
# RECUPERAÇÃO DE INFORMAÇÃO ORIENTADA AO DOMÍNIO DA MATEMÁTICA

Flavio Barbieri Gonzaga

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientadores: Valmir Carneiro Barbosa
Geraldo Bonorino Xexéo

Rio de Janeiro
Março de 2013

RECUPERAÇÃO DE INFORMAÇÃO ORIENTADA AO DOMÍNIO DA MATEMÁTICA

Flavio Barbieri Gonzaga

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Valmir Carneiro Barbosa, Ph.D.

Prof. Geraldo Bonorino Xexéo, D.Sc.

Prof. Artur Ziviani, Dr.

Prof. Daniel Ratton Figueiredo, Ph.D.

Prof. Geraldo Zimbrão da Silva, D.Sc.

Prof. Sean Wolfgand Matsui Siqueira, Dr.

RIO DE JANEIRO, RJ – BRASIL
MARÇO DE 2013

Gonzaga, Flavio Barbieri

Recuperação de Informação Orientada ao Domínio da Matemática/Flavio Barbieri Gonzaga. – Rio de Janeiro: UFRJ/COPPE, 2013.

XII, 101 p.: il.; 29, 7cm.

Orientadores: Valmir Carneiro Barbosa

Geraldo Bonorino Xexéo

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2013.

Referências Bibliográficas: p. 75 – 79.

1. Busca Matemática. 2. Redes Complexas. 3. MathWorld. 4. DLMF. 5. Wikipedia. I. Barbosa, Valmir Carneiro *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

*Este trabalho a dedicado à minha*
*mãe Rosângela, e à minha*
*namorada Lilian.*

# Agradecimentos

Primeiramente gostaria de agradecer aos meus familiares, em especial à minha mãe Rosângela, minha namorada Lilian, meus irmãos Fabrício, Fabiano e Gustavo, minhas cunhadas Layra e Gisele, meu pai Edino e minha madrasta Majô.

A todos os professores e funcionários do Programa de Engenharia de Sistemas. Em especial aos meus orientadores Prof. Valmir e Prof. Geraldo Xexéo pelo empenho e por terem sido sempre muito presentes e atuantes no trabalho desenvolvido. Sem a ajuda deles a ideia desse trabalho não estaria se concretizando hoje. Aprendi com ambos muitas coisas que levarei para o resto da vida, inclusive no meu modo de trabalhar.

Aos amigos Professores da Universidade Federal de Alfenas, aos nossos técnicos pelo suporte prestado e pelo comprometimento com o nosso curso, e aos estudantes que trabalharam e continuam trabalhando em projetos relacionados ao aqui apresentado.

Gostaria de agradecer aos membros da banca pelas correções e sugestões ao presente trabalho. Foram devidamente anotadas, e servirão de norte nas nossas pesquisas futuras.

Encerro agradecendo aos amigos dos tempos de república, que me receberam de braços abertos, e aos muitos amigos que fiz na COPPE durante esses 6 anos de desenvolvimento do trabalho. Cada um de vocês foi extremamente importante, não só pela ralação nas disciplinas, como também pelos conselhos e momentos de descontração.

Procurei não citar muitos nomes, para não correr o risco de esquecer de alguém. Foram muitas pessoas que me ajudaram nesses 6 longos anos, e pelas quais terei sempre muito carinho e admiração.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

# RECUPERAÇÃO DE INFORMAÇÃO ORIENTADA AO DOMÍNIO DA MATEMÁTICA

Flavio Barbieri Gonzaga

Março/2013

Orientadores: Valmir Carneiro Barbosa
Geraldo Bonorino Xexéo

Programa: Engenharia de Sistemas e Computação

O estudo da organização do conhecimento matemático, bem como a busca nesse domínio têm sido foco de alguns trabalhos recentes na literatura. O presente trabalho, cujo objetivo é o desenvolvimento de uma ferramenta de busca por fórmulas matemáticas, começa com a realização de um estudo detalhado da estrutura do conteúdo matemático com base em três das principais bibliotecas online: *Wikipedia* (apenas a parte matemática), *MathWorld* e *DLMF*. Como parte desse estudo são exibidas a presença de componentes fortemente conexas gigantes em todas elas, bem como a ausência de lei de potência nas distribuições que descrevem medidas locais (tais como graus, medidas de centralidade, dentre outras), juntamente com uma análise sobre o desempenho de cada uma dessas medidas como critério na ordenação de resultados em uma busca textual. O estudo dessas métricas fornece então uma intuição para a ordenação no domínio matemático. Na construção da busca por fórmulas, optou-se por desenvolver um analisador léxico para as linguagens nas quais as expressões são representadas nas bibliotecas. O objetivo é interpretar símbolos semelhantes (por exemplo, $x$ ou $y$) como um mesmo *token* (*VARIAVEL*). Assim, foi possível a obtenção da ferramenta de busca que abrange cerca de 330 000 expressões, que por possuir as fórmulas caracterizadas segundo *tokens* definidos, oferece um certo grau de liberdade para diferentes notações dada uma mesma fórmula. O trabalho encerra com um estudo comparativo da qualidade da ferramenta desenvolvida com a *Symbolab*, outra proposta semelhante. A ferramenta aqui descrita é denominada *SearchOnMath* e está disponível em `http://searchonmath.com/`.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

