DO ÂNGULO
DE ABATIMENTO E DIMENSIONAMENTO DE VENTILAÇÃO DE MINA**

OVÍDIO FRANCCESCO GOMES DUARTE

ORIENTADOR

MICHEL MELO OLIVEIRA

CO-ORIENTADOR

LEANDRO RESENDE MATTIOLI

**ARAXÁ
2016**

D812c Duarte, Ovidio Francesco Gomes

Criação de uma solução de software para o cálculo do ângulo de abatimento e dimensionamento de ventilação de mina / Ovidio Francesco Gomes Duarte. – 2016.

83 f.: il.

Orientador: Prof. MSc. Michel Melo Oliveira
Trabalho de Conclusão de Curso (Graduação em Engenharia de Minas) – Centro Federal de Educação Tecnológica de Minas Gerais, 2016.

1. Lavra Subterrânea. 2. Programação. 3. Python. I. Oliveira, Michel Melo. II. Centro Federal de Educação Tecnológica de Minas Gerais. III. Título.

CDU: 622.261.2 + 004.42

OVÍDIO FRANCCESCO GOMES DUARTE

CRIAÇÃO DE UMA SOLUÇÃO DE *SOFTWARE* PARA O CÁLCULO DO ÂNGULO
DE ABATIMENTO E DIMENSIONAMENTO DE VENTILAÇÃO DE MINA

Trabalho de conclusão de curso apresentado ao Centro de Educação tecnológica de Minas Gerais, Campus IV Araxá, como requisito parcial para obtenção do título de Engenheiro de Minas.

Data de aprovação: ____/____/____

Banca Examinadora:

Prof. MSc. Michel Melo Oliveira - CEFET - MG
Orientador - Presidente da Banca Examinadora

Prof. MSc. Leandro Resende Mattioli - CEFET – MG
Co-orientador

Prof. Dr. Mauricio Antônio Carneiro – CEFET - MG

Prof. MSc. Felipe Valença de Oliveira - CEFET - MG

Dedico este trabalho a meus pais, Creusmar e Sebastião, meu irmão Samuel e a minha avó Jandira que sempre me apoiaram e acreditaram em mim. Dedico também a meus avôs, Osvaldo e Ovídio, que de maneira indireta contribuíram para essa importante conquista.

AGRADECIMENTOS

Agradeço primeiramente a Deus por estar comigo e me auxiliar em todos os momentos. Agradeço também a meus pais, Creusmar e Sebastião, por sempre terem me apoiado e incentivado a estudar, pelas inúmeras vezes em que se sacrificaram e fizeram de tudo para que eu pudesse ter uma boa educação, por me darem liberdade de escolha quanto ao curso escolhido, pelos conselhos dados, mas acima de tudo sou grato pelo exemplo que sempre me deram. Mãe eu admiro a sua resiliência e capacidade de superação, com você aprendi que posso superar qualquer obstáculo se tiver persistência e garra. Pai com o senhor eu aprendi a ter paciência e procurar ver sempre o melhor não importando o quanto uma situação possa parecer ruim.

Sou grato também a minha avó Jandira por tudo que ela fez por mim e pelo meu irmão seja de maneira direta ou através de minha mãe ou meu pai. Serei sempre grato não pelo fato de ter uma avó, mas uma segunda mãe que está sempre presente principalmente nos momentos em que errei, pois são esses os que eu mais preciso de ajuda. Agradeço a meu irmão Samuel por todo o apoio que ele sempre me deu, pelos sacrifícios que ele teve que fazer devido a minha ausência em casa e também por me ajudar em meu amadurecimento.

Agradeço a meu orientador, Michel, não só pelo conhecimento transmitido em sala de aula, mas também por me auxiliar na escolha do tema do TCC, por acreditar em mim, pelas dúvidas esclarecidas dentro e fora da sala de aula e pela paciência que sempre teve.

Agradeço a meu Co-orientador, Mattioli, por toda a ajuda neste trabalho, pelas dúvidas esclarecidas, pela paciência e por me apresentar ao python e me mostrar o quão interessante e desafiador pode ser a programação.

Sou grato também a todos os meus professores que me auxiliaram nesta jornada não apenas pelo conhecimento transmitido em sala, mas também por acreditarem em mim e terem paciência com muitas das minhas perguntas e dúvidas que depois percebi serem um pouco absurdas.

Sou grato também aos professores que a vida me deu, sejam eles os amigos que me ajudaram em diversos momentos, aos companheiros de república, as pessoas que me receberam em suas casas, aos amigos de sala de aula e também a galera do posto.

Tudo o que temos que decidir é o que
fazer com o tempo que nos é dado. J. R. R.
Tolkien

RESUMO

O aumento na produção de bens minerais e as inovações tecnológicas do século 20 fizeram com que os métodos de lavra subterrânea por abatimento serem amplamente empregados devido principalmente aos seus baixos custos e a sua capacidade de produção. Entretanto, associado a estes métodos estão os danos provocados pelo grande rebaixamento e deslocamento da superfície topográfica aonde é realizada a lavra, fenômeno esse chamado de subsidência. As regiões mais afetadas pela subsidência podem ser identificadas através da determinação de um parâmetro chamado ângulo de abatimento. Contudo a maioria dos métodos para a determinação do ângulo de abatimento envolve cálculos grandes ou complexos. Outra preocupação em minas subterrâneas é o dimensionamento e cálculo dos parâmetros de ventilação, que embora possam ser calculados de forma simples são operações repetitivas. Assim, foi criada uma solução de *software* na linguagem *python* para determinar o ângulo de abatimento e os parâmetros de ventilação. A escolha da linguagem *python* é devido a sua sintaxe simples e pelo fato dela ser uma linguagem orientada a objetos, o que facilita a manutenção do código.

Palavras chave: Ângulo de abatimento, Ventilação de mina, *Python*.

ABSTRACT

The increase in production of minerals commodities and the technological innovation in the 20th century increased the use of unsupported underground mining methods that began to be widely used because of their low costs and large production capacity. However, associated with these methods is the damage caused by the surface displacement at the place where mining occurs, phenomenon known as mine subsidence. The regions most affected by subsidence can be identified by the determination of a parameter called angle of break. However, most methods to determine the angle of break involve large or complex calculations. Another concern in underground mining is the determination and calculation of ventilation parameters, which are easy to calculate but is a repetitive task. Thinking in these issues, a software solution in python language was created to determine the angle of break and the ventilation parameters. The choice of using python is because it is a language of simple syntax and an object oriented programming language, which facilitates and improve the code maintenance.

Key Words: Angle of break, ventilation, Python.

Índice de Ilustrações

Figura 2.1 - Classificação dos métodos de lavra subterrânea. Adaptado de Brady & Brown (2004)...	17
Figura 2.2 - <i>Shield</i> utilizado para suporte do painel (direita) e <i>shearer</i> utilizado para extrair o minério. Adaptado de Bessinger (2011).	19
Figura 2.3 - Vista superior de uma lavra por longwall. Adaptado de Hartman & Mutmansky (2002)...	21
Figura 2.4 - Operações unitárias do sublevel caving. Adaptado de Hartman & Mutmansky (2002)...	23
Figura 2.5 - Subníveis e o padrão de furos de desmonte em sublevel caving. Adaptado de Hartman & Mutmansky (2002).	23
Figura 2.6 - Principais aspectos geométricos de uma lavra por Block caving. Adaptado de Brown (2002).	25
Figura 3.1 - Principais regiões de subsidência em uma mina de Block Caving. Adaptado de Vyazmensky (2008).	30
Figura 3.2 - Subsidência da mina de Northparkes. Adaptado de Vyazmensky (2008).	30
Figura 3.3 - Gráfico empírico relacionando o MRMR e o ângulo de abatimento. Adaptado de Vyazmensky (2008).	36
Figura 3.4 - Abatimento progressivo do hanging wall da mina de Grängesberg. Adaptado de Brady & Brown (2004) e Hoek (1974).	38
Figura 3.5 - Falhamento progressivo do hanging wall. Adaptado de Vyazmensky (2005) e Hoek (1974).	38
Figura 5.1 - Popularidade das linguagens de programação em 2015 (terceira coluna) e 2014 (quarta coluna).	47
Figura 6.1 - Diagrama de caso de uso do programa.	49
Figura 6.2 - Diagrama de classes conceitual.	51
Figura 6.3 - Arquitetura detalhada para a seleção de módulos.	52
Figura 6.4 - Arquitetura detalhada do grupo funcional Laubscher.	52
Figura 6.5 - Arquitetura detalhada do grupo funcional Numérico.	53
Figura 6.6 – Arquitetura detalhada do grupo funcional Ventilação.	53
Figura 6.7 - Entries com valores incorretos em seus campos de entrada.	55
Figura 6.8 - Um dos possíveis erros que podem acontecer no cálculo do ângulo de abatimento pelo método de equilíbrio limite.	56
Figura 6.9 - Classe laubscherform em execução e seus campos de entrada.	57
Figura 6.10 - Gráfico de Laubscher gerado pelo programa.	58
Figura 6.11 - Atributos fundo e gráfico ao serem invocados pelo método canvas.	59
Figura 6.12 - Interface criada ao selecionar calcular a perda de carga.	63
Figura 6.13 - Classe Ventilacao3 com seus principais atributos.	65
Figura 6.14 - Classe Ventilacao4 com seus principais atributos.	66
Figura 6.15 - Classe AbaVentilacao.	68
Figura 6.16 - Classe Janela exibindo a interface da classe AbaNumerico.	70

Lista de Quadros e Tabelas

Tabela 3.1 – Parâmetros e seus valores possíveis. Adaptado de Laubscher (1990).....	35
Tabela 6.1 – Valores mínimos, iniciais e finais dos campos do módulo <u>laubscherform</u>	57
Quadro 6.1 – Distribuição das classes em módulos.	69
Quadro 7.1 – Perfil dos testadores do protótipo <i>underpy</i>	72
Quadro 7.2 - Respostas das pessoas a avaliação a avaliação do <i>underpy</i>	72

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Apresentação	13
1.2	Justificativa	14
1.3	Objetivos	14
2	LAVRA SUBTERRÂNEA	15
2.1	Métodos de abatimento	18
2.1.1	Longwall	19
2.1.2	Sublevel Caving	21
2.1.3	Block Caving	24
3	ABATIMENTO	27
3.1	Classificação de maciços rochosos	31
3.1.1	Classificação de Bieniawski	31
3.1.2	MRMR	32
3.1.2.1	Intemperismo	33
3.1.2.2	Orientação das descontinuidades	34
3.1.2.3	Tensões induzidas	34
3.1.2.4	Efeitos de detonações	34
3.2	Cálculo do ângulo de abatimento	35
3.2.1	Métodos empíricos	35
3.2.2	Métodos de equilíbrio limite	37
4	VENTILAÇÃO DE MINA	41
5	PROGRAMAÇÃO ORIENTADA A OBJETOS	45
5.1	Classes	45
5.2	Métodos, Atributos e Encapsulamento	46
5.3	Herança e Polimorfismo	46
5.4	A linguagem Python	47
6	METODOLOGIA DE DESENVOLVIMENTO ORIENTADA A OBJETOS	49
6.1	Análise de Requisitos	49
6.2	Arquitetura do sistema	50
6.3	Projeto orientado a objetos	51
6.3.1	Frame e Entry	53
6.3.2	SmartEntry	54
6.3.3	AbaNumerico	55
6.3.4	LaubscherForm	56
6.3.5	LaubscherGrafico	58
6.3.6	AbaLaubscher	60
6.3.7	Ventilacao1	61
6.3.8	Ventilacao2	62
6.3.9	Ventilacao3	64
6.3.10	Ventilacao4	66
6.3.11	AbaVentilacao	67
6.3.12	Janela	68
6.4	Detalhes de implementação	68
6.5	Aspectos de usabilidade	69
7	RESULTADOS	71
8	CONCLUSÃO	75
9	REFERÊNCIAS BIBLIOGRÁFICAS	77

1 INTRODUÇÃO

1.1 Apresentação

Mineração é considerada a segunda mais antiga das conquistas humanas – considerando que a agricultura foi à primeira. As duas juntas são classificadas como atividades primárias ou básicas das primeiras civilizações. Pouco foi alterado na importância destas atividades desde o surgimento da civilização. Se considerarmos pesca e as atividades madeireiras como agricultura e a produção de óleo e gás como parte da mineração, então agricultura e mineração continuam a fornecer todos os recursos básicos usados pela sociedade moderna (Hartman & Mutmansky 2002, tradução nossa).

A mineração é composta de vários estágios que envolvem desde a busca por depósitos que contenham o bem mineral até a etapa de beneficiamento que é a etapa final. Na etapa de beneficiamento o bem mineral de interesse é separado do restante dos minerais e é vendido. Porém, antes de ser enviado para o beneficiamento o minério deve ser extraído da superfície da terra. Esta atividade de extração do bem mineral da superfície da terra recebe o nome de lavra ou exploração e se divide em dois grupos principais: Lavra a céu aberto e lavra subterrânea.

A lavra subterrânea embora muito antiga só começou a ser executada mais intensamente nos últimos dois séculos e mais especificamente a partir da segunda metade do século 20. O aumento de minas subterrâneas no mundo nesta época foi devido a melhoria dos equipamentos de escavação e no desenvolvimento dos computadores. Com as melhorias dos equipamentos, a vida útil e a durabilidade deles foram aumentadas e com o advento dos computadores, cálculos que demoravam dias passaram a ser realizados em minutos. Outra importância da utilização de programas está relacionada à segurança, já que atualmente existem programas que permitem que o operador controle o equipamento à distância, permitindo assim que o trabalhador realize sua função com conforto e segurança.

1.2 Justificativa

Em métodos de lavra por abatimento o correto cálculo do ângulo de abatimento reduz os custos de projeto, indica à equipe de planejamento qual é o melhor local das instalações de superfície e também fornece qual é a área que será afetada devido à extração do minério. Quanto ao dimensionamento dos parâmetros de ventilação, se eles forem insuficientes o ambiente se torna perigoso à ocupação humana, o que ocasiona um comprometimento da segurança, um aumento dos custos e uma possível paralisação da atividade de extração. Assim, a utilização de um programa para o cálculo do ângulo de abatimento e dos parâmetros de ventilação tem a vantagem de fornecer resultados rápidos e precisos. Além disso, permite que o usuário fique livre de tarefas repetitivas e que demandam muito tempo. Uma vez que a parte dos cálculos é realizada pelo computador, o profissional pode se ocupar com aspectos mais relevantes tais como o entendimento de como diferentes fatores podem influenciar no ângulo de abatimento e nos parâmetros de ventilação e como solucionar os diferentes problemas que surgem.

1.3 Objetivos

Este trabalho de conclusão de curso tem como objetivo verificar a viabilidade de construção e utilização de um programa computacional para o cálculo do ângulo de abatimento nos métodos de lavra do tipo *caving* e para o cálculo dos parâmetros de ventilação de minas subterrâneas. Os objetivos específicos deste trabalho são:

- Especificar de maneira formal o problema em questão através de técnicas de análise de requisitos;
- Escolher uma metodologia de desenvolvimento de *software* adequada;
- Prototipar o programa em uma linguagem de programação de alto nível;
- Verificar se o *software* pode ser utilizado como uma ferramenta auxiliar de ensino.

2 LAVRA SUBTERRÂNEA

Lavra subterrânea pode ser definida como a exploração de um corpo de minério que se encontra a uma determinada profundidade sem que ocorra a remoção do material que está acima do minério. Segundo Hartman & Mutmansky (2002) a lavra subterrânea pode ser empregada quando a profundidade da mina e/ou a quantidade de estéril a ser removida se tornam excessivas para a lavra a céu aberto.

Para extrair o mineral ou rocha de interesse que está abaixo da superfície é necessária a execução de várias atividades, porém Brown (2002) afirma que a exploração subterrânea de minerais se resume a basicamente três grupos de atividades:

1. Desenvolvimento de acesso físico a região mineralizada;
2. A extração do minério da rocha encaixante; e
3. O transporte do minério para as instalações de beneficiamento na superfície.

O primeiro grupo corresponde à etapa de criação das estruturas necessárias de uma mina para que a exploração possa acontecer, esta é a etapa de desenvolvimento da mina e nela são feitas as aberturas principais como *shafts*¹ e níveis principais (*main levels*). Já o segundo grupo corresponde à execução da lavra propriamente dita e o terceiro grupo ao carregamento e transporte do minério extraído para fora das estruturas da mina de forma que ele possa ser adequadamente beneficiado na superfície. Porém, durante a execução destas atividades são criadas diferentes escavações subterrâneas, pois são elas que permitirão acesso ao corpo de minério e acesso das instalações da superfície as instalações subterrâneas. Brown (2002) afirma que para cada grupo de atividades subterrâneas descritas acima existem escavações subterrâneas que devem ser criadas, podendo estas escavações ser:

¹ Para palavras ou termos que forem desconhecidos procurar sua definição no glossário.

1. Acessos permanentes e aberturas de serviços ou componentes de infraestrutura da mina;
2. Acessos ao realce e as aberturas de serviços ou desenvolvimento de realces;
3. Realces através do quais o corpo de minério é removido.

O primeiro conjunto de aberturas inclui rampas e poços (*shafts*) e está relacionado ao desenvolvimento de acesso físico a região mineralizada. O segundo inclui entradas principais (*Main entries*), *orepasses* e níveis de transporte enquanto que o terceiro grupo inclui os realces (*stopes*) propriamente ditos.

Segundo Nelson (2011) a lavra subterrânea é mais cara, menos segura e mais trabalhosa, porém como vantagem dela tem-se a reduzida quantidade de estéril que é extraída. Assim, embora os custos para a exploração subterrânea sejam elevados quase não há uma preocupação com a deposição de pilhas de estéril e muitas vezes o estéril que é retirado é utilizado para preencher escavações feitas em outras regiões da mina de forma a fornecer suporte as estruturas lá existentes e permitir a continuidade da extração do minério (Stephan 2011). Além disso, “a lavra subterrânea visualmente é menos impactante que a lavra a céu aberto e por esse motivo às preocupações com fatores ambientais e locais costuma ser menor” (Hartman & Mutmansky 2002, tradução nossa).

Com o decorrer do tempo a grande variabilidade dos depósitos minerais em sua forma, teor, profundidade, resistência mecânica e o aumento da profundidade destes depósitos impulsionaram o surgimento de diferentes métodos de lavra subterrânea. Dentre os vários métodos que podem ser utilizados Brady & Brown (2004) dividem os métodos subterrâneos como ilustrado na Figura 2.1 em três grupos: **suportes naturais**, **suportes artificiais** e **métodos com abatimento**.

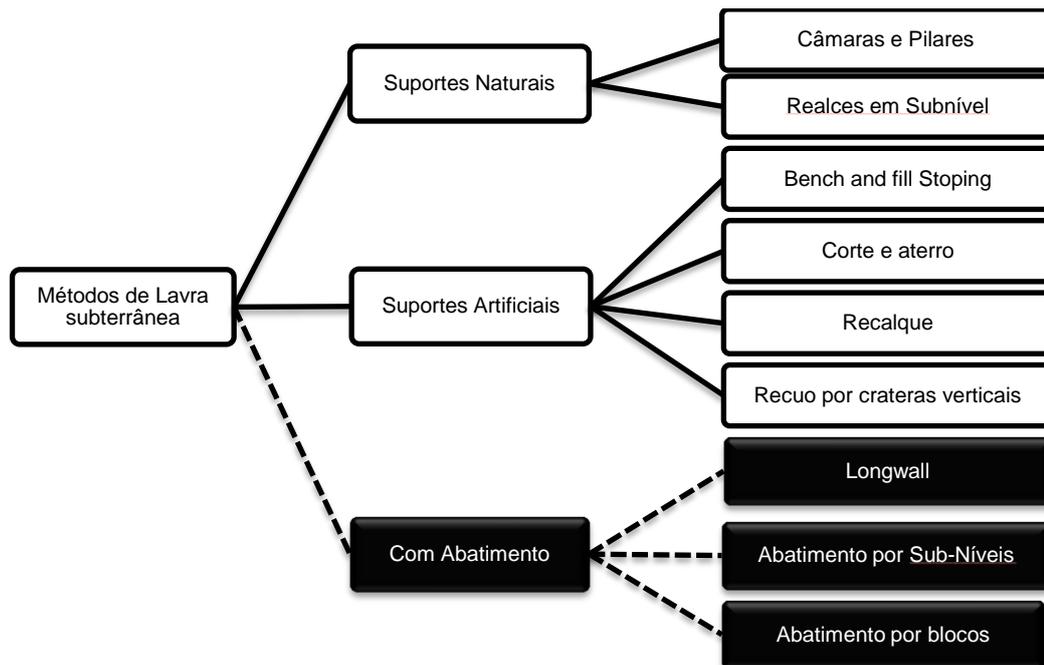


Figura 2.1 - Classificação dos métodos de lavra subterrânea. Adaptado de Brady & Brown (2004).

A diferença entre estes grupos consiste principalmente na forma como o realce é criado e na forma como ele será mantido ou até mesmo se não será mantido. Hartman & Mutmanský (2002) explicam que a diferença entre as três categorias apresentadas é que os métodos de suportes naturais ou métodos sem suporte consistem em métodos de lavra em que a rocha suporta o próprio peso e o peso sobrejacente a ela, já os métodos de suporte artificiais consistem em métodos que é necessário o enchimento das aberturas criadas ou até mesmo a utilização de estruturas artificiais para manter as escavações feitas seguras. Por último, os métodos de abatimento (*Caving Methods*) são formas de lavra em que as aberturas de exploração são propositalmente projetadas para colapsar. Dos nove métodos de lavra apresentados, somente os da terceira categoria, com abatimento, serão discutidos em detalhes, uma vez que as outras duas categorias fogem do escopo deste trabalho.

2.1 Métodos de abatimento

Métodos de abatimento são formas de lavra subterrânea em que “as aberturas de exploração são projetadas para colapsar, isto é, a queda do minério e da rocha que se encontra acima dele é intencional e é a essência do método” (Hartman & Mutmansky 2002, tradução nossa).

Porém apenas a criação das aberturas subterrâneas sem os suportes adequados não garante a ocorrência do abatimento, pois existem alguns requisitos e mecanismos envolvidos na queda do material. Brown (2002) afirma que o abatimento acontece como resultado de duas influências principais - a **gravidade** e as **tensões induzidas** no topo do *undercut* e que os mecanismos pelo qual o abatimento ocorrerá dependem das relações entre as tensões induzidas, da resistência do maciço rochoso e da resistência e geometria das descontinuidades presentes no maciço. De acordo com Brown (2002), para o abatimento ocorrer é necessário às descontinuidades sejam bem desenvolvidas e que apresentem ângulos de mergulho pequeno. Este autor afirma que a melhor condição para o abatimento acontecer é quando se tem dois conjuntos de descontinuidades, sendo um horizontal ou sub-horizontal e o outro vertical ou sub-vertical.

Assim como existem as condições básicas necessárias para que o abatimento aconteça também existem fatores que podem ser prejudiciais a ele e que devem ser observados, pois caso contrário podem inviabilizar a lavra. De acordo com Brown (2002), o abatimento não ocorrerá quando as tensões tangenciais compressivas induzidas no topo do *undercut* ou da escavação forem baixas, ou tenderem a ser tracionais. Assim, os blocos de rocha não se tornam livres para cair sob a influência da gravidade ou para deslizarem sobre descontinuidades inclinadas ou verticais que possam existir. Estas condições podem ocorrer quando as tensões horizontais *in situ* são baixas ou a presença de escavações anteriores reduzem as tensões ou as redistribuem para longe do bloco, ou do painel que está sendo lavrado. Assim, a preocupação dos métodos de abatimento se concentra mais em evitar as situações que impeçam ele de ocorrer.

Como ilustrado na Figura 2.1 os principais métodos de lavra sem suporte são: *Longwall*, *Sublevel Caving* (Abatimento por subníveis) e *Block Caving* (Abatimento por blocos) que serão discutidos em maiores detalhes.

2.1.1 Longwall

Segundo Hartman & Mutmansky (2002) *longwall* é um método de lavra utilizado em depósitos tabulares plano-horizontais, relativamente finos onde uma grande abertura é criada para extrair o minério. Embora ele também seja utilizado em corpos com pequenos ângulos de mergulho seu uso nestes casos é raro devido à complexidade do processo. Este método é aplicável tanto em minas metalíferas em ambientes de rochas duras, assim como na lavra de carvão em rochas brandas (Brady & Brown 2004).

A lavra do *longwall* acontece com o desenvolvimento de painéis, sendo que estes são mantidos abertos por um poderoso sistema de pesados suportes mecanizados que agindo em conjunto ficam com um formato semelhante a um arco ou guarda-chuva de proteção sobre o realce (Hartman & Mutmansky 2002). Uma ilustração do sistema utilizado para o suporte do teto e para a extração do minério é mostrada na Figura 2.2.

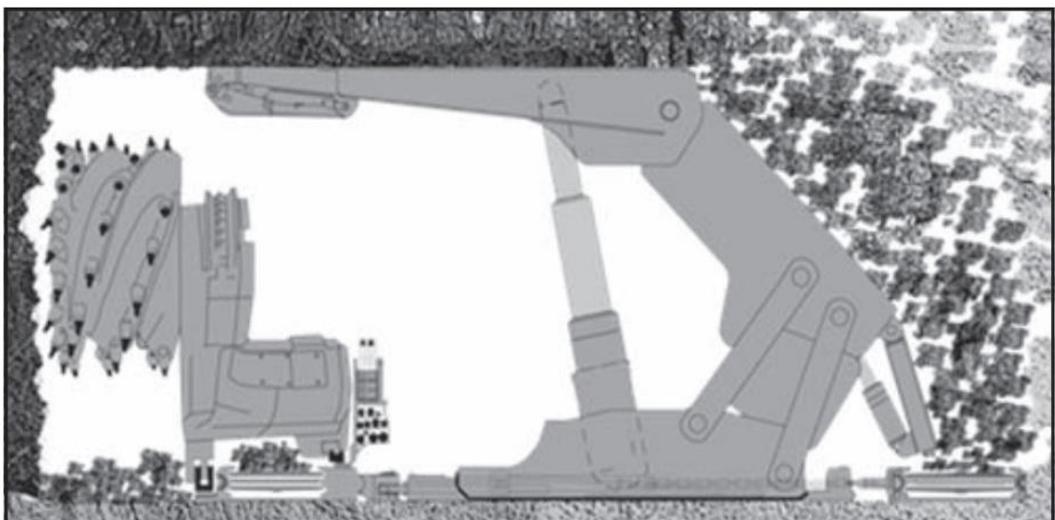


Figura 2.2 - *Shield* utilizado para suporte do painel (direita) e *shearer* utilizado para extrair o minério. Adaptado de Bessinger (2011).

A criação das aberturas do *longwall* se dá primeiramente com a criação de *declines* para servir de acesso da superfície até a camada mineralizada e também para a entrada e saída de ar. Em seguida é feito o restante das aberturas dentro das camadas de carvão (ou rocha). Dentre estas aberturas estão os *drifts* e *crosscuts* que são utilizados tanto para transporte quanto para ventilação dentro da mina.

A lavra acontece em partes, sendo que cada uma destas partes que é lavrada é chamada de painel e o avanço da lavra acontece paralela ao comprimento destes painéis. Segundo Bessinger (2011) o minério é removido da face do painel por um poderoso equipamento chamado *shearer* (Figura 2.2), que retira o minério da camada a uma altura de extração pré-definida e este após ser removido é encaminhado para equipamentos de transportes que se encontram nas laterais de cada painel. De acordo com Hartman & Mutmanský (2002), conforme parte da camada de minério é lavrada pelo *shearer*, o comprimento desta é reduzida e, com o objetivo de manter a abertura subterrânea estável e também proteger todos os equipamentos de extração, os *shields* avançam na mesma direção que o *shearer*. Os *shields*, ao se movimentarem, permitem que o material, que antes estava sendo suportado por eles, seja abatido, sendo esta região de material colapsado chamada de *gob*. Na Figura 2.3 podem ser observados cinco painéis, sendo que o segundo painel (de cima para baixo) é lavrado da esquerda para a direita estando o *gob* à esquerda. Nesta figura, todo o painel 1 já foi explotado, o 2 está sendo lavrado e os painéis 3, 4 e 5 ainda serão explotados. Ao sair dos painéis, o minério é então encaminhado para a superfície aonde será submetido a etapas de beneficiamento.

Diferente do *Sublevel Caving* e do *Block Caving*, neste método parte do minério é deixada no local aonde a lavra é realizada. Pelo fato de algum suporte ser utilizado na lavra, a deformação do *hanging wall* e do teto é apenas moderada.

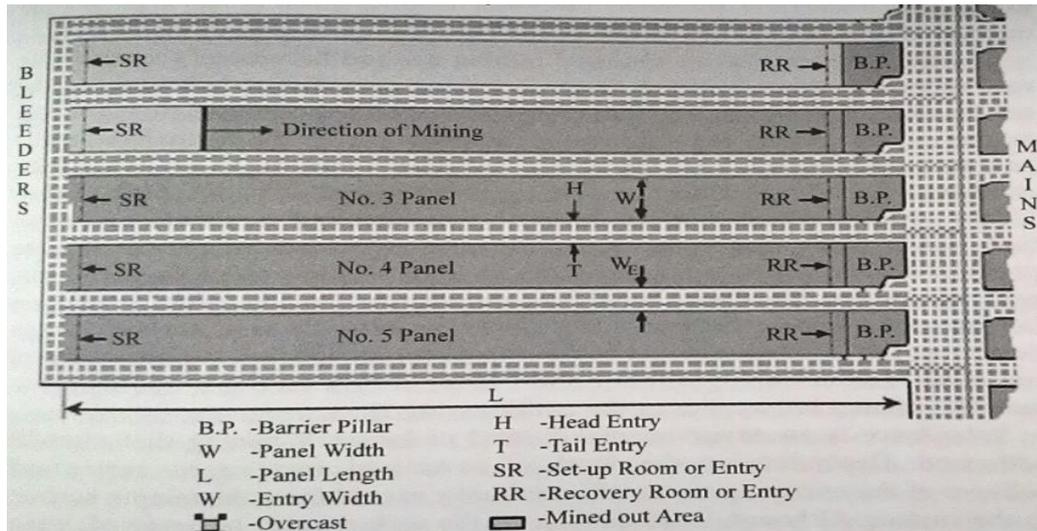


Figura 2.3 - Vista superior de uma lavra por longwall. Adaptado de Hartman & Mutmansky (2002).

2.1.2 Sublevel Caving

A lavra no *sublevel caving* acontece em níveis de maneira descendente sendo que o minério entre estes níveis é desmontado por explosivos. Conforme o minério é progressivamente desmontado e extraído, a rocha circundante ao minério (*hanging wall*) colapsa preenchendo o vazio criado pela extração do minério. Como apenas o estéril é abatido, o minério deve ser perfurado e desmontado da forma tradicional (Hartman & Mutmansky 2002).

Dunstan & Power (2011) afirmam que para a lavra por *sublevel caving* ser implementada com sucesso, o corpo de minério deve ter as seguintes características:

- Ser relativamente grande em sua área ou a rocha do seu *hanging wall* deve ser fraca para que o abatimento se inicie.
- Ter corpo vertical ou semi-vertical e uniforme;
- Apresentar distribuição de teor uniforme;
- O minério e o estéril devem ser visualmente diferentes;

- O corpo de minério deve ser competente de maneira a reduzir a necessidade de suporte das escavações durante o desenvolvimento da lavra.

No *sublevel caving* a maior parte do desenvolvimento da lavra é horizontal tanto para as aberturas usadas para o transporte quanto para aquelas utilizadas para exploração (*drifts, crosscuts*), embora algumas aberturas sejam inclinadas tais como rampas de acesso e *orepasses* (Hartman & Mutmansky 2002).

A Figura 2.4 mostra uma vista de perfil em que é possível ver a forma como a lavra acontece. No nível superior, ou nível 1, que corresponde ao primeiro *drift* ocorre à extração do minério e o subsequente abatimento do estéril enquanto que no segundo *drift* (nível 2) já foi realizado no realce os furos necessários para o desmonte e parte do minério já foi fragmentado sendo que o restante do minério já está pronto para ser detonado. No nível 3, estão sendo feitos os furos para a detonação do realce e o último nível da mina se encontra na fase de desenvolvimento, que consiste na abertura e criação do realce.

Na Figura 2.4 também está ilustrado a maneira como ocorre o transporte do minério. Este após ser extraído, é carregado até declines e estas ao receberem o minério o encaminham para o *orepass* até o minério chegar ao nível de transporte principal de onde o minério é encaminhado para a superfície.

É importante ressaltar que “o *drift* de um determinado nível é feito de maneira que ele se encontre entre dois drifts do nível superior e entre dois *drifts* do nível inferior” (Hartman & Mutmansky 2002, tradução nossa). Dessa forma, a execução de furos para detonação e a detonação propriamente dita não interferirão de forma significativa nos níveis superiores ou inferiores. A Figura 2.5 ilustra a forma como os *drifts* são construídos de maneira alternada.

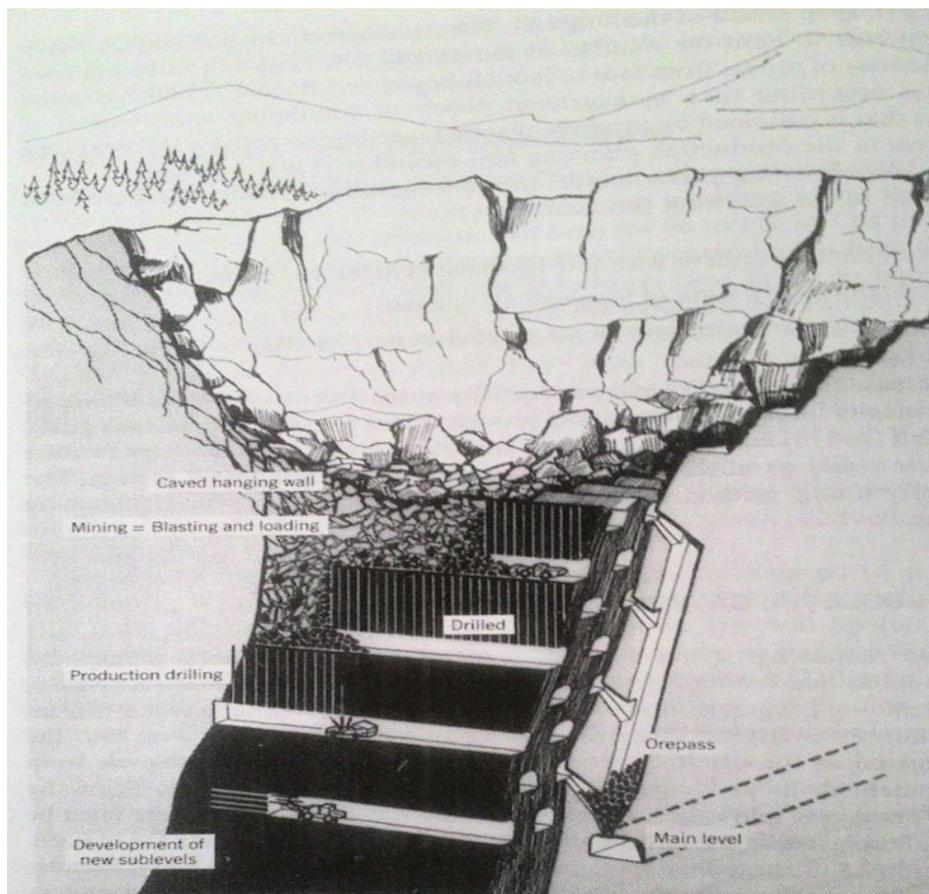


Figura 2.4 - Operações unitárias do sublevel caving. Adaptado de Hartman & Mutmanský (2002).

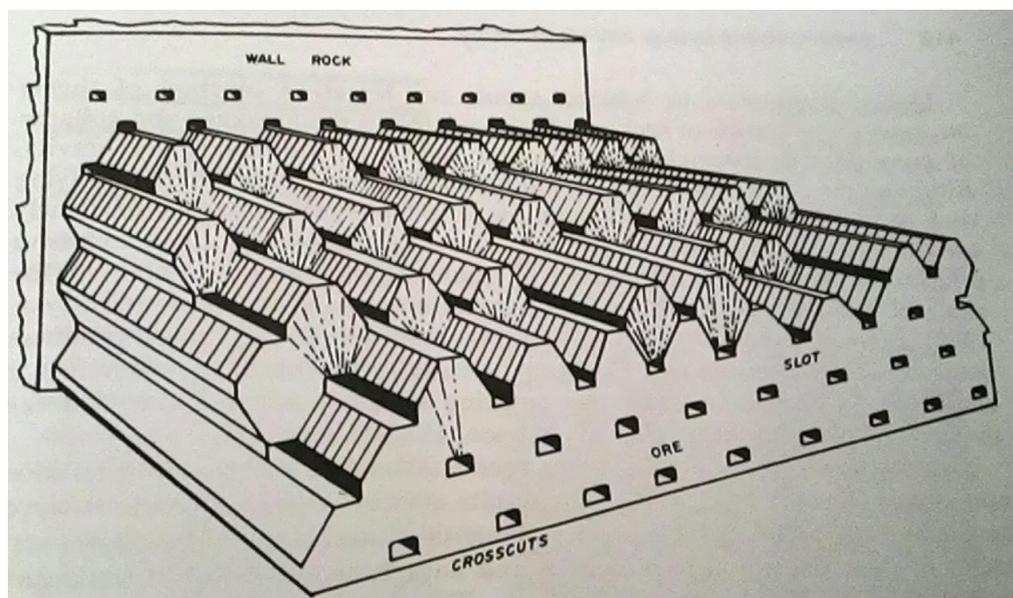


Figura 2.5 - Subníveis e o padrão de furos de desmonte em sublevel caving. Adaptado de Hartman & Mutmanský (2002).