INFORMATION RETRIEVAL ORIENTED TO MATHEMATICAL DOMAIN

Flavio Barbieri Gonzaga

March/2013

Advisors: Valmir Carneiro Barbosa
Geraldo Bonorino Xexéo

Department: Systems Engineering and Computer Science

The study of the organization of mathematical knowledge, as well as the search in this area has been focused by some recent works in the literature. The present study whose goal is the development of a search engine for mathematical expressions, starts with the realization of a detailed study of the mathematica structure content, based on three major online libraries: Wikipedia (only the mathematical content), MathWorld and DLMF. As part of this study are shown the presence of giant strongly connected components (GSCC) in all of them, well as the absence of power law in distributions that describe local features (such as degrees, centrality measures, among others), together with an analysis of the behavior of each of these measures as a criterion on ranking results in text search. The study of such metrics provides then an intuition about ranking order applied on Math domain. On development of search engine for formulas, a lexical analyzer was built for languages in which expressions are represented in libraries. The objective is interpret similar symbols (like $x$ or $y$) as a same token (VARIABLE). Thus, it was possible to achieve the search engine that covers about 330 000 formulas, where formulas are characterized according to some defined tokens, giving so a certain degree of freedom for different representations of a same formula. The work concludes with a study of the quality of the tool developed compared to Symbolab, another similar proposal. The tool described here is called SearchOnMath, and is available at `http://searchonmath.com/`.

# Sumário

# Lista de Figuras

# Lista de Tabelas

# Capítulo 1

# Introdução

Com o contínuo crescimento da Internet, bem como dos recursos disponíveis, surgem a cada dia novos desafios que vão desde a recuperação de conteúdo de interesse de cada usuário bem como da exibição dos recursos de forma agradável, onde pode-se citar como exemplo o *Google*, que permite visualizar uma miniatura das páginas retornadas, sem que seja necessário acessá-las. Ferramentas de busca são uma alternativa na recuperação desse conteúdo, que justamente pelo crescimento da rede mundial, necessitam de algoritmos escaláveis, e de mecanismos que facilitem buscas específicas. Ao se analisar a evolução da quantidade de consultas feitas por dia em ferramentas de busca, observa-se a crescente demanda por esses serviços: em 1994, a *World Wide Web Worm* recebia cerca de 1 500 consultas por dia; em 1997, o *Altavista* realizava cerca de 20 milhões de consultas por dia [4]. A estimativa em 2009 era de que só nos Estados Unidos, algo próximo a 300 milhões de consultas eram realizadas por dia no *Google*, desconsiderando o uso de demais ferramentas como o *Bing* e o *Yahoo!* [5].

Além do formato tradicional das ferramentas citadas, que buscam à partir de palavras, as empresas começam a voltar o foco para buscas mais inteligentes e de conteúdo específico. Alguns exemplos são a busca por artigos científicos, realizada pelo *Scholar*[1] do *Google*; além da ferramenta de conhecimento computacional *Wolfram Alpha*[2], onde o usuário pode obter relatórios elaborados de forma automática com base na consulta realizada. No *Wolfram Alpha*, uma busca envolvendo as palavras-chave *Microsoft* e *Google* por exemplo, realizará uma comparação entre as empresas, incluindo até o valor das ações de ambas na bolsa de valores.

Ao se observar a tendência no desenvolvimento de ferramentas capazes de realizar buscas específicas, pode-se listar alguns conteúdos que ainda não são tratados de forma adequada pelas ferramentas atuais. Um dos exemplos é a busca por fórmulas matemáticas, que é exatamente o foco do presente trabalho. Um exemplo sim-

[1] `http://scholar.google.com/`
[2] `http://www.wolframalpha.com/`

ples pode ser demonstrado analisando novamente a ferramenta *Wolfram Alpha*. Se o usuário informar uma função matemática (por exemplo $f(x) = 1/x$), são apresentadas algumas informações pertinentes, como o gráfico e a derivada da função. Para a entrada $\delta = b^2 - 4ac$ é apresentada apenas uma representação alternativa $4ac - b^2 + \delta = 0$. Se a entrada for trocada para $\Delta$ em maiúscula, a saída é a mesma, conforme exibida na Figura 1.1.