Durante a exploração ocorre o abatimento da região próxima à mina e as regiões abatidas criam uma ligação entre a superfície e as estruturas subterrâneas existentes. Assim, “é fundamental a criação de restrições para os fluxos de água provenientes da superfície tanto para minas de *sublevel caving* que se localizam abaixo de minas do tipo céu aberto, quanto para minas localizadas no fundo de um vale. O fluxo de água para dentro da região abatida ocasiona não somente custos adicionais com bombeamento de água das frentes de trabalho, mas também cria uma enchente de lama nos pontos de trabalho” (Dunstan & Power 2011, tradução nossa).

2.1.3 Block Caving

Block caving é um método de lavra que usa a ação da gravidade para fraturar um bloco de minério não suportado permitindo que ele se abata e seja extraído através de *drawpoints* (Brannon *et al.* 2011). Segundo Brown (2002), para a lavra ser iniciada, é feito um *undercut* em todo o corpo de minério ou em apenas um bloco de minério para o abatimento iniciar. A região do *undercut* é perfurada e detonada progressivamente e alguns blocos de minério fragmentados neste processo são retirados para criar um vazio de maneira a possibilitar que o deslocamento e abatimento do material sobrejacente. Conforme o material é removido, o abatimento prossegue ascendentemente ao longo do corpo de minério. Quanto à maneira que a lavra é realizada, Brannon *et al.* (2011) destacam que embora alguns corpos de minério sejam lavrados apenas como um único bloco de produção, a maioria das minas usa uma das seguintes geometrias:

- Um sistema de *Block caving* maior que divide o depósito em blocos de produção menores;
- Uma única frente de lavra que avança através do corpo de minério, continuamente criando novas áreas de produção conforme as antigas áreas abatidas se exauriram.

A remoção do minério fragmentado é feita através dos níveis de produção que são desenvolvidos abaixo dos *undercuts*. A conexão entre *undercuts* e o nível de

produção é realizada por *drawbells* que permitem a passagem do minério por gravidade e também o direciona para os *drawpoints* (Brown 2002). Na Figura 2.6 está ilustrado de forma simplificada a geometria de uma lavra por *Block caving*, nela se observa o nível dos *undercuts* e o padrão de furos utilizados. Logo abaixo dos *undercuts* estão os *drawbells* que se localizam acima do nível de produção, abaixo deste nível têm-se os seguintes níveis: nível de ventilação, de fragmentação primária e de transporte.

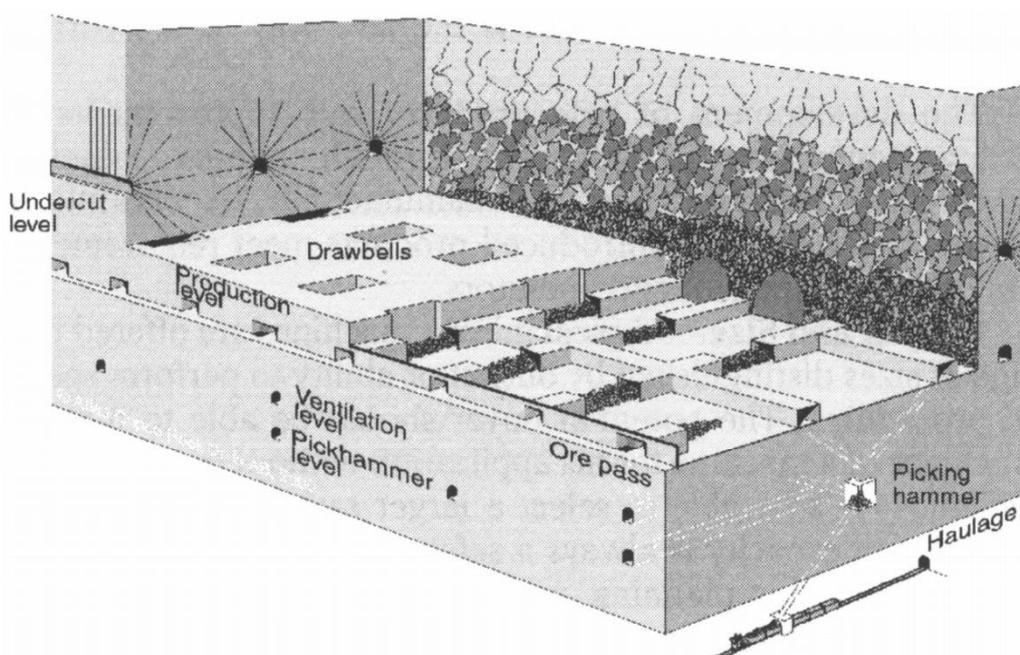


Figura 2.6 - Principais aspectos geométricos de uma lavra por Block caving. Adaptado de Brown (2002).

Embora o *Block Caving* rivalize com a lavra a céu aberto em custo e produção, este método de lavra é aplicável apenas em grandes corpos de minérios que sejam disseminados, uniformes, que apresentam baixa resistência mecânica (se fraturem facilmente), que apresentam uma forma regular e que sejam verticais ou sub-verticais (Hartman & Mutmansky 2002).

3 ABATIMENTO

Os métodos classificados como de abatimento ou subsidência se caracterizam como o próprio nome diz por apresentarem um rebaixamento da superfície topográfica durante a sua exploração. Antes de definir o que é subsidência de uma maneira mais específica convém esclarecer quais são os fenômenos envolvidos na sua ocorrência e qual é a causa dela.

Segundo Kratzsch (1983) a causa do abatimento é que as cavidades subterrâneas que são artificialmente criadas pela extração de minerais removem o suporte natural das camadas sobrejacentes e como resultado, sucessivas camadas de rocha sobre as aberturas criadas pela retirada de minério cedem sobre a influência da gravidade, até que finalmente o movimento atinge a superfície que é rebaixada até a cavidade subterrânea. Assim, as dimensões dos movimentos nas camadas superiores dependem do fechamento, em um período de tempo, das cavidades artificialmente criadas e, portanto, o tamanho destas cavidades está consequentemente relacionado ao deslocamento das camadas de rocha sobrejacentes e da superfície.

Então subsidência pode ser definida, de forma sucinta, como o deslocamento e deformação da superfície topográfica que está associada com o movimento do material sobrejacente que é causado devido ao colapso em uma escavação mineira (Harrison 2011). O rebaixamento da superfície está associado à remoção de material e à incapacidade do maciço rochoso de suportar o peso que as camadas superiores estão exercendo sobre ele.

É importante destacar que existem dois tipos de subsidência: a **contínua** e a **descontínua**. De acordo com Harrison (2011), na subsidência descontínua existe a formação de fraturas e cavidades na superfície, enquanto que na subsidência contínua a superfície se deforma de maneira moderada e gradual. Esta diferença também é feita por Brown (2002), segundo o qual a subsidência descontínua é caracterizada por grandes deslocamentos verticais e formação de descontinuidades sobre restritas áreas da superfície.

A subsidência provocada por escavações subterrâneas depende do método de lavra, da geometria do local de extração do minério e das propriedades da rocha acima da área de extração. Quanto aos fatores geológicos e operacionais que podem influenciar na subsidência, Harrison (2011) destaca os seguintes:

- **Espessura de Extração:** Quanto maior a espessura do material extraído mais provável que a subsidência superficial aconteça;
- **Profundidade:** Com exceção do *longwall* tanto o tempo de subsidência quanto a magnitude são dependentes da profundidade;
- **Taxa de material retirado:** Reduzindo a quantidade de material extraído a subsidência será reduzida;
- **Tensões *in situ*:** Elevadas tensões horizontais podem criar um arco no material que se encontra acima do vazio que foi causado pela lavra do minério, fazendo com que essa região fique menos propícia a se abater.
- **Descontinuidades geológicas:** A existência de falhas, dobras e estruturas similares podem aumentar o potencial para subsidência de forma tão intensa que em regiões em que as condições geológicas são adversas os efeitos dos outros parâmetros podem ser reduzidos.
- **Hidrogeologia:** A deformação da região lavrada pode alterar o gradiente hidráulico, causando tanto a inundação ou a drenagem das áreas superficiais e a formação ou drenagem de reservas subterrâneas. Rochas podem se enfraquecer devido a mudanças na saturação da água e em áreas carbonáticas, cavernas ou carstes podem se desenvolver com o tempo.
- **Tempo decorrido:** Em operações de *caving* a subsidência pode acontecer quase imediatamente embora a presença de camadas competentes possa aumentar o tempo para sua ocorrência.

Os deslocamentos verticais na superfície embora possam ser graduais, eles se alteram e mudam em aparência conforme se afastam do ponto acima de onde a escavação foi feita. Vyazmensky (2008) destaca três regiões distintas que estão relacionadas aos métodos de lavra do tipo *Caving* e mais especificamente ao *block caving*.

- Região abatida;
- Região de fraturamento de grande escala ou zona fraturada; e
- Região de subsidência contínua.

Cada uma destas três regiões apresenta diferentes feições e, conforme Vyazmensky (2008) destaca, a zona abatida corresponde à região que está acima da área da escavação e se manifesta na forma de uma cratera que é formada devido a contínua extração do minério. Ao longo da vida da mina, as deformações na superfície desta região vão gradualmente aumentando, podendo a subsidência total na zona abatida alcançar dezenas de metros. Já a segunda região, zona fraturada, é caracterizada por uma superfície fragmentada irregular com a presença de escarpas, grandes fraturas de tração, bancadas e grandes blocos submetidos a falhas cisalhantes rotacionais e colapsando na direção da cratera.

A terceira região, a de subsidência contínua, é caracterizada por uma suave depressão na superfície que também pode envolver um pequeno fraturamento. As deformações na superfície dentro desta região são relativamente pequenas quando comparadas com as outras duas regiões, embora elas não devam ser negligenciadas no planejamento de lavra (Vyazmensky 2008).

Na Figura 3.1 tem-se uma vista de perfil que ilustra as principais zonas de abatimento. Nesta imagem, a região central é a zona abatida, do lado exterior à zona abatida tem-se a zona fraturada e nas regiões extremas da figura a região de subsidência contínua. Na Figura 3.1 também pode ser observado o ângulo de abatimento (*angle of break*), ângulo de subsidência e ângulo de iniciação das fraturas do local assim como às fraturas de tração e a relação que elas possuem

com os blocos formados. Já a Figura 3.2 corresponde a um exemplo real de abatimento e subsidência na mina de *Northparks*, sendo que nesta figura também podem ser observadas as três regiões de abatimento descritas.

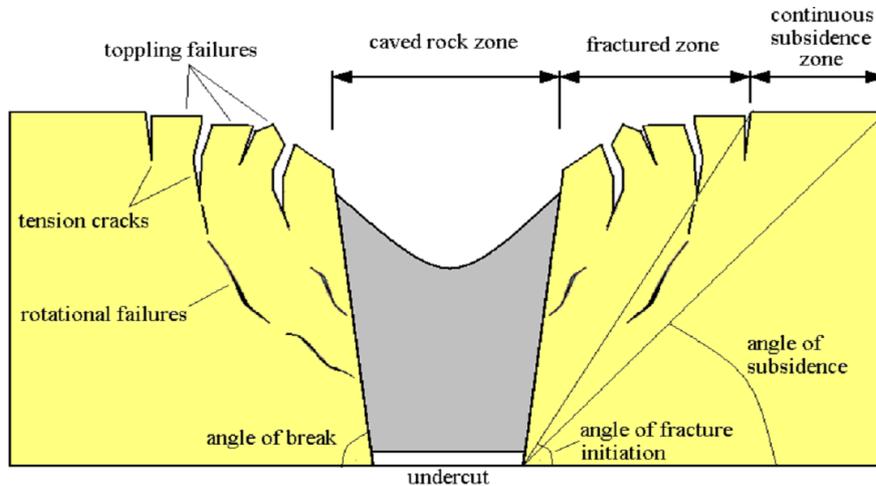


Figura 3.1 - Principais regiões de subsidência em uma mina de Block Caving. Adaptado de Vyazmensky (2008).

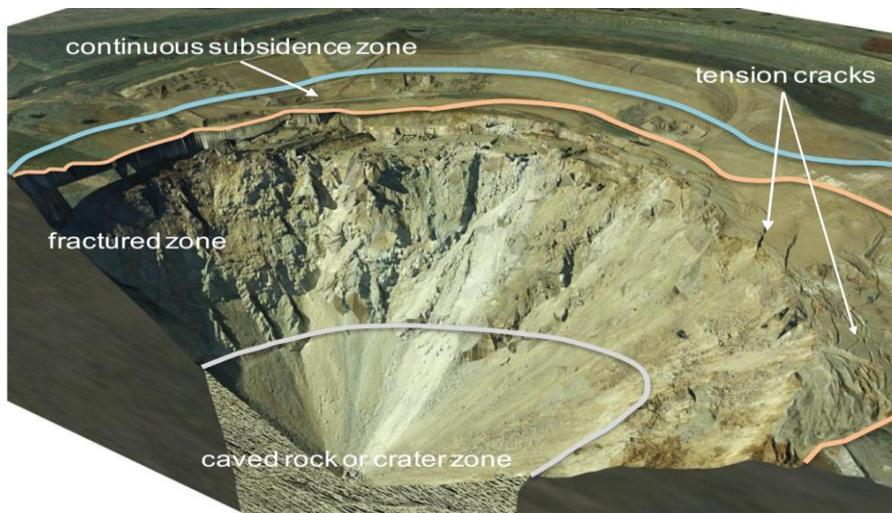


Figura 3.2 - Subsidência da mina de Northparks. Adaptado de Vyazmensky (2008).

Os termos ângulo de quebra (ou ângulo de abatimento), ângulo de iniciação de fratura e ângulo de subsidência apresentam algumas diferenças. Para Vyazmensky (2008) o ângulo de quebra (ou de abatimento) mede a extensão da

zona abatida ou o limite do material abatido e o ângulo de iniciação de fratura indica a extensão das maiores superfícies de fraturamento onde a superfície já está colapsada ou instável. Segundo este autor, o ângulo de subsidência define os limites das deformações na superfície sejam elas milimétricas ou até mesmo maiores. Os valores destes ângulos são obtidos ao se considerar o ângulo formado entre uma reta horizontal e uma reta que liga o fim da escavação até o ponto da superfície mais distante da região considerada. Assim, o ângulo de abatimento pode ser definido como uma reta que liga o fim da escavação até o ponto da região abatida que se encontra mais distante na superfície, então o ângulo entre esta linha criada e uma linha horizontal corresponde ao ângulo de abatimento (ver Figura 3.1).

De acordo com Vyazmensky (2008) para a obtenção do ângulo de abatimento existem dois diferentes métodos que são muito utilizados: os métodos empíricos e os métodos de equilíbrio limite. Estes métodos variam de autor para autor devido as diferentes considerações e simplificações que eles fazem, por isso neste trabalho será apresentado apenas um exemplo de cada um destes dois métodos.

3.1 Classificação de maciços rochosos

Rochas diferem da grande maioria de outros materiais usados na engenharia devido ao fato das rochas apresentarem fraturas de vários tipos que tornam a sua estrutura descontínua (Brady & Brown 2004). Assim, de forma a tentar compreender as propriedades dos maciços rochosos várias formas de classificação foram criadas.

3.1.1 Classificação de Bieniawski

O sistema geomecânico de classificação de maciços rochosos de Bieniawski ou RMR como ele é conhecido de acordo Brady & Brown (2004) foi criado no período de 1973 a 1976 por Richard Bieniawski von Preinl utilizando dados obtidos de escavações realizadas em rochas sedimentares em obras de construções civil feitas na África do Sul. O RMR usa cinco parâmetros de classificação:

- 1. Resistencia da rocha intacta:** Essa resistência é obtida através de ensaios uniaxiais que são ensaios em que um corpo de rocha cilíndrico

é comprimido de forma unidirecional até que este corpo se fracture e se quebre.

2. **Rock Quality Designation (RQD):** O RQD também é um sistema de classificação de maciço rochoso. Para o cálculo do seu valor é realizado a soma do comprimento de todos os fragmentos de rocha que são recuperados de um furo de sondagem que sejam maiores que 0.1 metros, o resultado desta soma então é multiplicado por 100 e depois dividido pelo comprimento total do furo.
3. **Espaçamento entre descontinuidades:** É a distância perpendicular média entre descontinuidades adjacentes. Este valor é frequentemente expresso como uma média para um determinado valor de área.
4. **Condição das Juntas:** Se refere à separação ou distância entre as paredes das descontinuidades, a continuidade ou persistência, a rugosidade da superfície e a condição das superfícies das paredes da descontinuidade.
5. **Condições das águas subterrâneas:** Indica a influência que a pressão de águas subterrâneas existente entre descontinuidades ou o fluxo delas exercem sobre a estabilidade das escavações subterrâneas.

Como Brady & Brown (2004) destacam o valor de RMR é obtido através de valores que estão associados a cada um destes cinco parâmetros. O valor do RMR é obtido ao se somar todos os valores de cada um destes cinco parâmetros individuais. É importante ressaltar que os valores do RMR podem ir de oito que é o valor mínimo até 100 que corresponde ao valor máximo.

3.1.2 MRMR

O MRMR é um sistema de classificação de maciços rochosos que foi derivado do RMR. Esse sistema foi introduzido em 1974 por Laubscher e os seus parâmetros e conceitos foram obtidos através do estudo de complexas situações mineiras.

Desde a sua criação este sistema sofreu várias modificações e melhorias de forma que ele tem sido usado com sucesso em minas no Canadá, Chile, Filipinas, Sri Lanka, África do Sul, Estados Unidos e Zimbábwe (Laubscher 1990).

O sistema MRMR corresponde ao sistema RMR com alguns ajustes na forma de percentual e segundo Laubscher (1990), o valor de MRMR é o resultado do valor do RMR multiplicado por fatores de correção, sendo que estes fatores são empíricos e baseados em várias observações de campo. A equação 1 ilustra como é obtido o valor de MRMR.

$$MRMR = RMR \times ajuste_1 \times ajuste_2 \times ajuste_3 \times ajuste_4 \quad (1)$$

Os quatro fatores de ajuste são o intemperismo, a orientação das descontinuidades, as tensões induzidas na escavação e os efeitos de detonações na escavação.

3.1.2.1 *Intemperismo*

Segundo Laubscher (1990) certos tipos de rochas podem ser intemperizadas facilmente e devem ser levadas em consideração na hora de se definir o tamanho da escavação. Ainda de acordo com este autor, o intemperismo é dependente do tempo e influencia tanto no momento da utilização de suportes assim como na taxa de avanço de lavra em uma escavação.

Os três parâmetros afetados pelo intemperismo são a resistência da rocha intacta, o RQD e a condição das juntas. Com um aumento do intemperismo sobre um maciço rochoso tanto o valor do RQD quanto o valor da resistência da rocha intacta são reduzidos significativamente. As juntas são alteradas pelo intemperismo da rocha que as contém ou devido à alteração do material que as preenchem (Laubscher 1990).

3.1.2.2 *Orientação das descontinuidades*

De acordo com Laubscher (1990) o tamanho, a forma e a orientação de uma escavação afetam o comportamento de um maciço rochoso. A magnitude do ajuste que deve ser feito devido à orientação das descontinuidades depende do mergulho das juntas em relação ao eixo vertical de um bloco. A instabilidade do maciço depende do número de blocos que serão formados e estes dependem do número de juntas que apresentam um mergulho em uma direção que se afasta do eixo vertical. O cálculo do ajuste relativo à orientação das descontinuidades leva em conta o produto de dois fatores. O primeiro relacionado à orientação das juntas e o segundo é um fator de correção que considera a direção de avanço da escavação.

3.1.2.3 *Tensões induzidas*

Tensão induzida é o resultado da redistribuição do campo de tensões regionais que é causada devido à realização de uma escavação com uma determinada orientação e geometria (Brady & Brown 2004).

Laubscher (1990) afirma que neste ajuste são levados em conta apenas as tensões máximas, as mínimas e a diferença entre elas e segundo este autor, a diferença entre as tensões principais máximas e mínimas tem um impacto significativo nas juntas do maciço. O efeito desta diferença de tensões é mais sentido conforme a densidade do número de juntas aumenta ou as condições das juntas sejam reduzidas. Os fatores de ajuste pra tensões induzidas podem estar entre 0.6 a 1.2.

3.1.2.4 *Efeitos de detonações*

Detonações devem ser consideradas ao se calcular o MRMR porque elas criam novas fraturas e enfraquecem o maciço rochoso ao provocar o deslocamento das juntas e de blocos.

Levando em conta todos os fatores discutidos anteriormente, Laubscher chegou a valores de ajuste para os quatro itens mencionados acima. A Tabela 3.1

apresenta os limites mínimos e máximos para os ajustes propostos por Laubscher para o valor do MRMR.

Tabela 3.1 - Parâmetros e seus valores possíveis. Adaptado de Laubscher (1990).

Parâmetro	Ajustes Possíveis (%)
Intemperismo	30-100
Orientação	63-100
Tensões induzidas	60-120
Detonação	80-100

3.2 Cálculo do ângulo de abatimento

3.2.1 Métodos empíricos

Vyazmensky (2008) afirma que métodos empíricos são tradicionalmente usados em engenharia e estão baseados na unificação de observações passadas, comumente em situações similares, para descrever as tendências das respostas experimentais que estão associadas com o fenômeno estudado. Estimativas empíricas da subsidência em *Block Caving* incluem guias práticos (“Rules of Thumb”) e gráficos projetados experimentalmente que relacionam o ângulo de quebra, classificação de maciço rochoso e outros parâmetros.

O método empírico mais usado para estimar os valores dos ângulos de abatimento é o proposto por Laubscher. Laubscher propôs um gráfico que relaciona o ângulo de quebra ao MRMR, a densidade da rocha abatida, altura do material abatido e a geometria da mina (Vyazmensky 2008). O gráfico criado por Laubscher pode ser visto na Figura 3.3.

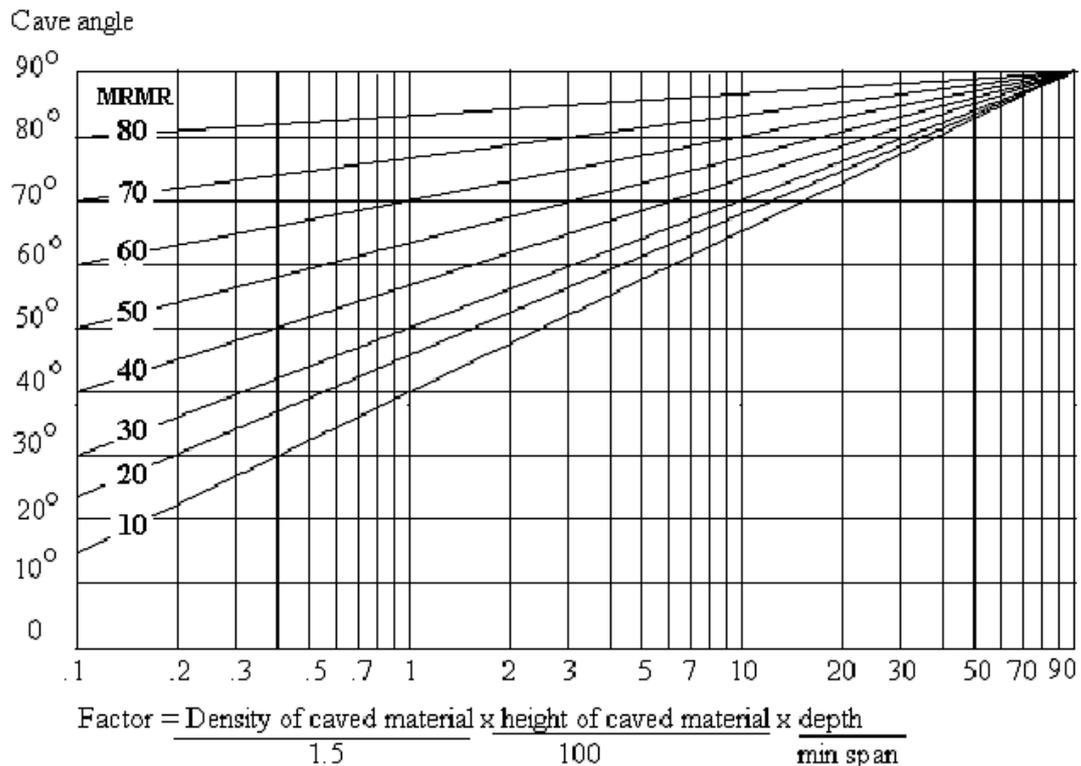


Figura 3.3 - Gráfico empírico relacionando o MRMR e o ângulo de abatimento. Adaptado de Vyazmensky (2008).

A Figura 3.3 apresenta no eixo das abcissas um valor para um fator empírico que neste trabalho será chamado de fator ou fator de Laubscher. O valor deste fator vai de 0.1 até 100 e ele é obtido pela equação 2. Como esta equação ilustra, a obtenção do valor do fator depende tanto de fatores físicos quanto de fatores geométricos.

$$Fator = \frac{Dm}{1.5} \times \frac{Hm}{100} \times \frac{p}{v} \quad (2)$$

Sendo:

Dm = Densidade do material abatido;

Hm = Altura do material abatido;

p = Profundidade da mina;

v = Vão livre da escavação da mina

Os critérios levados em conta são a densidade do minério, pois conforme foi dito o abatimento está relacionado à gravidade e, minérios que contenham densidades maiores sofrerão um efeito maior da gravidade. Os outros critérios são a altura do material, a profundidade da escavação e a largura da escavação ou vão livre. O vão livre nesse caso se refere à largura da escavação da mina ou a largura do *undercut* da mina.

Embora seja amplamente usado o método de Laubscher não leva em conta o efeito de estruturas geológicas como falhas que podem influenciar no ângulo de abatimento ou não considera a dificuldade para se calcular com precisão a densidade do material abatido. Assim, o gráfico de Laubscher é uma útil ferramenta para estimativas preliminares do ângulo de quebra, embora seja muito vago apoiar-se somente na geometria (Vyazmensky 2008).

3.2.2 Métodos de equilíbrio limite

Quando o corpo de minério não é muito grande e apresenta um ângulo de mergulho relativamente alto, o abatimento apenas do *hanging wall* precisa ser considerado. Em tais casos, o abatimento progressivo do *hanging wall* pode resultar conforme a lavra progride na direção do mergulho do corpo com a utilização de métodos como o *sublevel caving* (Brady & Brown 2004). O primeiro modelo de lavra por abatimento do *hanging wall* foi desenvolvido por Hoek em 1974 para a análise da subsidência da mina de Grängesberg na Suécia. O modelo proposto se baseava em um mecanismo de queda do *hanging wall* enquanto a lavra ocorria de forma descendente (Vyazmensky 2005). O abatimento da mina de Grängesberg pode ser visto na Figura 3.4 e o modelo conceitual proposto por Hoek pode ser observado na Figura 3.5.

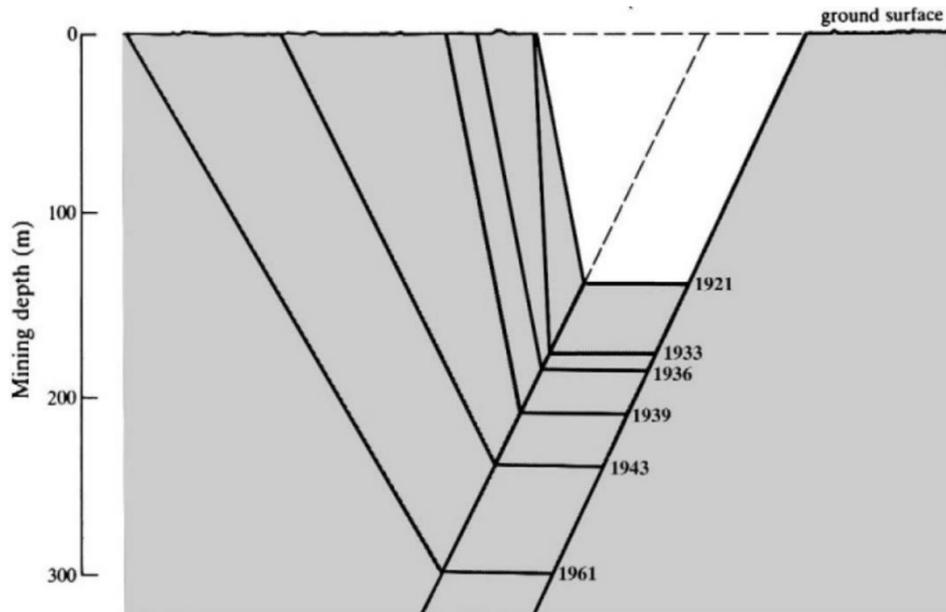


Figura 3.4 - Abatimento progressivo do hanging wall da mina de Grängesberg. Adaptado de Brady & Brown (2004) e Hoek (1974).

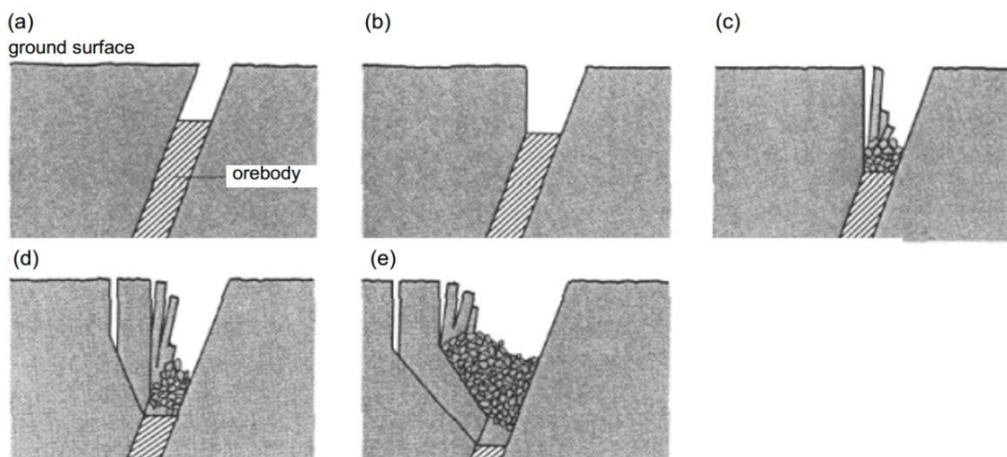


Figura 3.5 - Falhamento progressivo do hanging wall. Adaptado de Vyazmensky (2005) e Hoek (1974).

A subsidência causada pelo abatimento do *hanging wall* está associada a minas em que parte do corpo de minério está próxima a superfície e já foi extraído por métodos de lavra a céu aberto. Alternativamente, operações de lavra subterrânea podem se iniciar em regiões relativamente fracas, previamente não lavradas, de pouca profundidade e progredir em etapas de maneira descendente (Brown 2002). Segundo Brown (2002) e Brady & Brown (2004) o abatimento ocorre porque a cada nova etapa da lavra, uma fratura tracional e um plano de falha se

formam no *hanging wall* do maciço rochoso em um local crítico que é determinado pela resistência do maciço rochoso e pelas tensões induzidas. Em alguns casos, outros mecanismos podem causar o abatimento, como a existência de descontinuidades como falhas que fornecem planos de cisalhamento.

De acordo com Brown (2002), análise de equilíbrio limite desenvolvida por Hoek em 1974 foi criada para prever o progresso, a partir de uma posição inicial conhecida, da falha do *hanging wall* conforme a profundidade da lavra é aumentada e conforme o *hanging wall* colapsa, conseqüentemente a região da superfície acima dele será abatida. Brady & Brown (2004) e Vyazmensky (2008) citam uma equação que foi desenvolvida por Brown e Ferguson em 1979 a partir da análise de Hoek. Esta equação para o cálculo do ângulo de abatimento considera a inclinação da superfície topográfica e a pressão exercida pela água nas fraturas tracionais e no plano de cisalhamento (se houver a presença de água). A equação mencionada está representada na equação 3.

$$\tan \Psi_b = \tan \Psi_{p2} + \frac{(Z_2 \times \sin(\Psi_{p2} - \alpha))}{(\cos \Psi_{p2}) \times \left[\left(\frac{H_2 \times \sin(\alpha + \Psi_0)}{\sin(\Psi_0)} \right) - (Z_2 \times \cos(\alpha)) \right]} \quad (3)$$

Sendo:

Ψ_b = Ângulo de abatimento.

Ψ_{p2} = Inclinação do novo plano de falha

Z_2 = Nova profundidade da fenda de tração

α = Mergulho da superfície topográfica

H_2 = Profundidade do nível que está sendo lavrado

Ψ_0 = Mergulho do corpo de minério

Segundo Brady & Brown (2004) algumas as considerações feitas ao se desenvolver a equação 3 foram:

- Quando comparada com a seção transversal (Figura 3.4) a lavra e o abatimento ocorrem por uma grande distância ao longo da direção que é paralela ao corpo de minério;
- A posição inicial do corpo de minério, do *hanging wall* e do plano de falha anterior são conhecidas;
- A dimensão do abatimento a uma nova profundidade, H2, é definida pela fratura de tensão que forma a uma profundidade crítica e que possui a mesma direção que o corpo de minério;
- A falha do *hanging wall* do maciço rochoso ocorre ao longo de uma superfície crítica, planar, cisalhante cuja localização é definida pela resistência do maciço rochoso e pelas tensões efetivas impostas a ele;
- A posição inicial da face do *hanging wall* é definida por valores conhecidos dos parâmetros **H1** (profundidade do nível lavrado anterior), **Hc** (altura do material abatido), **Z1** (valor anterior da profundidade da fenda de tração) e **Ψ_{p1}** (inclinação do antigo plano de falha);
- A rocha que constitui o *hanging wall* apresenta propriedades mecânicas homogêneas e isotrópicas e a tensão cisalhante pode ser definida pelo método de tensões efetivas de Coulomb;

Nos cálculos utilizados pelo método de equilíbrio limite, uma simplificação da distribuição do campo de tensões da rocha que foi abatida e que está sendo abatida foi usada.

4 VENTILAÇÃO DE MINA

A ventilação é frequentemente descrita como o pulmão de uma mina, sendo as aberturas de ventilação as artérias que levam oxigênio para as frentes de trabalho e as veias de retorno que retiram os poluentes de maneira que eles sejam expelidos na atmosfera. Sem um sistema de ventilação eficiente, nenhum ambiente subterrâneo que requer pessoas para trabalhar pode operar de maneira segura (Mcpherson 1993).

Segundo Hartman & Mutmanský (2002) ventilação é a operação auxiliar mais importante em lavra subterrânea, pois ela não só mantém a qualidade atmosférica do ambiente, mas também é o pilar fundamental do sistema dos programas de saúde e segurança do trabalhador dentro da mina. Estes autores definem ventilação de mina como sendo: “O processo de resfriamento responsável pelo controle da quantidade de ar, seu movimento e sua distribuição”. Como dito, a ventilação tem grande importância para o sucesso da lavra subterrânea e é por esse motivo ela é quase tão antiga quanto à mineração subterrânea. Mcpherson (1993) afirma que em 600 a.C. os gregos já estavam cientes da necessidade da criação de circuitos de ventilação conectados. Este autor destaca também os métodos ilustrados na obra de Agricola, *De Re Metallica*, que ilustrava sistemas de ventiladores movidos pela força humana ou tração animal, porém a obra de maior influência foi a desenvolvida por John Job Atkinson em 1854 de nome ***On the theory of the ventilation of mines***.

Os conceitos desenvolvidos por Atkinson são à base da teoria da ventilação de mina utilizados nos dias de hoje e são baseados em estudos e teorias da mecânica dos fluidos e da termodinâmica, mais especificamente a equação de Bernoulli e a equação de Chezy-Darcy (Tuck 2011). Mcpherson (1993) afirma que de maneira a quantificar as relações que governam o comportamento dos fluxos de ar das minas subterrâneas, Atkinson desenvolveu uma equação que relaciona a geometria da escavação com parâmetros físicos do ar. A equação de Atkinson pode ser observada na equação 4.

$$p = f \times L \times \frac{per}{A} \times \rho \times \frac{u^2}{2} \quad (4)$$

Sendo:

p = Queda de pressão (Pa)

f = Coeficiente de fricção (adimensional)

L = Comprimento da escavação (m)

per = Perímetro da seção transversal da escavação (m)

A = Área da secção transversal da escavação (m²)

ρ = Densidade do ar (kg/m³)

u = Velocidade de deslocamento do ar (m/s)

Tuck (2011) afirma que a equação de Atkinson se baseia em algumas simplificações que nem sempre acontecem. Entre elas está a incompressibilidade do ar e a consideração de que o fator f e a densidade do ar são constantes. Como Mcpherson (2011) destaca, Atkinson agrupou todos os termos que ele considerava constante e originou uma nova constante, chamada de coeficiente de Atkinson. A simplificação de Atkinson está representada na equação 5 e com esta nova constante a equação 4 foi reescrita dando origem a equação 6.

$$k = \frac{f \times \rho}{2} \quad (5)$$

$$p = k \times L \times \frac{per}{A} \times u^2 \quad (6)$$

A equação 6 é chamada equação de Atkinson e nela pode se observar que o coeficiente de Atkinson é função da densidade, diferentemente do fator f que é adimensional. Assim, as unidades da constante k são dadas com as mesmas unidades que a densidade (Mcpherson 1993).