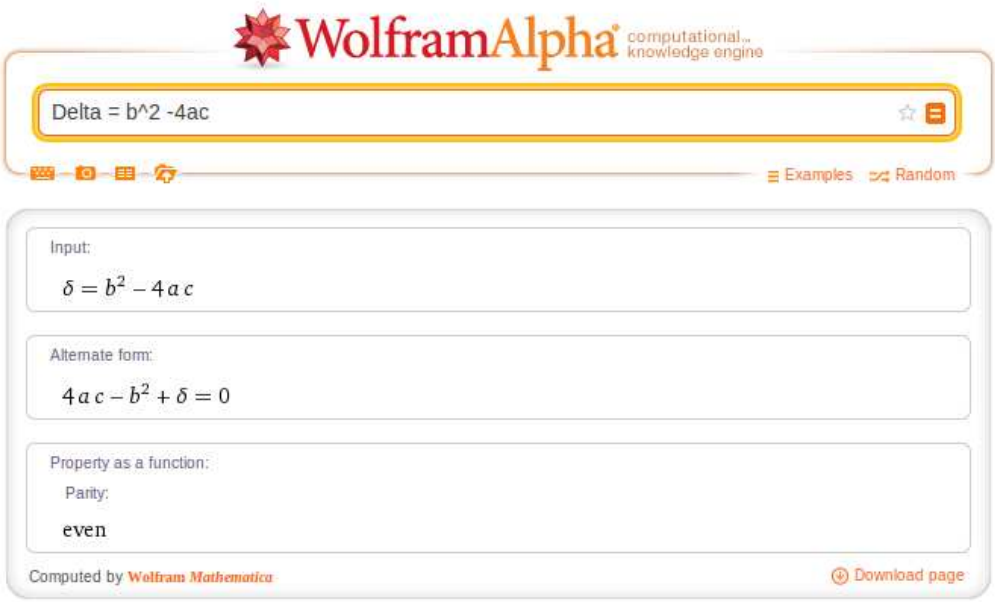



Figura 1.1: Resultado exibido pelo *Wolfram Alpha* (acesso em 10/02/2013).

Com base nesse simples exemplo mostrado, pode-se acrescentar pelo menos mais duas possíveis necessidades de um usuário no que se relaciona a fórmulas matemáticas: *i*) com base em uma fórmula encontrar uma página ou documento que ajude no entendimento da mesma; *ii*) encontrar aplicações diferentes para uma fórmula deduzida em um determinado estudo (vários exemplos de aplicação de fórmulas podem ser encontrados na seção *Optimization*[3]). Por essa carência não ser resolvida pelas ferramentas de busca textuais, começam a surgir ferramentas cujo objetivo é encontrar páginas que contenham determinada fórmula (informada pelo usuário) ou outras semelhantes. No passado já foram feitas propostas como a *Math Web Search*[2] e a *Whelp*[6]. Elas serão apresentadas no Capítulo 2. No entanto, nenhuma delas se encontra mais operacional, inviabilizando testes práticos, como por exemplo a medição da qualidade dos resultados dada uma determinada busca, ou a ordem de ranqueamento ou ainda o tempo de resposta. Mais recente, a ferramenta *Symbolab*[7] foi criada com esse mesmo intuito (lançada aparentemente em Outubro de 2012). Contudo, parece não existir artigo científico relacionado a ela, o que dificulta o aprofundamento na metodologia. Porém, através de alguns

[3] `http://mathworld.wolfram.com/topics/Optimization.html`

testes feitos, já foi possível captar um pouco de como ela funciona. Ela será também melhor discutida no Capítulo 2. Até a data de conclusão desse trabalho, essa foi a única ferramenta funcional online encontrada que se propõe a desempenhar uma busca semelhante à aqui proposta. Dessa forma, os testes de busca no Capítulo 5 serão também baseados nela.

Ferramentas de busca por fórmulas matemáticas possuem diversas aplicações a curto, médio e longo prazo. Além das já citadas de recuperar páginas que ajudem no entendimento de uma fórmula, ou a visualizar aplicações semelhantes, estima-se ainda que a médio e longo prazo, algoritmos poderão propor soluções para problemas matemáticos ainda não resolvidos com base em estudos já desenvolvidos[8].

Um exemplo que ilustra a descoberta de soluções com base em estudos já desenvolvidos ocorreu no chamado "Último Teorema de *Fermat*". O matemático *Pierre Fermat* no século $XVII$ (por volta do ano de 1 637) escreveu na margem de um dos seus livros a seguinte afirmação: "Tenho uma demonstração maravilhosa desta proposição, mas a margem deste papel é muito estreita para contê-la". O teorema que *Fermat* alegou ter demonstrado diz que a equação $x^n + y^n = z^n$ não tem soluções com números inteiros para $n > 2$. Passados cerca de 357 anos (em 1 994), o matemático *Andrew Wiles* percebeu que se uma importante conjectura apresentada por dois matemáticos fosse verdadeira, o teorema de *Fermat* também seria. A então conjectura de *Taniyama-Shimura*, hoje conhecida como teorema de *Shimura-Taniyama-Wiles* (ou ainda teorema da modularidade) foi utilizada por *Andrew Wiles* na formulação do teorema de *Wiles*, utilizado na demonstração do "Último Teorema de *Fermat*".[9].

O presente trabalho apresenta como principal resultado uma metodologia nova que permite a busca por fórmulas matemáticas (explicada no Capítulo 5). A ferramenta aqui apresentada denominada *SearchOnMath*[4] tem então como objetivo encontrar páginas que contenham fórmulas semelhantes a uma determinada pelo usuário. Nessa primeira versão *alfa* a ferramenta realiza a busca específica nas seguintes bibliotecas:

- *Wikipedia* (somente a parte matemática)[5]
- *MathWorld*[6]
- *DLMF* (*Digital Library of Mathematical Functions*)[7]

A parte matemática da *Wikipedia* é a maior dentre as três bibliotecas citadas, com 37 723 páginas. A biblioteca *MathWorld* é mantida pela *Wolfram*, e é a segunda

[4] `http://searchonmath.com/`
[5] `http://en.wikipedia.org/wiki/Portal:Mathematics`
[6] `http://mathworld.wolfram.com/`
[7] `http://dlmf.nist.gov/`

maior do trabalho, com 15 095 páginas. A *DLMF* é um projeto do *NIST* (*National Institute of Standards and Technology*) cujo objetivo é revisar e disponibilizar o livro [10] em formato digital, e é a menor das três, com 908 páginas de conteúdo. Mais informações sobre as bibliotecas, como data em que foram obtidas e maneira como as fórmulas foram extraídas estão presentes no Capítulo 3.