Tuck (2011) afirma que a equação 6 pode ser modificada ao se reescrever a vazão como o produto da velocidade e da área, como exibido na equação 7. Assim, a equação 6 se altera e fica da maneira como está exibida na equação 8.

$$Q = u \times A \quad (7)$$

$$p = k \times L \times \frac{per}{A^3} \times Q^2 \quad (8)$$

Para qualquer escavação de ventilação o comprimento, L, o perímetro, per, e a seção transversal, A, são conhecidas e se for ignorada a variação da densidade, o fator de fricção depende apenas da rugosidade das paredes da escavação. Assim, todas estas variáveis podem ser simplificadas em apenas uma variável, R, sendo R conhecida como a resistência da via de ventilação (Mcpherson 1993). A equação 9 representa a forma como a resistência da via de ventilação é calculada e a equação 10 corresponde a forma como a perda de pressão pode ser calculada quando se tem apenas a vazão de ar e a resistência da via de ventilação.

$$R = k \times L \times \frac{per}{A^3} \quad (9)$$

$$p = R \times Q^2 \quad (10)$$

A equação 10 é conhecida como a lei quadrada de ventilação de mina e é provavelmente a equação mais usada em ventilação subterrânea (Mcpherson 1993). Embora as equações 4, 6, 8 e 10 sejam a mesma, seu uso depende da quantidade de informações disponíveis e estas equações também fornecem uma maior versatilidade para o cálculo das variáveis que estão relacionadas à ventilação de mina.

5 PROGRAMAÇÃO ORIENTADA A OBJETOS

Em programação, entende-se por objeto “uma entidade que tem um estado, um comportamento e uma identidade definida, podendo esta entidade ser modificada, criada, destruída e compartilhada” (Grady *et al.* 2007, tradução nossa). Os objetos podem ser entidades tangíveis, como números, letras, listas de nomes, figuras e janelas, ou abstratas, como requisições HTTP e histórico de comandos. O paradigma de programação orientada a objetos é aquele no qual os programas são organizados como uma cooperação de objetos, cada um representando uma instância de uma classe. O programador é responsável por moldar o mundo dos objetos, ou variáveis de um programa e definir como eles devem interagir entre si para alcançar o comportamento do todo.

Dentre as principais vantagens desta abordagem, pode-se citar a alta capacidade de evolução dos programas, uma vez que novos objetos e novos comportamentos podem ser facilmente incorporados, até certo ponto. Além disso, um problema complexo quando modelado por objetos, é subdividido em problemas menores, em diferentes níveis de abstração, viabilizando o desenvolvimento.

Para melhor entender o que é programação orientada a objetos convém definir alguns conceitos chave.

5.1 Classes

De acordo com Grady *et al.* (2007), sendo um objeto uma entidade concreta que existe no espaço e no tempo, uma classe representa somente uma abstração, a essência de um objeto. Assim, uma classe pode ser definida como um conjunto de objetos que compartilham uma estrutura e um comportamento em comum. Os botões *OK* e *Cancelar* de um diálogo de confirmação, por exemplo, são instâncias de uma classe que generaliza a estrutura e o comportamento de todos os botões. As classes são associadas entre si por meio de uma hierarquia e unidas com relacionamentos de herança. Uma classe *Veículo*, por exemplo, pode ser vista como uma generalização de uma classe *Carro*.

5.2 Métodos, Atributos e Encapsulamento

Atributos são as variáveis que estarão dentro de cada um dos objetos da classe e podem ser de qualquer tipo (David 2011). Assim, os atributos são os valores armazenados e manipulados pelo objeto que definem seu estado. São exemplos de atributos o texto de um botão, a cor de um veículo e o estado de operação de uma máquina (ligada ou desligada).

Métodos são comportamentos do objeto, sendo similares às funções por receberem parâmetros e, eventualmente, retornarem um valor. Por meio dos métodos os objetos manipulam e fazem operações com seus atributos.

Normalmente, um objeto não concede acesso para leitura e escrita de seus atributos senão por meio de uma política bem definida, o que evita a presença de estados inválidos e reduz a quantidade de *bugs* (erros) gerados na fase de integração. Assim sendo, o atributo **saldo** de um objeto de uma classe *ContaCorrente* só pode ser alterado mediante os métodos **depositar**, **sacar** e **transferir**, e a implementação destes garante a integridade do objeto, evitando, por exemplo, que uma operação de saque seja realizada quando não há saldo suficiente. Esta importante característica deste paradigma é chamada de encapsulamento.

5.3 Herança e Polimorfismo

De forma simples, a herança pode ser definida como “uma relação entre classes onde uma classe compartilha a estrutura e/ou o comportamento que foi definido com outras classes” (Grady *et al.* 2007, tradução nossa). Uma classe que tem seus atributos e métodos herdados por outra é chamada de **superclasse**, enquanto a classe que herda os atributos e métodos é chamada de **subclasse**. A herança em programação funciona de forma similar à herança biológica, com a diferença que o programador define o que será ou não herdado.

Retomando o exemplo das classes *Carro* e *Veículo*, um atributo *velocidade* e um método *deslocar* poderiam ser descritos na classe *Veículo* e, por herança,

incorporados à classe *Carro*. No entanto, a implementação da classe *Carro* poderia definir novos métodos, como por exemplo, a operação **trocarPneu**, ou ainda restringir o comportamento de alguns métodos da superclasse, por exemplo ignorando ou eliminando a coordenada z do método **deslocar**.

5.4 A linguagem Python

Python atualmente é uma das linguagens de programação mais utilizadas, juntamente com linguagens como Java, C, C#, PHP e a recém-criada *Ruby* (Figura 5.1).

Language Rank	Types	Spectrum Ranking	Spectrum Ranking
1. Java	🌐 📱 🖥️	100.0	100.0
2. C	📱 🖥️ 🖨️	99.9	99.3
3. C++	📱 🖥️ 🖨️	99.4	95.5
4. Python	🌐 🖥️	96.5	93.5
5. C#	🌐 📱 🖥️	91.3	92.4
6. R	🖥️	84.8	84.8
7. PHP	🌐	84.5	84.5
8. JavaScript	🌐 📱	83.0	78.9
9. Ruby	🌐 🖥️	76.2	74.3
10. Matlab	🖥️	72.4	72.8

Figura 5.1 - Popularidade das linguagens de programação em 2015 (terceira coluna) e 2014 (quarta coluna)².

Python é usada tanto por cientistas escrevendo aplicações para os supercomputadores mais rápidos do mundo quanto por crianças aprendendo a programar (Rossum 2009).

² Fonte: <http://spectrum.ieee.org/computing/software/the-2015-top-ten-programming-languages>

Criada por Guido van Rossum no início da década de 90, a linguagem *Python* se tornou extremamente popular com o tempo. Rossum (2009) afirma, ainda, que o desenvolvimento se iniciou no Instituto de Pesquisas de Amsterdam (CWI³) e que sua decisão de criar uma nova linguagem de programação foi devido à dificuldade em encontrar uma linguagem de alto nível⁴ que atendesse às suas necessidades para um projeto que ele estava desenvolvendo na época em que ele estava no CWI.

Summerfield (2013) destaca que *Python* é uma linguagem multiplataforma, ou seja, o código por ela gerado pode ser executado tanto no *Windows* como em sistemas derivados do Unix, como Linux, BSD e Mac OS X, simplesmente por meio da cópia dos arquivos que compõem o programa.

Além disso, destacam-se as seguintes características:

- Módulos e pacotes para a organização e o acesso ao código;
- Programação multiparadigma, incluindo programação orientada a objetos e programação funcional;
- Exceções, um sofisticado mecanismo para a gestão de erros, no qual a responsabilidade pelo tratamento de um erro pode ser delegada a qualquer membro da cadeia de elementos envolvidos.

³ Acrônimo holandês para Centro de Matemática e Ciência da Computação.

⁴ Linguagem de alto nível são aquelas mais próximas da linguagem humana e, por conseguinte, mais fáceis de serem entendidas pelo programador.

6 METODOLOGIA DE DESENVOLVIMENTO ORIENTADA A OBJETOS

Como este trabalho trata do desenvolvimento de um software, foi adotada uma metodologia concebida particularmente para este fim, norteadas por etapas que vão desde a captura das funcionalidades necessárias até os testes finais de validação. As próximas seções descrevem brevemente cada um desses passos. A linguagem gráfica usada para a modelagem foi a UML (*Unified Modeling Language*⁵) versão 2.

6.1 Análise de Requisitos

Primeiramente é importante destacar como o programa será usado e a forma como o usuário vai interagir com ele. Na Figura 6.1 está ilustrado um diagrama de casos de uso do programa, que corresponde a uma representação gráfica das principais funções do programa, de quem interage com ele e o que resulta dessa interação.

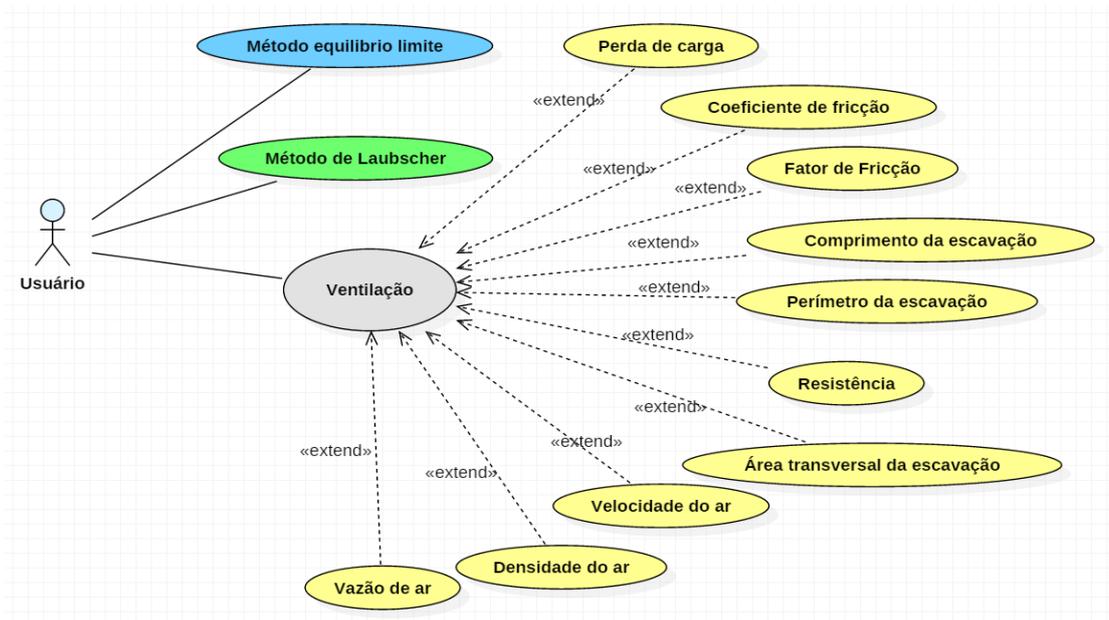


Figura 6.1 - Diagrama de caso de uso do programa.

⁵ UML é uma linguagem gráfica padrão para visualização, especificação e construção de projetos de software. A UML é utilizada com o objetivo de otimizar a etapa de desenvolvimento e assegurar uma maior qualidade do software.

Como ilustrado na Figura 6.1 o programa é constituído de três partes principais: Método de Equilíbrio Limite, Método de Laubscher e Ventilação. Assim, ao usar o programa o usuário deve escolher entre uma destas três opções de uso. A primeira e a segunda calculam o ângulo de abatimento pelo método de equilíbrio limite e pelo método numérico de Laubscher, respectivamente. Já a terceira parte do programa, calcula os parâmetros de ventilação expostos na Figura 6.1, sendo eles os seguintes: perda de carga, coeficiente de fricção, fator de fricção, comprimento da escavação, resistência da via de ventilação, área transversal, velocidade do ar, densidade do ar e a vazão de ar. É importante ressaltar que apenas um destes parâmetros pode ser calculado por vez. Além disso, os valores calculados não são salvos permanentemente, de modo que se outra opção for selecionada, tanto os dados de entrada como os resultados obtidos são perdidos.

6.2 Arquitetura do sistema

Uma vez enumerada às funcionalidades desejadas do sistema, uma arquitetura preliminar é definida para o protótipo. A modelagem foi feita com o uso do paradigma orientado a objetos e os principais componentes são apresentados no diagrama de classes da Figura 6.2.

A classe *Entry* é a base para os campos de entrada de valores numéricos e um *Frame* é um contêiner genérico para agrupar outros elementos de interface gráfica. A classe *LaubscherForm* é responsável pela entrada dos valores para o cálculo do fator de Laubscher. Já a classe *LaubscherGrafico* criará o gráfico que apresenta os resultados obtidos a partir dos dados digitados no formulário *LaubscherForm*. A obtenção do ângulo de abatimento propriamente dito é de responsabilidade da classe *AbaLaubscher*, que obtém os valores do formulário, realiza as operações necessárias e então configura os parâmetros de saída para a plotagem no gráfico. Em outras palavras, a classe *AbaLaubscher* realiza a integração entre *LaubscherForm* e *LaubscherGrafico*.

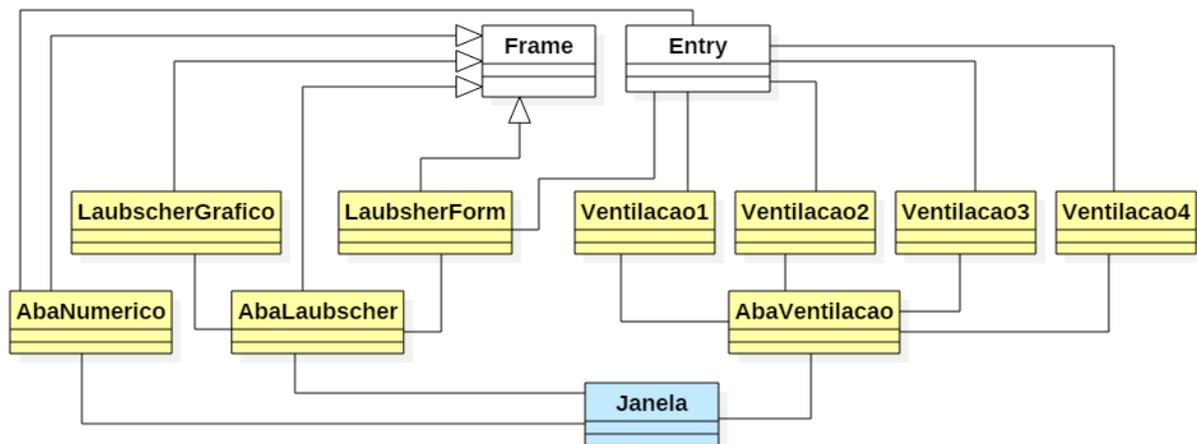


Figura 6.2 - Diagrama de classes conceitual.

O cálculo do ângulo de abatimento através do método de equilíbrio limite é feito pelo componente *AbaNumerico*. As classes *Ventilacao1*, *Ventilacao2*, *Ventilacao3* e *Ventilacao4* são responsáveis por calcular qualquer um dos parâmetros de ventilação que estão presentes nas equações 4, 6, 8 e 10, respectivamente. Já a classe *AbaVentilacao*, lida com a exibição dos campos de entrada de acordo com a equação que o usuário escolher. Assim, se o usuário escolher a equação 4, uma instância da classe *Ventilacao1* é criada na classe *AbaVentilacao*; se a equação 6 for a escolhida, o objeto criado e exibido será uma instância da classe *Ventilacao2*, e assim por diante.

Finalmente, a classe *Janela* representa a janela principal do programa, contendo instâncias das classes *AbaNumerico*, *AbaLaubscher* e *AbaVentilacao* e escolhendo uma delas para ser exibida de acordo com a seleção feita pelo usuário.

6.3 Projeto orientado a objetos

Após a definição de uma arquitetura preliminar, as classes são refinadas, ganhando um nível de detalhes mais próximo do necessário para a implementação. Nesse sentido, a fase de projeto constitui na definição dos atributos e métodos de cada classe. Os diagramas de classe das figuras 6.3, 6.4, 6.5 e 6.6 ilustram a arquitetura final do sistema. As próximas seções descrevem detalhadamente cada classe.

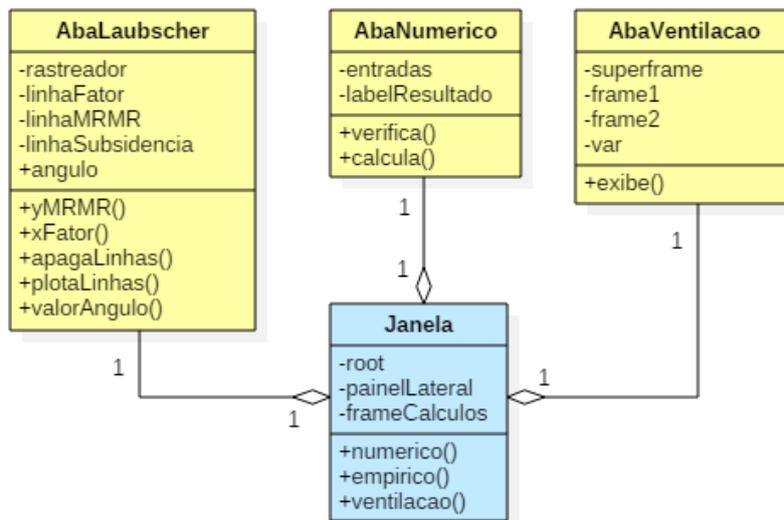


Figura 6.3 - Arquitetura detalhada para a seleção de módulos.