Após a obtenção das bibliotecas, um estudo que antecede o desenvolvimento da busca em si foi realizado nos grafos obtidos à partir dos links dessas bibliotecas. Com base na interconexão entre as diversas páginas de conteúdo foram calculadas então dez medidas que caracterizam o conhecimento matemático tanto em aspecto global quanto local. O estudo da estrutura do conhecimento matemático caracterizado segundo medidas clássicas da literatura será exibido no Capítulo 4.

O trabalho encerra no Capítulo 5 com uma avaliação das medidas de ranqueamento (obtidas no Capítulo 4) considerando-se a busca textual booleana nas páginas de conteúdo matemático, e depois com o detalhamento da busca por fórmulas desenvolvida. São destacadas as dificuldades inerentes a esse tipo de busca, e a forma como foram resolvidas. O Capítulo traz ainda o teste da qualidade dos resultados obtidos pela ferramenta proposta *SearchOnMath* e a outra ferramenta disponível *Symbolab*.

Nos anexos são apresentados detalhes extras de cada etapa do trabalho, que aparecem apenas de forma sucinta no texto. Diversas propostas de trabalhos futuros serão ainda apresentadas na Conclusão. Os objetivos e contribuições do trabalho estão sintetizados a seguir.

## 1.1 Objetivos e Contribuições

Os principais objetivos do trabalho, bem como as contribuições do mesmo foram:

### 1.1.1 Objetivos

- Entendimento das estruturas das redes e do impacto em mecanismos de busca;
- Desenvolvimento da ferramenta de busca;

### 1.1.2 Contribuições

- Estudo das estruturas das redes das bibliotecas matemáticas;
- Desenvolvimento da busca textual no domínio da matemática;
- Desenvolvimento da busca por fórmulas;

## 1.2 Organização da Tese

A tese segue organizada da seguinte forma. No Capítulo 2 são apresentadas propostas semelhantes de ferramentas já feitas na literatura. No Capítulo 3 é mostrada uma visão geral das bibliotecas que o estudo abrange. O Capítulo 4 exibe a metodologia e os resultados obtidos no estudo estrutural das bibliotecas, destacando medidas globais e locais nos grafos obtidos. O Capítulo 5 traz os resultados da busca textual, com destaque para a eficiência das medidas de ranqueamento na ordenação das páginas dos domínios; bem como a teoria por trás da busca matemática, o teste desenvolvido, e os respectivos resultados. As conclusões são apresentadas no Capítulo 6, juntamente com diversas propostas de trabalhos futuros.

# Capítulo 2

# Ferramentas de busca relacionadas

A busca por fórmulas matemáticas têm sido foco de pesquisas nos últimos anos. Além dos esforços da equipe do *NIST* nesse sentido [1], outros trabalhos semelhantes vêm sendo (ou foram) realizados, dentre os quais pode-se citar três ferramentas como referências, sendo duas mais antigas: *Math Web Search*[1] e a *Whelp*[2]; e uma mais recente: *Symbolab*[3]. Sobre as duas primeiras, nenhuma delas funciona desde 2010, sendo que a primeira citada ainda possui página no ar, enquanto que na segunda, o endereço citado já não se encontra mais disponível (serão apresentadas telas das mesmas, obtidas em 2010). A terceira ferramenta é a única funcional encontrada e com proposta semelhante. No entanto, parece não haver artigo disponível sobre ela na data da realização do trabalho. Assim, sobre as ferramentas mais antigas, foi possível abordar um pouco sobre as informações fornecidas pelos artigos, mas sem um critério prático de comparação por já não funcionarem. Sobre a mais recente, a situação se inverte. Só se pode fazer uma análise prática e alguns testes, mas sendo bastante difícil alguma comparação em nível metodológico. Assim, esse capítulo se divide em duas seções principais, sendo na primeira abordados os aspectos metodológicos das ferramentas antigas, enquanto que na segunda são mostrados aspectos práticos da ferramenta mais recente.

## 2.1 Metodologias

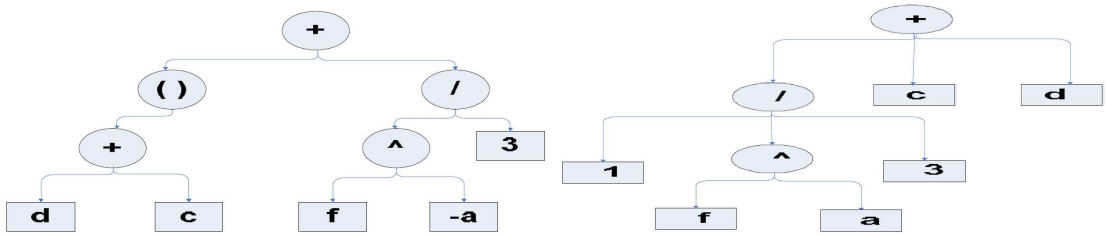
Esta seção se divide em três subseções: *NIST*, *Math Web Search* e *Whelp*. A primeira explica a metodologia teórica proposta em [1] e tem esse nome em virtude de um dos autores compôr o quadro atual de desenvolvedores da biblioteca *DLMF*, que é mantida pelo órgão. Vale destacar contudo que a biblioteca em si não possui a busca por fórmulas e, portanto, não emprega na prática os conceitos propostos no artigo.