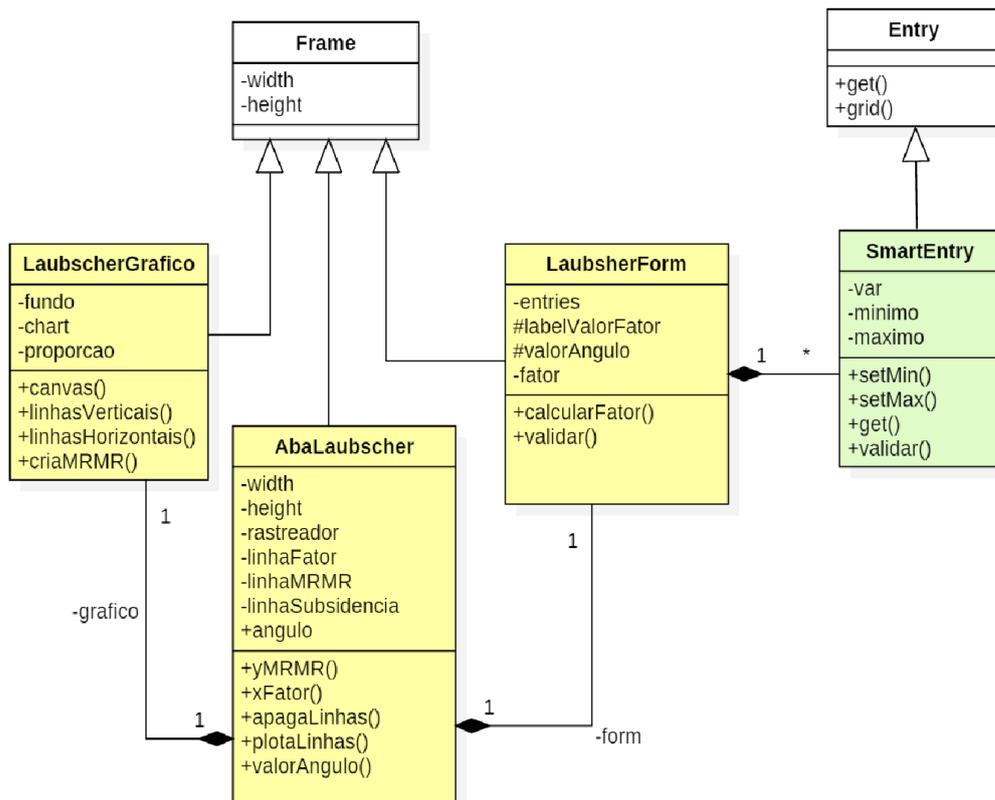


Figura 6.4 - Arquitetura detalhada do grupo funcional Laubscher.

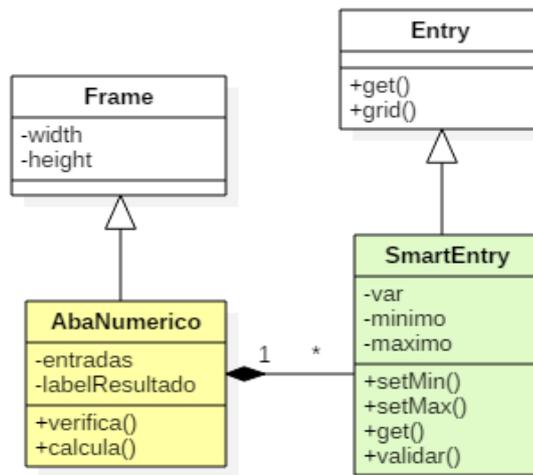


Figura 6.5 - Arquitetura detalhada do grupo funcional Numérico.

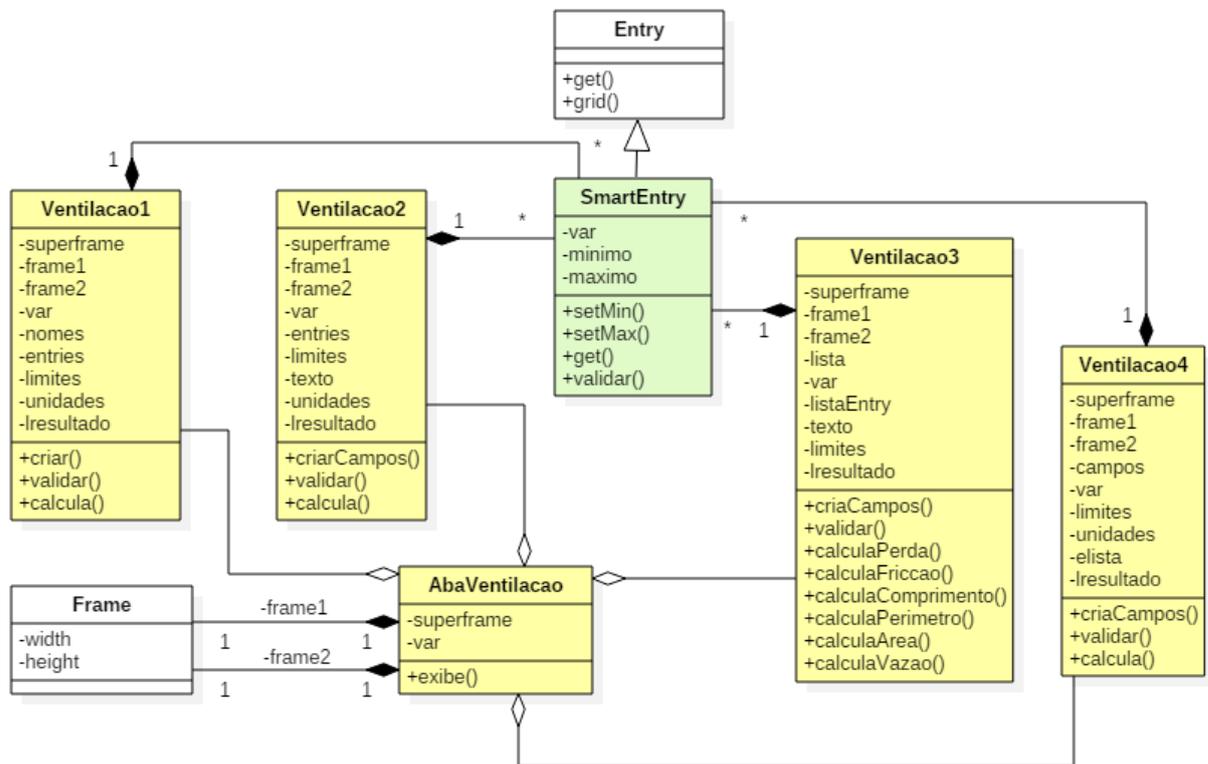


Figura 6.6 – Arquitetura detalhada do grupo funcional Ventilação.

6.3.1 Frame e Entry

As classes *Frame* e *Entry* fazem parte do *toolkit* para construção de interfaces gráficas do módulo **tkinter**, que é instalado por padrão junto com o *Python*.

6.3.2 SmartEntry

A classe *SmartEntry* é largamente utilizada em todos os módulos, tendo sido concebida para ampliar a funcionalidade da classe *Entry* permitindo a entrada de campos numéricos com validação de erros instantânea. Ela contém os atributos **var**, **minimo** e **maximo** e os métodos **setMin**, **setMax**, **get** e **validar**.

Os atributos **minimo** e **maximo** correspondem ao menor e maior valor que podem ser digitados na *Entry* criada. O atributo **var** corresponde ao valor numérico presente na *Entry*.

Os métodos **setMin** e **setMax** servem apenas para definir os atributos **minimo** e **maximo** como os menores e maiores valores aceitados na *Entry*. O método **get** armazena o valor que está no campo de entrada (*Entry*) e analisa se este valor é um caractere não numérico, um campo vazio ou um número menor ou maior que os atributos **minimo** e **maximo**. Se o que estiver no campo de entrada for um número menor ou igual ao atributo **maximo** e maior ou igual ao atributo **minimo** o método **get** devolve este número. Se o que estiver no campo de entrada for uma letra, um campo vazio ou um número fora dos limites permitidos o método **get** retorna um valor que indica para a classe *SmartEntry* que o campo de entrada contém um caractere inválido. Já o método **validar**, apenas modifica o fundo dos campos de entrada de acordo com os valores que são retornados pelo método **get**. Como observado na Figura 6.7 os quatro primeiros campos estão com o fundo em vermelho por conterem letras, estarem vazios ou possuírem valores que estão acima ou abaixo do que é aceito, este comportamento das cinco *entries* criadas só acontece porque os métodos **get** e **validar** trabalham juntos.

Densidade Material Abatido:	3FG
Altura Material abatido:	
Profundidade:	'%\$ _
Largura Mínima:	2500000000
Valor do MRMR:	50

Figura 6.7 - Entries com valores incorretos em seus campos de entrada.

Os critérios adotados como máximos e mínimos em todas as *entries* utilizadas nas diferentes classes do programa foram adotados considerando os valores encontrados na bibliografia ou utilizados em projetos de minas reais⁶.

6.3.3 AbaNumerico

Esta classe realiza o cálculo do ângulo de abatimento pelo método de equilíbrio limite de acordo com a equação 3. O atributo **entrada** é uma lista que armazena e posiciona as *entries* criadas pela classe *AbaNumerico*. Já o atributo **labelResultado** é apenas uma *Label* que tem como função armazenar o resultado dos cálculos.

O atributo **verifica** analisa se qualquer uma das *entries* contém um caractere inválido ou um número acima ou abaixo do permitido e caso isto ocorra é emitida uma mensagem informando ao usuário que os valores digitados estão incorretos. O método **calcula** tem como função fazer uma segunda verificação para evitar que os seguintes erros ocorram:

1. A nova profundidade da fenda de tração ser maior que a profundidade do nível lavrado atual.
2. O denominador da equação 3 ser igual à zero.

⁶ Os valores utilizados como máximo e mínimo estão presentes na bibliográfica, mais especificamente nos livros *Subsurface ventilation and Environmental Engineering* e no livro *SME Mining Engineering Handbook*.

Caso um destes dois casos ocorram, uma mensagem aparece na tela indicando qual dos dois casos aconteceu e pedindo ao usuário para corrigir os campos de entrada. Se nenhum destes erros ocorrerem o método **calcula** realiza os cálculos para a obtenção do ângulo de abatimento de acordo com a equação 3. Na Figura 6.8 está ilustrada uma das mensagens que aparece quando ocorre um erro no cálculo do ângulo de abatimento.

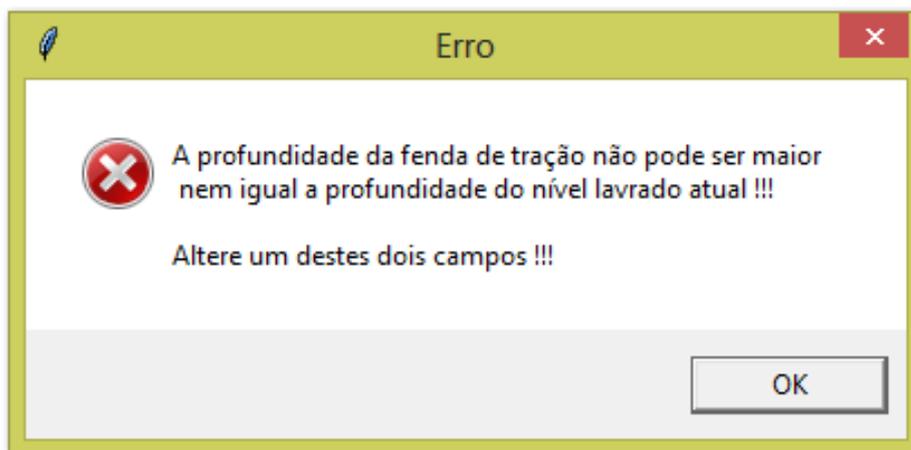


Figura 6.8 - Um dos possíveis erros que podem acontecer no cálculo do ângulo de abatimento pelo método de equilíbrio limite.

6.3.4 LaubscherForm

A classe *LaubscherForm* cria os campos de entrada que são utilizados para o cálculo do ângulo de abatimento pelo método empírico de Laubscher. Ela é composta pelos atributos **entries**, **labelValorFator**, **valorAngulo**, **fator** e pelos métodos **validar** e **calcularFator**. O atributo **entries** é apenas uma lista que guarda as *entries* criadas nesta classe, os atributos **labelValorFator** e **valorAngulo** são as *Labels* que armazenam e exibem os números referentes ao fator calculado e ao ângulo de abatimento encontrado, respectivamente. O atributo **fator** é o valor que foi encontrado para o fator de Laubscher que é calculado de acordo com a equação 2. Na Figura 6.9 pode ser observada a classe *LaubscherForm* em execução.

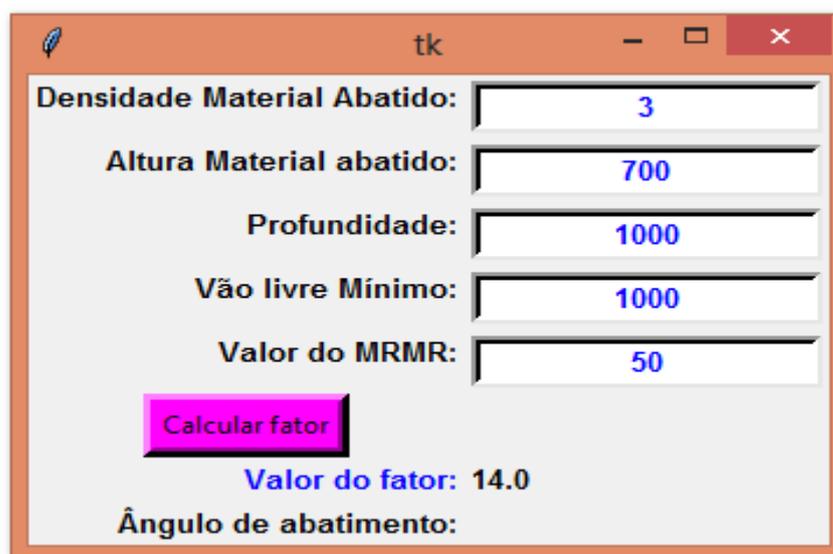


Figura 6.9 - Classe *laubscherform* em execução e seus campos de entrada.

Como pode ser visto na Figura 6.9 a classe *LaubscherForm* é composta por seis instâncias da classe *SmartEntry* cujos valores mínimos, iniciais e finais que são aceitos em cada *Entry* estão ilustrados na Tabela 6.1 sendo que estes valores foram baseados em valores utilizados em casos reais.

Tabela 6.1 - Valores mínimos, iniciais e finais dos campos do módulo *laubscherform*.

Campos	Mínimo	Inicial	Máximo
Densidade do material abatido (kg/m ³)	1	3	5
Altura do material abatido (m)	1	700	4000
Profundidade (m)	1	1000	4000
Vão livre mínimo (m)	1	1000	2500
MRMR	10	50	90

O método **validar** serve para analisar os valores que estão contidos em cada *Entry* criada e o método **calcularFator** tem como objetivo calcular o fator de Laubscher de acordo com a equação 2. Entretanto no método **calcularFator** uma mensagem de erro é exibida na tela caso um dos seguintes eventos aconteça:

1. Existirem valores incorretos em uma ou mais *entries*;
2. A altura do material abatido ser maior ou igual a profundidade da mina;

3. O valor do atributo **fator** ser maior que 100 ou menor que 0.1.

Assim, a classe *LaubscherForm* tem como função criar os campos de entrada para o cálculo do ângulo de abatimento pelo método de Laubscher e calcular o valor do fator de Laubscher através dos valores digitados nos quatro primeiros campos de entrada.

6.3.5 LaubscherGrafico

Esta classe tem como função criar o gráfico de Laubscher (Figura 3.3) sendo a classe *LaubscherGrafico* composta pelos atributos **fundo**, **chart**, **proporcao** e pelos métodos **canvas**, **linhasVerticais**, **linhasHorizontais** e **criaMRMR**. Na Figura 6.10 está ilustrado o gráfico que é criado pela classe *LaubscherGrafico*.

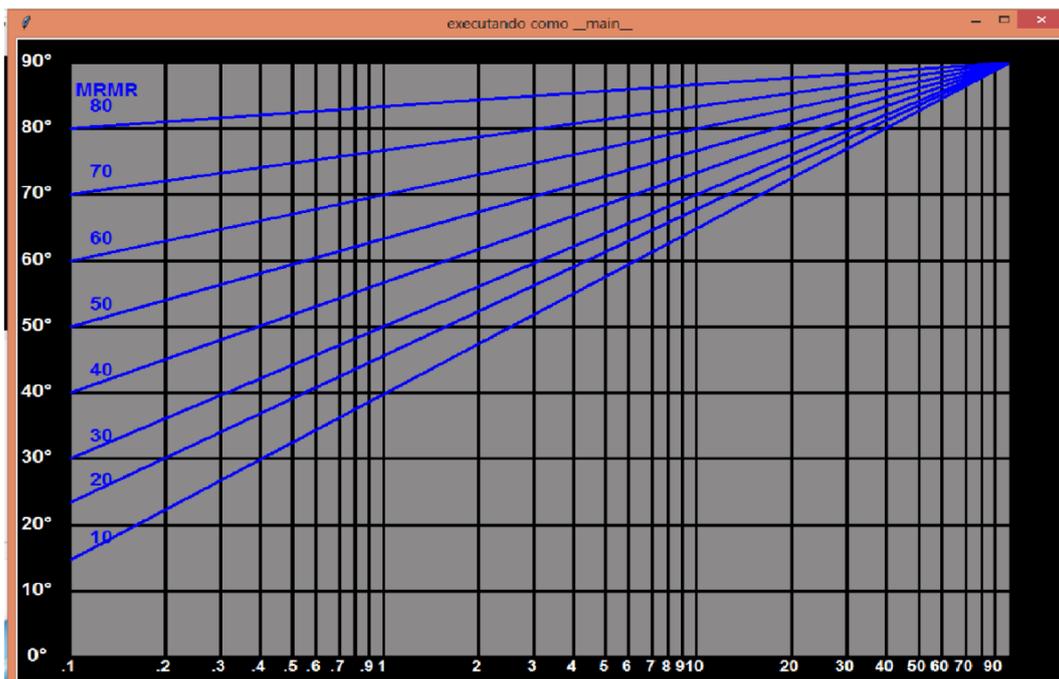


Figura 6.10 - Gráfico de Laubscher gerado pelo programa.

O método **canvas** é responsável por criar o fundo do gráfico e o gráfico propriamente dito. Para criá-los ele precisa dos atributos, **fundo** e **chart**, sendo que o atributo **fundo** consiste em um retângulo de cor preta e o atributo **chart** consiste em um retângulo cinza que está sobreposto sobre o atributo **fundo**. Uma ilustração

dos atributos **fundo** e **chart** ao serem invocados pelo método **canvas** pode ser observada na Figura 6.11.