[1] `http://search.mathweb.org/`
[2] `http://helm.cs.unibo.it/whelp/`
[3] `http://symbolab.com/`

A segunda e terceira subseções apresentam as metodologias apresentadas nos artigos das bibliotecas *Math Web Search* e *Whelp*.

### 2.1.1 *NIST*

A metodologia proposta em [1] para a realização da busca consiste primeiramente na construção de uma árvore sintática abstrata (geralmente referenciada pela sigla AST) normalizada da expressão que o usuário deseja buscar. Exemplificando, à partir da expressão (d+c)+fˆ-a/3 inicialmente a árvore exibida na Figura 2.1 *(a)* seria obtida. Após a obtenção dessa árvore então alguns passos de normalização são propostos na fórmula, até a obtenção da expressão final, mostrada na Figura 2.1 *(b)*.




(a) Árvore de expressão inicial. (b) Árvore de expressão final.

Figura 2.1: *Normalização proposta no artigo [1].*

Uma vez feita a normalização, o autor sugere que a busca seja feita através da comparação entre a árvore normalizada e as árvores das demais expressões (previamente armazenadas no banco de dados). Contudo, não fica claro qual método seria utilizado na comparação das árvores, uma vez que existem várias possibilidades, sendo que algumas dessas podem ser encontradas nessa referência [11]. Exemplificando, a comparação de árvores consiste basicamente em se verificar quantas operações são necessárias a fim de que uma determinada árvore seja "transformada" na outra, considerando como operações: inserção, remoção ou edição de nós. Por fim, é apresentada no trabalho uma versão básica de gramática que poderia ser usada na interpretação de fórmulas matemáticas. Na gramática proposta por exemplo, considera-se multiplicações ocorrendo sempre com o uso do operador ($*$), fator esse que geralmente não ocorre quanto se extrai fórmulas de páginas da Internet (onde a multiplicação é representada de forma implícita, ou seja, sem operador).

Os passos necessários para a obtenção das ASTs são descritos na Figura 2.2. A partir então desses passos, é possível acrescentar ainda ao trabalho [1] mais algumas possibilidades não exploradas no mesmo, e que podem ter sua eficiência testada na busca por fórmulas. Na explicação das possibilidades, considere a AST $a$ como sendo a árvore obtida a partir da fórmula que se deseja buscar, e uma outra AST $b$ como sendo a árvore resultante de uma determinada fórmula obtida do banco de dados, e

na qual deseja-se saber se são semelhantes.

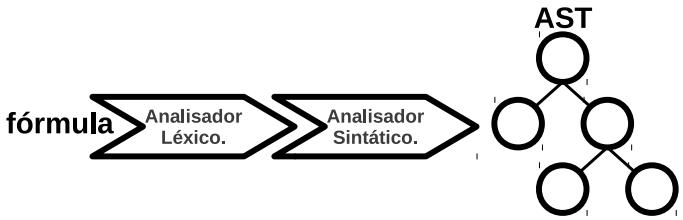



Figura 2.2: Passos para a obtenção de uma árvore sintática abstrata (AST).

1. Percorre-se $a$ em alguma ordem (pré, in ou pós-ordem), obtendo como resultado uma *String* composta pelos *tokens* da expressão de acordo com a ordem escolhida. Supondo que a árvore $b$ do banco já tenha sido percorrida na ordem escolhida, e que o resultado esteja também armazenado, o problema poderia ser reduzido à comparação de *Strings* no que diz respeito ao custo necessário para se tornar a *String* obtida em $a$ na *String* resultante de $b$. Quanto menor esse custo, mais semelhantes são as fórmulas comparadas;

2. Não chegar à fase onde se realiza a análise sintática (mostrada em 2.2). Ao invés disso, obter a partir do analisador léxico a lista dos *tokens* na ordem em que aparecem na fórmula $a$, e comparar essa *String* com outra obtida de forma similar para $b$, de maneira semelhante ao método explicado no item 1.;

Como exemplo dos métodos sugeridos, considere a seguinte expressão $x+y\hat{}2$. A Figura 2.3 mostra as diferenças entre se obter a *String* dos *tokens* a partir da AST ou diretamente da análise léxica. Para isso, deve-se considerar que os seguintes *tokens* são retornados na expressão:

- VAR: variáveis $x$ e $y$;
- POW: operador de potência $\hat{}$;
- CON: constante 2;

Nesse ponto vale destacar o seguinte: parece não existir na literatura ferramenta funcionando que utilize nenhum dos três métodos citados (comparação de ASTs, comparação de *Strings* de *tokens* construídas de ASTs, comparação de *Strings* de *tokens* construídas da fase de análise léxica). O presente trabalho é desenvolvido então em cima do método 2. da Figura 2.3, que consiste portanto na comparação de *Strings* obtidas do analisador léxico. A escolha se deu em função de ser o mais simples (não apenas de desenvolver, como também em complexidade) dentre os três. Descobrir se esse método já retorna resultados consistentes constitui uma

base essencial na literatura para que os outros possam ser propostos, testados e comparados futuramente. Os aspectos do desenvolvimento da ferramenta, bem como a forma como as *Strings* são comparadas serão apresentados em detalhes no Capítulo 5.

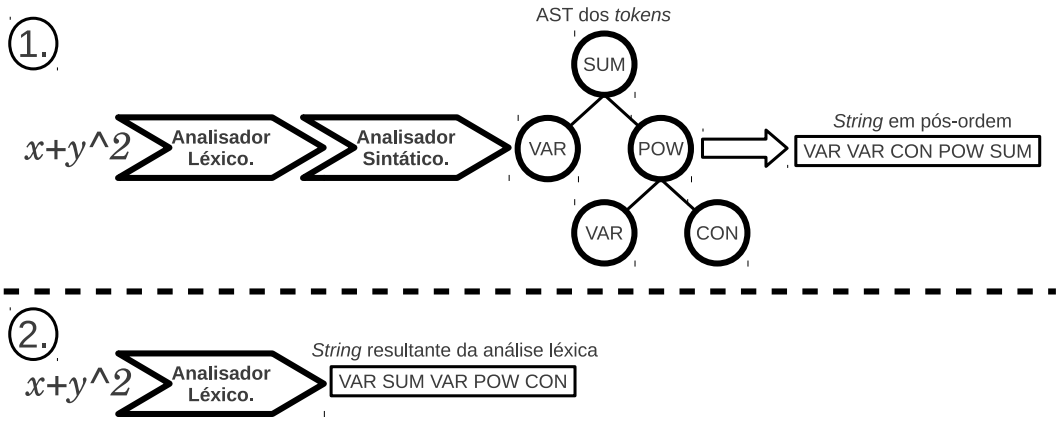



Figura 2.3: Comparação entre *Strings* obtidas em diferentes etapas.

Alguns artigos vieram depois das referências de Youssef [1, 8], inclusive mencionam se basear nos aspectos explicados pelo autor, podendo-se citar [12]. Em paralelo, nessa época surgiram alguns outros trabalhos que propõem metodologias semelhantes [13], e até requisitos levantados com usuários [14]. Alguns dos requisitos mencionados nesse trabalho foram considerados no desenvolvimento da ferramenta proposta e estão também presentes no Capítulo 5.

### 2.1.2 Math Web Search

No trabalho que originou a *Math Web Search* [2] foi utilizado como repositório o *CONNEXIONS Project* [4], que consiste em um ambiente colaborativo na Internet que disponibiliza diversos tipos de conteúdo como cursos e livros. O tipo de conteúdo é bastante abrangente (não sendo fechado ao domínio matemático), e de diferentes níveis (alcançando usuários em diversas faixas etárias).

A ferramenta desenvolvida possuía uma base de dados com mais de 3 400 artigos, e cerca de 53 000 termos (ou 77 000 se forem considerados os subtermos). O projeto incluiu ainda um índice contendo 87 000 fórmulas em formato *MathML*, obtidas na página *functions* do *Wolfram* [5]. Todos esses dados foram retirados do artigo da ferramenta. A tela inicial da ferramenta é exibida na Figura 2.4.

A ferramenta de busca foi então construída em torno de uma técnica indexação chamada árvore de substituição. Uma substituição $\varphi$ é um mapeamento de símbolos matemáticos ($v_i$) para termos ($t_i$), e geralmente é representado como

[4] `http://cnx.org/`
[5] `http://functions.wolfram.com/`

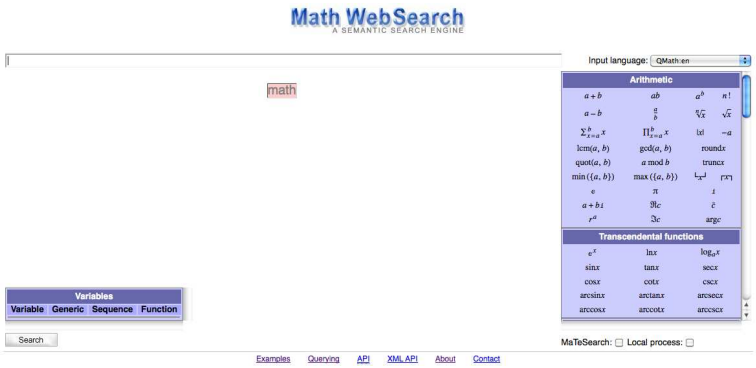



Figura 2.4: Tela inicial da *MathWebSearch* (acesso em 03/2010).

$\{v_1 \rightsquigarrow t_1, \ldots, v_n \rightsquigarrow t_n\}$ com $v_i \in V$ e $t_i \in T$, onde $V$ é o conjunto de símbolos matemáticos e $T$ o conjunto de termos. Uma árvore de substituição é então uma árvore, onde as substituições são os nós. Um termo é obtido após se realizar sucessivas substituições no decorrer do caminho na árvore. Ao se alcançar uma folha da árvore, tem-se então, um termo construído [15]. Logo, nos nós folha dessa árvore já se encontra diretamente a expressão desejada. Para ficar mais fácil a compreensão, é exibido na Figura 2.5 um exemplo da árvore. Nesse exemplo o índice criado engloba os termos $h\left(f\left(z,a,z\right)\right)$, $x$, $g\left(f\left(z,y,a\right)\right)$, $g\left(f\left(7,z,a\right)\right)$, e $g\left(f\left(7,z,f\right)\right)$. Internamente em cada nó são mostradas as referências para a substituição de modo a expressão ser obtida. Por exemplo, a expressão $g\left(f\left(7,z,a\right)\right)$ seria encontrada percorrendo o seguinte caminho: $@0 -> g\left(f\left(@1,@2,@3\right)\right); @3 -> a; @1 -> 7, @2 -> x$.