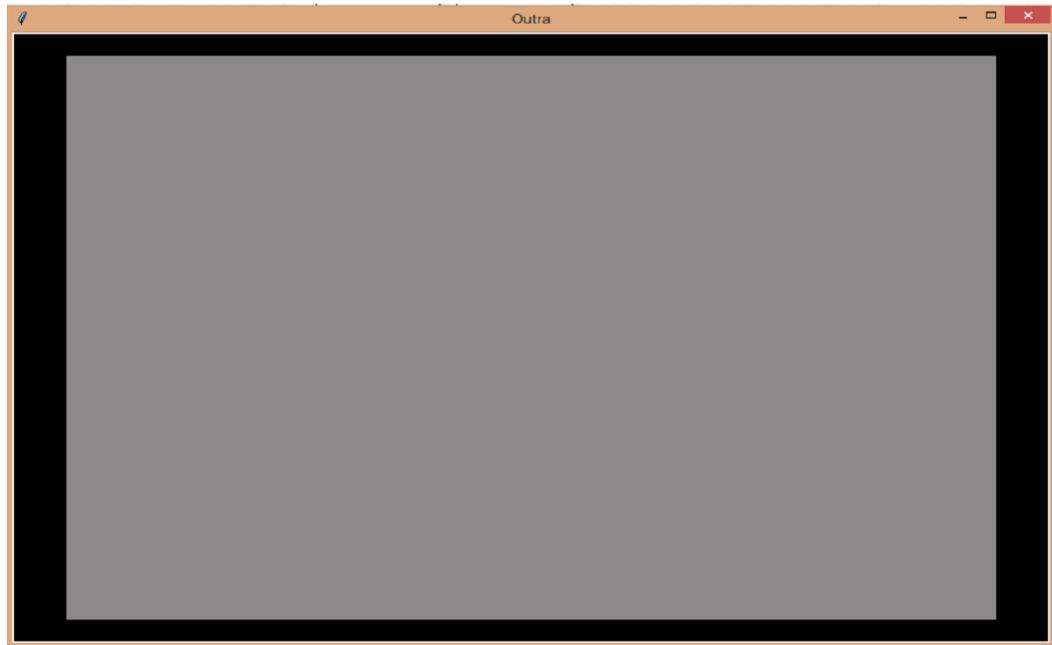


Figura 6.11 - Atributos fundo e gráfico ao serem invocados pelo método canvas.

Como ilustrado na Figura 6.10, o gráfico consiste de linhas verticais, horizontais e inclinadas além de textos numéricos que indicam os valores destas linhas. O método **linhasVerticais** é responsável por criar as linhas verticais que estão exibidas na Figura 6.10. Para criar estas linhas foi necessário dividir todas as linhas em três grupos. O primeiro grupo é composto das linhas verticais que apresentam valores para o fator de Laubscher maiores que 0.1 e menores que 1. O segundo grupo consiste de linhas com valores do fator de Laubscher entre 1 e menores que 10 e o terceiro consiste nas linhas com valores entre 10 e 100. Esta divisão foi feita porque durante a análise do gráfico original foi observado que as linhas do segundo e terceiro grupo correspondiam a uma repetição das linhas do primeiro grupo simplesmente deslocadas. Para criar as linhas verticais e posicioná-las é utilizado o atributo **proporção**, sendo função deste atributo indicar para a classe *LaubsherGrafico* qual é a coordenada x das 10 primeiras linhas verticais em relação a origem. Logo, baseado na posição destas 10 linhas o método

linhasVerticais cria as linhas referentes ao primeiro grupo (fator maior que 0.1 e menor que 1), em seguida o método **linhasVerticais** cria mais 10 linhas verticais, porém estas linhas são deslocadas para a direita. Por fim, o processo é repetido mais uma vez, seguido por um último deslocamento das linhas, de forma a criar o terceiro grupo de linhas. O valor do deslocamento utilizado nas linhas do segundo e terceiro grupo são um terço e dois terços da largura do atributo **gráfico**, respectivamente.

O método **linhasHorizontais** cria as linhas que indicam os diferentes valores de ângulo de abatimento e o método **criaMRMR** cria as linhas inclinadas que correspondem aos valores da classificação de maciço rochosos. Os textos que indicam os valores das linhas verticais, horizontais e inclinadas são criados e posicionados simultaneamente a criação destas linhas.

6.3.6 AbaLaubscher

Este componente é responsável por criar uma instância das classes *LaubscherForm* e *LaubscherGrafico* e fazer uma integração entre elas para obter o ângulo de abatimento a partir do método empírico de Laubscher. Esta classe é composta pelos atributos **width**, **height**, **rastreador**, **linhaFator**, **linhaMRMR**, **linhaSubsidencia** e **angulo** e pelos métodos **yMRMR**, **xFator**, **apagaLinhas**, **plotaLinhas** e **valorAngulo**.

A classe *AbaLaubscher* é um *Frame* que cria uma instância da classe *LaubscherForm* cujo nome é **form** e uma instância da classe *LaubscherGrafico* que recebe o nome **gráfico** sendo que estas instâncias são posicionadas e exibidas na tela uma do lado da outra. O atributo **width** é um número igual a um terço da largura do gráfico (parte cinza da Figura 6.11) e o atributo **height** é um número que corresponde a um nono da altura do gráfico. O atributo **rastreador** é uma variável *booleana* que tem como função auxiliar o programa a excluir os resultados anteriores antes de plotar no gráfico os próximos resultados. Os atributos **linhaFator**, **linhaMRMR** e **linhaSubsidencia** são os resultados exibidos no gráfico na forma de três linhas que correspondem ao valor do fator de Laubscher, MRMR e

do ângulo de abatimento, respectivamente. O atributo **ângulo** é uma *Label* que recebe o texto correspondente ao valor do ângulo de abatimento que está indicado no gráfico pelo atributo **linhaSubsidencia**.

O método **yMRMR** tem como objetivo calcular as coordenadas da linha que representa o valor do MRMR digitado pelo usuário. O método **xFator** tem a função de transformar o valor do fator de Laubscher encontrado em uma coordenada x de forma que esta coordenada será utilizada pelo método **plotaLinhas**. O método **plotaLinhas** cria três linhas no gráfico que correspondem ao valor do fator de Laubscher, do MRMR e do ângulo de abatimento. Também é função do método **plotaLinhas** atribuir qual valor o atributo **angulo** terá. Caso o usuário decida calcular o ângulo de abatimento novamente o método **apagaLinhas** com a ajuda do atributo **rastreador** tem como objetivo apagar as linhas que estão plotadas no gráfico, porém caso nenhuma linha tenha sido gerada o método **apagaLinhas** não será invocado e, portanto, não deletará nenhuma linha. O método **valorAngulo** tem como tarefa apenas exibir na tela o valor do ângulo de abatimento com uma precisão maior do que o valor exposto no gráfico.

6.3.7 Ventilacao1

Esta classe é responsável por calcular os parâmetros de ventilação que estão exibidos na equação 4. Assim, a partir da relação exposta na equação 4, podem ser calculadas pela classe *Ventilacao1* as seguintes variáveis de ventilação de mina: perda de carga, o coeficiente de fricção, densidade do ar, velocidade do ar, o comprimento, o perímetro ou a área da seção transversal da escavação.

A classe é composta pelos atributos **superframe**, **frame1**, **frame2**, **var**, **nomes**, **entries**, **limites**, **unidades**, **lresultado** e pelos seguintes métodos **criarCampos**, **validar** e **calcula**. O atributo **nomes** é uma lista com os nomes dos parâmetros que podem ser calculados (perda de carga, comprimento, perímetro, velocidade, etc.) e o atributo **unidades** é uma lista das unidades destes parâmetros (Pa, m², m, m/s, etc.).

Os atributos **superframe**, **frame1**, **frame2** são *frames* que são utilizados de forma a melhor organizar e dividir a estrutura de exibição das diferentes partes desta classe. O atributo **var** é um número que tem como função indicar para a classe qual parâmetro de ventilação o usuário quer calcular. O atributo **limites** é uma lista com os valores mínimo, inicial e máximo de cada campo de entrada que é criado pela classe *Ventilacao1*. Já os atributos **entries** e **lresultado** são, respectivamente, uma lista que armazena os campos de entrada criados pela classe e uma *Label* que guarda o resultado dos cálculos realizados.

O método **criarCampos** tem como objetivo criar os campos de entrada assim como indicar para a classe *Ventilacao1* a que corresponde cada campo de entrada. Finalmente, o método **validar** é responsável por verificar se existem campos com valores incorretos e o método **calcula** tem como função calcular os parâmetros de ventilação baseado na escolha do usuário de qual termo da equação 4 o usuário deseja calcular.

6.3.8 Ventilacao2

Este componente tem como função realizar o cálculo dos parâmetros de ventilação que estão exibidos na equação 6. Logo, a partir da relação exposta na equação 6 a classe *Ventilacao2* pode calcular os seguintes parâmetros: perda de carga, o fator de fricção, velocidade do ar, o comprimento, o perímetro ou a área da seção transversal da escavação. Para encontrar qualquer um destes valores, esta classe utiliza os seguintes atributos: **superframe**, **frame1**, **frame2**, **var**, **entries**, **limites**, **texto**, **unidades** e **lresultado**. Os métodos desta classe que são utilizados tanto em suas operações de cálculos quanto na operação de criar os campos para a entrada de dados são: **criaCampos**, **validar** e **calcula**.

Os atributos **superframe**, **frame1** e **frame2** são *frames* que tem como função auxiliar no armazenamento dos campos de informações, campos de entrada e resultados gerados. O atributo **var** é um número e, tem como função indicar para a classe *Ventilacao2* qual parâmetro de ventilação presente na equação 6 o usuário quer calcular. O atributo **entries** é uma lista e tem como função armazenar os

campos de entrada de dados e o atributo **texto** é uma lista de *labels* que tem a função de indicar a que corresponde cada um dos campos de entrada armazenados pelo atributo **entries**. O atributo **limites** é uma lista contendo os valores mínimos, iniciais e máximos de cada campo de entrada de valores. Já o atributo **unidades**, tem como função apenas indicar qual é a unidade de medida de cada campo de entrada e do parâmetro que foi calculado. Por último, o atributo **resultado** tem a função de armazenar e exibir o resultado dos cálculos realizados.

O método **criaCampos** cria os campos que serão exibidos na tela a partir da escolha do usuário do que deverá ser calculado. Assim, se o usuário escolher calcular a perda de carga da escavação o método **criaCampos** criará uma interface igual à da Figura 6.12 em que se tem exibido o atributo **texto** (Comprimento, perímetro, velocidade, etc.) e a direita dele os campos de entrada de valores que correspondem ao atributo **entries**.

Calcular: Perda de Carga
 Fator de Fricção
 Comprimento
 Perímetro
 Área Transversal
 Velocidade

Fator de Fricção (kg/m³):

Comprimento (m):

Perímetro (m):

Área Transversal (m²):

Velocidade (m/s):

Perda de Carga :

Figura 6.12 - Interface criada ao selecionar calcular a perda de carga.

Os outros dois métodos restantes são **validar** e **calcula** que têm como função verificar se existem valores inválidos nos campos de entrada e calcular o que foi selecionado pelo usuário, respectivamente.

6.3.9 Ventilacao3

A classe *Ventilacao3* realiza os cálculos e calcula o valor dos parâmetros de ventilação de mina que estão exibidos na equação 8. Assim, através do uso desta classe o usuário pode calcular um dos seguintes parâmetros: perda de carga, fator de fricção, vazão de ar, comprimento, perímetro ou área transversal da escavação. A classe *Ventilacao3* é composta pelos atributos **superframe**, **frame1**, **frame2**, **lista**, **var**, **listaEntry**, **texto**, **limites** e **lresultado** e pelos métodos **criaCampos**, **validar**, **carculaPerda**, **calculaFriccao**, **calculaComprimento**, **calculaPerimetro**, **calculaArea** e **calculaVazao**.

Os atributos **superframe**, **frame1** e **frame2** têm como função apenas auxiliar na organização da interface da tela e na exibição dos campos de entrada de valores. O atributo **lista** é uma lista com os nomes dos parâmetros que podem ser calculados, sendo estes nomes: perda de carga, fator de fricção, comprimento, perímetro, área transversal e vazão. A função do atributo **lista** é indicar ao usuário o que cada campo de entrada é e o que foi calculado pela classe. O atributo **var** é um número que indica à classe *Ventilacao3* qual opção foi selecionada pelo usuário e, portanto, que ação o programa deve tomar caso o botão de calcular seja pressionado. Já o atributo **listaEntry**, tem como objetivo armazenar os campos de entrada de dados e o atributo **texto** exibir um texto indicando o que cada campo de entrada de dados representa. Na Figura 6.13 pode ser observada a classe *Ventilação3* em execução, nela está ilustrado os atributos **lista** (canto superior), **texto** (canto inferior esquerdo) e o atributo **listaEntry** que são os campos de entrada para valores localizados a direita do atributo **texto**.

Calcular: Perda de carga
 Fator de Fricção
 Comprimento
 Perímetro
 Área Transversal
 Vazão

Fator de Fricção (kg/m³):

Comprimento (m):

Perímetro (m):

Área Transversal (m²):

Vazão (m³/s):

Perda de carga: 90.0 Pa

Calcular

Figura 6.13 - Classe Ventilacao3 com seus principais atributos.

O atributo **limites** é apenas uma lista com os valores mínimos, iniciais e máximos permitidos em cada campo de entrada de valores. Já o atributo **resultado** é apenas uma *Label* que exibe o resultado dos cálculos realizados. Na Figura 6.13, por exemplo, **resultado** exibe o texto 90.0 Pa, porém quando o programa é iniciado **resultado** se encontra em branco pois nenhum cálculo ainda foi realizado.

O método **criaCampos** tem como função criar e exibir na tela os campos relacionados a entrada de dados, que podem ser vistos na Figura 6.13. O método **validar** tem como objetivo verificar se os campos de entrada apresentam valores que estão entre os valores máximos e mínimos permitidos. Os métodos **calculaPerda**, **calculaFriccao**, **calculaComprimento**, **calculaPerimetro**, **calculaArea** e **calculaVazao** têm como função calcularem a perda de carga, fator de fricção, comprimento da escavação, perímetro, área da seção transversal e a vazão de ar que passa pela escavação, respectivamente. Estes métodos são chamados pelo botão que está exibido na Figura 6.13 e apenas um destes métodos pode ser invocado por vez. Como ilustrado pela Figura 6.13, a perda de carga está selecionada como a opção a ser calculada, logo se o botão calcular for pressionado ele invocará o método **calculaPerda**. Se o *rádio button* referente ao perímetro estivesse selecionado o botão calcular invocaria o método **calculaPerimetro**.

6.3.10 Ventilacao4

Esta classe tem como função realizar os cálculos e encontrar os parâmetros de ventilação de mina que estão exibidos na equação 10. Logo, é através desta classe que o usuário pode calcular os seguintes parâmetros: perda de carga, resistência da via de ventilação ou vazão de ar. A classe *Ventilacao4* contém os atributos **superframe**, **frame1**, **frame2**, **campos**, **var**, **limites**, **unidades**, **elista** e **lresultado** e os métodos **criaCampos**, **validar** e **calcula**.

Os atributos **superframe**, **frame1** e **frame2** são *frames* que têm como objetivo apenas auxiliar na organização das informações que estão exibidas na tela. O atributo **campos** tem a função de criar e exibir as informações que indicam ao usuário a que corresponde cada campo de entrada. A direita do atributo **campos** se encontra o atributo **elista** cujo objetivo é exibir e armazenar os campos destinados à entrada de dados. Já o atributo **unidades**, é uma lista com as unidades dos campos que são exibidos na tela e o atributo **limites** é uma lista contendo os valores mínimos, iniciais e máximos de cada campo de entrada. O atributo **var** é um número que tem como função indicar a classe *Ventilacao4* qual parâmetro o usuário escolheu calcular: perda de carga, resistência ou vazão. Já o atributo **lresultado** tem como objetivo simplesmente armazenar o resultado dos cálculos realizados e exibi-lo para o usuário. Na Figura 6.14 está ilustrada uma instância da classe *Ventilacao4* que apresenta os atributos **campos**, **unidades** e **elista**.

Calcular: Perda de Carga
 Resistência
 Vazão

Resistência (kg/m⁷):

Vazão (m³/s):

Perda de Carga:

Figura 6.14 - Classe *Ventilacao4* com seus principais atributos.

O método **criaCampos** tem como função criar os campos de informações e os campos de entrada como ilustrado na Figura 6.14, já o método **validar** tem como objetivo verificar se os valores digitados em cada campo de entrada são valores válidos. Caso os valores não sejam válidos, o método **validar** informa ao usuário que existem campos com valores incorretos e que ele deve corrigi-los. O método **calcular** tem como objetivo realizar os cálculos para encontrar a perda de pressão, resistência ou vazão de ar, sendo que qual destes cálculos será realizado dependerá da escolha do usuário.

6.3.11 AbaVentilacao

Esta classe tem como objetivo apenas criar uma instância das classes *Ventilacao1*, *Ventilacao2*, *Ventilacao3* ou *Ventilacao4*. A classe *AbaVentilacao* possui os atributos **superframe**, **frame1**, **frame2** e **var** e o método **exibe**.

Os atributos **superframe**, **frame1** e **frame2** são *frames* com a função de auxiliar na organização de como as diferentes partes do programa são exibidas para o usuário. O atributo **var** é um número que indica para a classe *AbaVentilacao* qual fórmula o usuário escolheu usar. Na Figura 6.15 pode ser observada a classe *AbaVentilacao*, com as 4 equações que podem ser usadas para o cálculo dos parâmetros de ventilação de mina (equações 4, 6, 8 e 10). O usuário deve então selecionar qual equação ele vai usar e partir desta escolha a classe *AbaVentilacao* criará uma instância da classe correspondente. Na Figura 6.15 a primeira equação está selecionada, assim na parte inferior é criada uma instância da classe *Ventilacao1* que exibe todos os componentes da equação 4. Se a segunda equação, exibida na Figura 6.15, fosse selecionada, a parte inferior da Figura 6.15 ficaria idêntica a Figura 6.12.

Fórmula escolhida: 1) $p = (f \cdot L) \cdot (\text{per} / A) \cdot d \cdot (v^2/2)$

2) $p = k \cdot L \cdot (\text{per} / A) \cdot v^2$

3) $p = k \cdot L \cdot (\text{per} / A^3) \cdot Q^2$

4) $p = R \cdot Q^2$

Calcular: Perda de Carga

Coeficiente de Fricção

Comprimento

Perímetro

Área Transversal

Densidade do ar

Velocidade

Coeficiente de Fricção:

Comprimento (m):

Perímetro (m):

Área Transversal (m²):

Densidade do ar (kg/m³):

Velocidade (m/s):

Perda de Carga :

Figura 6.15 - Classe *AbaVentilacao*.