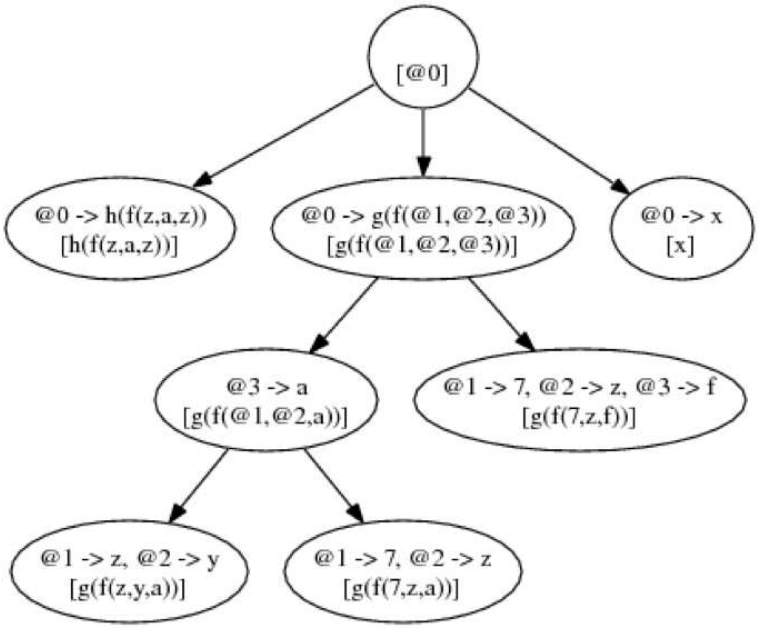



Figura 2.5: Árvore de substituição [2].

Conforme pode-se verificar, uma subexpressão não casaria nesse método, uma vez que as expressões precisam coincidir desde o nó raiz. Por exemplo, considere

a seguinte fórmula $\sum_{k=1}^{\infty} \frac{1}{k^s} = \prod_{p \text{ prime}} \frac{1}{1-p^{-s}}$. Caso a mesma tenha sido colocada tal qual é apresentada na árvore de substituição, e um usuário informasse apenas a segunda parte $\prod_{p \text{ prime}} \frac{1}{1-p^{-s}}$, a ferramenta não encontraria, pois à partir do nó raiz da árvore não se encontra nó que case com a segunda parte da fórmula apenas. Para corrigir esse problema os autores citam que são criadas árvores também para as subexpressões.

A ferramenta *Math Web Search* foi desenvolvida recuperando fórmulas representadas em linguagem *MathML*, um formato *XML* criado para representar expressões matemáticas, que será ainda apresentado no Capítulo 3. Dessa forma, a linguagem usada internamente na indexação e consulta é uma variação de *MathML* também. Isso é um fato interessante porque dentre as bibliotecas que o presente trabalho abrange, apenas a *DLMF* possui representação nessa linguagem, e é justamente a menor das três. A *Wikipedia* possui representação apenas em TEX, enquanto que o *MathWorld* representa suas expressões em uma linguagem própria. Esses detalhes serão também explicados no Capítulo 3.

### 2.1.3 Whelp

A *Whelp* [6] é um protótipo de uma ferramenta que foi desenvolvida para buscar por expressões na biblioteca de conteúdo matemático do *Coq* [6], que é um gerenciador formal de provas. O *Coq* provê uma linguagem formal para a escrita de definições matemáticas, algoritmos executáveis e teoremas, juntamente com um ambiente interativo para o desenvolvimento de provas matemáticas. A *Whelp* utilizava uma linguagem própria do *Coq* semelhante ao TEX e de um conjunto de metadados na indexação das expressões. De forma semelhante à *Math Web Search*, essa ferramenta também não se encontra mais disponível online, o que impossibilita um estudo mais aprofundado entre as técnicas documentadas no artigo e o funcionamento prático. A tela inicial que era usada pela *Whelp* é exibida na Figura 2.6.

Para realizar a busca na ferramenta, 4 tipos diferentes de consulta foram definidos no artigo [6]:

- *Match*: Realiza a busca através da relação explícita entre os termos obtidos na expressão e os termos indexados na ferramenta;
- *Hint*: É um assistente de prova automatizado, onde dada uma hipótese, a ferramenta tenta encontrar teorema(s) para prová-la;
- *Elim*: Usado para provar um resultado por indução sobre um tipo $t$;
- *Locate*: Realiza a busca por palavra-chave;

[6] `http://coq.inria.fr/`

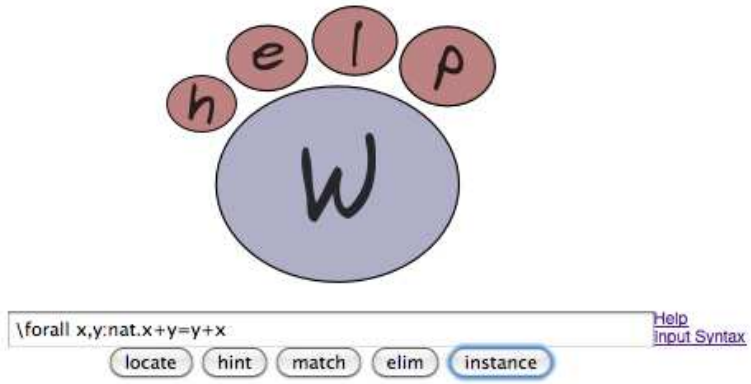



Figura 2.6: Tela inicial da *Whelp* (acesso em 03/2010).