6.3.12 Janela

A classe *Janela* tem como função criar uma instância das classes *AbaNumerico*, *AbaLaubscher* e *AbaVentilacao* em apenas uma única janela. Esta classe é o resultado final do programa, de forma que quando o programa for instalado é com esta classe que o usuário vai interagir.

O atributo **root** refere-se à janela da interface gráfica que armazena os *frames* **painelLateral** e **frameCalculos**. O **painelLateral** é um *Frame* que armazena os botões de funcionalidades do programa e o *Frame* **frameCalculos** armazena os campos de entrada de dados, os resultados e todos os campos relativos ao botão escolhido pelo usuário.

6.4 Detalhes de implementação

A solução foi desenvolvida em *Python*, versão 3.4.4, com o auxílio dos ambientes de desenvolvimento IDLE e LiClipse, e se encontra disponível no endereço <https://bitbucket.org/ovidioduarte/underpy>.

O programa foi dividido em 11 classes, contidas em sete módulos, conforme exibido no Quadro 6.1. O módulo mais usado é o smartentry, que contém a implementação da classe *SmartEntry* e é importado por quase todos os outros. Os módulos laubscherGrafico e laubscherform são importados apenas uma vez para auxiliarem na construção do módulo abalaubscher.

Quadro 6.1 – Distribuição das classes em módulos.

Módulos	Classes criadas
abalaubscher	AbaLaubscher
laubscherform	LaubscherForm
laubschergrafico	LaubscherGrafico
metodonumerico	AbaNumerico
smartentry	SmartEntry
underpy	Janela
ventilation	Ventilacao1
	Ventilacao2
	Ventilacao3
	Ventilacao4
	AbaVentilacao

Finalmente, os módulos abalaubscher, metodonumerico e ventilation são utilizados para a construção da interface final, por meio da classe *Janela*, definida em underpy.

6.5 Aspectos de usabilidade

A interface inicial apresenta um painel lateral com três botões, de nomes Método Numérico, Método Empírico e Ventilação, pelos quais o usuário seleciona a funcionalidade desejada, afetando o que é exibido no painel à direita (Figura 6.16). Ao ser iniciado, o componente para o cálculo do ângulo de abatimento pelo método de equilíbrio limite é exibido.

Metodo Numerico	Inclinação do novo plano de falha:	40
Metodo Empirico	Nova profundidade da fenda de tração:	500
Ventilação	Mergulho da superfície do solo:	10
	Mergulho do minério:	30
	Profundidade do nível lavrado atual:	800
	CALCULAR	
	Ângulo de Abatimento:	

Figura 6.16 - Classe Janela exibindo a interface da classe AbaNumerico

Os cálculos são realizados de maneira interna e o usuário não tem controle na forma como eles são realizados. Entretanto, é possível escolher quais cálculos são feitos. Por exemplo, se o usuário clicar no botão método empírico, digitar os valores desejados nos campos de entrada e em seguida pressionar o botão calcular fator (Figura 6.9), o programa realiza o cálculo para obter o fator de Laubscher.

As *smart entries* aceitam somente números, dentro de limites definidos pelo programador. Além de indicar campos inválidos pela alteração da cor de fundo do componente, uma mensagem de erro aparece para o usuário quando algum cálculo é solicitado por meio de um botão e os campos de entrada de dados contêm valores incorretos. Além disso, caso as *entries* contenham campos em branco, letras ou símbolos, os cálculos não são realizados e uma mensagem de erro também é exibida.

O botão Ventilação, quando pressionado, exibe um formulário no qual o usuário deve escolher qual fórmula ele deseja usar e, de acordo com a mesma, qual parâmetro deve ser calculado. Por padrão, o programa exibe a interface para o cálculo da perda de carga, pela relação exposta na equação 4.

7 RESULTADOS

O resultado obtido foi uma solução de *software* que recebeu o nome de **underpy** versão 0.1 e que apresenta uma interface idêntica à que pode ser observada na Figura 6.16. Com o intuito de verificar a relevância da utilização do *software*, a usabilidade de sua interface e o seu comportamento quando em execução por diferentes usuários, foi elaborada uma ficha de análise do programa (Apêndice A). A ficha de análise foi feita conforme alguns dos aspectos considerados relevantes pela NBR ISO/IEC 9126-1, que é uma norma para a qualidade de produtos de *software* e define um conjunto de parâmetros que padronizam o que deve ser considerado na avaliação. Os critérios considerados foram os seguintes:

- **Funcionalidade** – Indica se as funções e tarefas executadas pelo *software* atendem as necessidades do usuário;
- **Confiabilidade** – Indica se o *software* mantém seu desempenho e não apresenta comportamentos inesperados durante sua execução;
- **Usabilidade** – Está relacionada à capacidade do *software* de ser compreendido, operado e aprendido pelo usuário;
- **Eficiência** – Verifica se os recursos gastos pelo computador são compatíveis com o desempenho do *software*.

Os demais critérios não foram considerados, pois não se aplicavam a natureza de utilização do *underpy*. Como o programa desenvolvido não executa um número demasiado de funções, a ficha de análise criada contém poucas perguntas e também foi influenciada pelo perfil dos testadores candidatos: estudantes de Engenharia de Minas ou Técnicos em Mineração.

O Quadro 7.1 apresenta o perfil das seis pessoas que testaram o *software*. Foi escolhida uma pessoa de uma área distinta, presumivelmente com menos

experiência na área de mineração, com o objetivo de saber se a interface estava realmente clara e se o programa poderia ser facilmente utilizado e compreendido.

Antes de responderem a ficha de avaliação, os testadores receberam uma breve explicação sobre a finalidade do programa e os principais conceitos envolvidos.

Quadro 7.1 – Perfil dos testadores do protótipo *underpy*.

Testadores	Curso	Período
U1	Engenharia de Minas	10º
U2	Zootecnia	2º
U3	Engenharia de Minas	8º
U4	Engenharia de Minas	8º
U5	Engenharia de Minas	8º
U6	Engenharia de Minas	10º

Quadro 7.2 - Respostas das pessoas a avaliação a avaliação do *underpy*

Perguntas	Testadores					
	U1	U2	U3	U4	U5	U6
1	Razoável	Ruim	Razoável	Ruim	Bom	Razoável
2	Nenhum	Nenhum	Ruim	Ruim	Bom	Razoável
3	Razoável	Nenhum	Nenhum	Ruim	Razoável	Razoável
4	Razoável	Razoável	Nenhum	Razoável	Bom	Ruim
5	Bom	Bom	Bom	Bom	Razoável	Bom
6	Bom	Bom	Bom	Bom	Bom	Bom
7	Bom	Bom	Bom	Bom	Bom	Bom
8	Bom	Bom	Bom	Bom	Razoável	Bom
9	Bom	Bom	Razoável	Razoável	Bom	Bom
10	Bom	Bom	Bom	Bom	Bom	Bom
11	Bom	Nenhum	Bom	Bom	Bom	Bom
12	Bom	Bom	Bom	Bom	Bom	Bom

Como exibido no

Quadro 7.2, a maioria das pessoas que responderam o questionário não tem conhecimento dos métodos de abatimento nem de como calcular os diferentes parâmetros de ventilação de mina, embora tenham algum conhecimento dos

métodos de lavra subterrânea. Quanto ao domínio na utilização de programas de computador (pergunta quatro) as respostas oscilaram entre nenhum a bom, indicando que o conhecimento de informática dos usuários era de certa maneira variado. Nas perguntas referentes à instalação e ao consumo de recursos do computador (perguntas cinco a sete) as respostas indicam que o desempenho do programa é muito bom não afetando em nada o desempenho das máquinas em que ele foi instalado.

Em relação às perguntas que analisavam a facilidade de aprendizado do usuário, compreensão da interface, utilização do programa em sala de aula como uma ferramenta de aprendizado e ao uso dele na prática, todas obtiveram respostas quase idênticas, que comprovam a simplicidade e facilidade de manusear o programa, assim como a sua possível utilização em sala de aula.

8 CONCLUSÃO

O *software underpy*, por apresentar uma interface simples e de fácil compreensão, pode ser uma boa ferramenta para ser utilizada em sala de aula com o objetivo de auxiliar na compreensão e aumentar o entendimento dos alunos sobre como alguns fatores como a classificação de maciços rochosos, profundidade da escavação, mergulho do corpo de minério interferem no ângulo de abatimento. O *software* também pode ser empregado para o cálculo de diferentes parâmetros de ventilação em minas reais, uma vez que ele é versátil quanto ao parâmetro que pode ser calculado e também as equações usadas nos cálculos para a obtenção destes parâmetros são as mesmas usadas na prática pela maioria das minas subterrâneas.

Como possíveis trabalhos futuros estão melhorias no *underpy* para que ele possa salvar os resultados gerados, acrescentar ao programa funções para o cálculo de diluição, formas de calcular o volume que é extraído por dia e os lucros e custos provenientes dessa extração.

9 REFERÊNCIAS BIBLIOGRÁFICAS

- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. 2003. NBR ISO/IEC 9126-1: *Engenharia de software – Qualidade de produto – Parte 1: Qualidade de modelo*. Rio de Janeiro, 21 p.
- Bessinger S. L. 2011. Longwall Mining. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 1399-1415.
- Brady B. H. G. & Brown E. T. 2004. *Rock Mechanics for Underground Mining*. Boston, Kluwer Academic, 3 ed., 628 p.
- Brannon C. A., Carlson G. K., Casten T. P. 2011. Block Caving. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 1437-1451.
- Brown E. T. 2002. *Block Caving Geomechanics*. Brisbane, Julius Kruttschnitt Mineral Research Center, 1 ed., 515 p.
- Calizaya F. & Marks J. 2011. Heat, Humidity, and Air Conditioning. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 1611-1624.
- David M. F. 2011. *Programação Orientada a Objetos: uma introdução*. Disponível em: <http://www.hardware.com.br/artigos/programacao-orientada-objetos/>. Acesso em: 20 de Jun. 2016.
- Dunstan G. & Power G. 2011. Sublevel Caving. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 1417-1436.
- Grady B., Maksimchuk R. A., Engle M. W., Young B. J., Conallen J., Houston K. A. 2007. *Object-oriented analysis and design with applications*. Boston, Pearson Education, 3 ed., 691 p.
- Harrison J. P. 2011. Mine Subsidence. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 627-644.
- Hartman H. L. & Mutmasky J. M. 2002. *Introductory Mining Engineering*. Hoboken, John Wiley & Sons, 2 ed., 570 p.
- Kratzsch H. 1983. *Mining Subsidence Engineering*. Berlin, Springer-Verlag, 1 ed., 546 p.
- Laubscher D. H. 1990. A geomechanics classification system for the rating of rock mass in mine design. Johannesburg, *Journal of the South African Institute of mining and metallurgy*, **90**(10):257-273.
- Mcperson M. J. 1993. *Subsurface Ventilation and Environmental Engineering*. New Delhi, Springer Netherlands, 1 ed., 905 p.

- Nelson M. G. 2011. Evaluation of mining methods and systems. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 341-348.
- Rossum G. V. 2009. Introduction and Overview. Disponível em: <http://python-history.blogspot.com.br/2009/01/introduction-and-overview.html>. Acesso em: 12 de Mar. 2016.
- Rossum G. V. 2009. Personal History – part 1, CWI. Disponível em: <http://python-history.blogspot.com.br/2009/01/personal-history-part-1-cwi.html>. Acesso em: 11 de Mar. 2016.
- Rossum G. V. 2009. Personal History – part 2, CNRI and beyond. Disponível em: <http://python-history.blogspot.com.br/2009/01/personal-history-part-2-cnri-and-beyond.html>. Acesso em: 14 de Mar. 2016.
- Stephan G. 2011. Cut and Fill Mining. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 1365-1373.
- Summerfield M. 2013. *Programação em Python 3: uma introdução completa à linguagem Python*. Rio de Janeiro, Alta Books, 1 ed., 495 p.
- Tuck M. A. 2011. Mine Ventilation. *In: Darling P. SME Mining Engineering Handbook*. EUA, Society for Mining Metallurgy and Exploration, 3 ed., p. 1577-1594.
- Vyazmensky A. 2008. *Numerical modeling of surface subsidence associated with block cave mining using a finite element / discrete element approach*. Phd Theses, Simon Fraser University, Burnaby, 241 p.

Glossário

Canvas: Elemento de interface gráfica que pode ser usado para criar e plotar gráficos ou até mesmo ser usado para fazer desenhos.

Crosscut: chamada também de travessa abertura orientada perpendicular à direção do corpo de minério que frequentemente conecta outras aberturas tais como drifts.

Decline: Abertura secundária, inclinada, realizada de forma descendente para conectar níveis, e estando às vezes ligados à inclinação de um depósito.

Descontinuidade: termo usado para se referir ao conjunto de fraturas ou estruturas da rocha tais como juntas, falhas, planos de acamamento, contatos ou estruturas que tem pouca ou nenhuma resistência tracional.

Direção: Orientação ou rumo de uma linha horizontal que é o resultado da intersecção de um plano vertical com estruturas geológicas tais como camadas, fraturas, falhas ou descontinuidades.

Drawbell: Escavação em forma de funil utilizada para coletar minério desmontado utilizando somente o fluxo do material por gravidade.

Drawpoint: Ou ponto de carregamento, é uma região na mina subterrânea abaixo de um realce que utilizando a gravidade move material de maneira descendente até um transporte seja por uma calha ou por uma esteira transportadora.

Drift: Também chamada de cabeceira, é uma escavação horizontal no corpo de minério ou próxima a ele feita paralela à direção do corpo de minério.

Entry: Elemento de interface gráfica que permite a entrada de dados necessária que devem ser processados pelo programa.

Exploração: O processo de extrair óleo, gás, minerais ou rochas da superfície da terra para o seu posterior beneficiamento ou venda.

Frame: Contêiner utilizado para agrupar vários elementos de uma interface gráfica.

Fratura: Qualquer descontinuidade plana ou subplana, delgada em uma direção em comparação às outras duas e formada por esforço externo. Frequentemente estão relacionadas a descontinuidades nas propriedades mecânicas e a deslocamentos.

Função: É um pequeno pedaço de código dentro de um programa maior que possui a finalidade de resolver um problema ou operação específica dentro deste programa maior.

Hanging wall: Também chamado de capa, é a parte da rocha encaixante, acima de um veio, ou acima do minério que está sendo lavrado.

Instância: Ocorrência ou exemplo de uma classe. Quando se tem uma ou mais ocorrências de uma classe diz-se que se tem uma ou mais instâncias de uma classe.

Label: É um elemento de interface gráfica com um texto ou uma explicação com o objetivo de ajudar ao usuário interagir com o programa.

Lista: Uma lista é um conjunto ordenado de valores, onde cada valor é identificado por um índice. Os valores que compõem uma lista são chamados elementos. Em python uma lista pode ser composta apenas de números, letras ou de ambos juntos assim tanto [1, 2, 3], [a, b, c, d] e [1, a, e, 3] são listas.

Maciço Rochoso: Conjunto de rocha intacta e suas descontinuidades na presença ou não de água inserido em um campo de tensões.

Main entry: Ou entrada principal é uma abertura horizontal primária que dá acesso a uma mina subterrânea ou a outras aberturas secundárias e é usada para o transporte, ventilação entre outras funções.

Mergulho: Ângulo formado entre um plano horizontal e a reta de máxima inclinação de um plano de estruturas de interesse geológico tais como camadas, fraturas, falhas ou descontinuidades.

Objeto: Objetos são abstrações de objetos do mundo real. Assim objetos podem ser números, nomes, letras ou até mesmo funções em um programa.

Orepass: Passagem vertical ou inclinada para passagem descendente do minério geralmente equipada com comportas ou outros mecanismos de controle de fluxo. Tal abertura é escavada no minério e conecta um nível com um poço de içamento ou um nível inferior.

Panel: Ou simplesmente painel, é um grupo de frentes ou câmaras separadas das outras por grandes pilares.

Radio Button: Elemento de interface gráfica que pode apresentar dois estados: ativado ou desativado. *Radio Button* são usados quando se tem um conjunto limitado de escolhas que são mutuamente exclusivas.

Raise: Também chamada de subida é uma escavação feita de baixo para cima ligando um nível a outro da mina, com objetivo de ventilação, passagem de pessoal, passagem de minério, recebendo denominação, geralmente em função dessa finalidade.

Ramp: Ou rampa é uma escavação inclinada usualmente feita de maneira descendente criada para conectar níveis e usada para transporte.

Shaft: Também chamado de poço vertical é uma escavação vertical que conecta o subsolo com a superfície. *Shafts* têm como objetivos transportar minério ou estéril, bombear água, transportar homens, transportar materiais e equipamentos e tem até mesmo à função de ventilação para a mina subterrânea.

Stope: Ou realce, é a escavação subterrânea, com desenvolvimento em volume, resultante da extração de minério, ocupada ou não por equipamentos e por

pessoal. O termo é normalmente aplicado para a extração de minério, mas não se incluem os minérios retirados dos poços principais e túneis auxiliares, cabeceiras e outras aberturas para desenvolvimento.

Undercut: Escavação horizontal, de pequenas dimensões, realizada sob o minério, usualmente uma câmara, para induzir o seu abatimento.

APENDICE A – Ficha de avaliação do *underpy*

Nome:

Esta ficha tem como função analisar alguns aspectos de interface e de uso do software *underpy*. As perguntas têm como objetivo analisar alguns aspectos do programa como instalação e interface e a relevância do programa para o uso tanto na prática como em sala de aula.

O questionário é composto de doze perguntas (12) e para respondê-las basta preencher o campo correspondente a sua resposta com um **X**. Nos casos em que não souber responder ou caso a sua resposta para a pergunta não se enquadre nas respostas possíveis deve-se marcar o campo **nenhum** com um **X**.

Num.	Perguntas	Bom/Boa	Razoável	Ruim	Nenhum
1	Sobre métodos de lavra subterrânea, você considera seu conhecimento como:				
2	Seu conhecimento sobre as formas de calcular o ângulo de abatimento é:				
3	Seu conhecimento sobre os parâmetros de ventilação de mina e a forma como calculá-los é:				
4	Seu domínio sobre a utilização de programas de computador é:				
5	A velocidade de instalação do programa foi:				
6	Desempenho do software durante a sua execução foi:				
7	Desempenho do computador/notebook durante a execução do programa:				
8	Entendimento da interface gráfica e de como interagir com o programa:				
9	Você considera a sua velocidade de aprendizado de utilização do programa como:				
10	A velocidade que o programa fornece respostas é:				
11	Se utilizado em uma situação real você acredita que o desempenho do software seria:				
12	A utilização do software em sala de aula como uma ferramenta de aprendizado seria uma ideia:				
Observações e sugestões:					