Além dos 4 tipos apresentados, a interface possuía ainda um quinto tipo *Instance*, não explicado no artigo, não sendo portanto possível prover mais informações à respeito.

A opção *Match*, que parecia ser a que mais se aproxima do trabalho proposto, funcionava extraindo das fórmulas metadados, que traziam as seguintes informações: símbolos e respectivas posições em que ocorriam nas fórmulas. Por exemplo, para a fórmula: $\forall m, n : nat.m \leq n \rightarrow m < (S\ n)$. Os metadados extraídos e as respectivas posições seriam os seguintes, mostrados na Tabela 2.1:

Tabela 2.1: Símbolo e posição dos elementos na fórmula.

| Símbolo | Posição |
|---|---|
| $nat$ | MH |
| $\leq$ | MH |
| $<$ | MC |
| $S$ | C |

Onde MH e MC representam a posição superficial e mais aprofundada respectivamente, e C representa ser um símbolo presente na conclusão. A busca então ocorria buscando por fórmulas que tivessem metadados semelhantes localizados em posições parecidas à fórmula informada. Outro detalhe que ocorre no artigo, de forma semelhante ao trabalho de Youssef, as multiplicações aqui também aparecem sempre de forma explícita.

## 2.2 Aspectos Práticos

Conforme já citado, a única ferramenta funcional encontrada no momento da escrita do trabalho é a *Symbolab*, e será com base no funcionamento da mesma que os testes de medição da qualidade dos resultados serão realizados.

### 2.2.1 Symbolab

A ferramenta aparentemente foi lançada em Outubro de 2012 (data da primeira postagem no blog). A tela inicial da ferramenta é exibida na Figura 2.7.


Figura 2.7: Tela inicial da *Symbolab* (acesso em 03/2013).

As bibliotecas e páginas suportadas pela ferramenta podem ser vistas no blog[7] da mesma. Não é possível fazer considerações mais aprofundadas sobre a ferramenta, uma vez que a metodologia da mesma não está disponível. Um aspecto que chama atenção está no fato de que a mesma realiza busca em sites onde o conteúdo disponível é vídeo. Não fica claro se o conteúdo do vídeo é retornado com base no título ou metadados. Muitos documentos em formato *pdf* também aparecem como resultados.

Um aspecto importante observado é que a ferramenta retorna resultados diferentes para buscas semelhantes, como $x^2$ e $y^2$. Verificou-se que para $x^2$, os resultados apresentados na primeira página possuem de fato $x^2$, mas não expressões que apresentassem uma variável qualquer ($y$ ou $z$ por exemplo) elevada ao quadrado (ou elevada a uma outra constante). O mesmo se observa para a busca de $y^2$. Esta observação é um indício de que se uma pessoa deduzir uma fórmula usando notação própria e desejar fazer a busca, os resultados retornados talvez não tenham boa qualidade, dada a forte relação com notação que a ferramenta parece manter.

Apesar das poucas informações disponíveis, foi possível a realização de testes com o objetivo de medir a qualidade dos resultados retornados pela *Symbolab* em comparação aos retornados pela ferramenta proposta *SearchOnMath*. Os detalhes do teste serão exibidos no Capítulo 5.

[7] `http://blog.symbolab.com/2012/10/symbolab-sources-what-makes-symbolab-so.html`

# Capítulo 3

# Bibliotecas Estudadas

Neste capítulo serão apresentados os aspectos que compõem as bibliotecas *Wikipedia* (parte matemática), *MathWorld* e *DLMF*, foco do presente trabalho. As três foram escolhidas por estarem entre as principais fontes de conteúdo matemático. Em todas elas as páginas de conteúdo técnico são alcançadas à partir da navegação em páginas de categorias organizadas de maneira hierárquica.

## 3.1 *Wikipedia*

Na *Wikipedia* conforme já citado, foi considerada apenas a parte matemática da versão em inglês. Para uma página ser considerada na parte matemática, ela precisa estar em pelo menos uma das suas categorias[1]. Para a obtenção somente da parte matemática no entanto, é necessário que todo o download da *Wikipedia* seja realizado, para depois ser extraída somente essa parte. A versão utilizada no trabalho foi de um *dump* feito em Setembro de 2010, cujo link já não se encontra mais disponível. A seção a seguir detalha exatamente esses passos. Os nomes dos arquivos descritos foram atualizados com base na versão mais recente da *Wikipedia*.

### 3.1.1 Obtenção da Parte Matemática da *Wikipedia*

A obtenção da parte matemática começa, conforme já foi dito, com o download da *Wikipedia* toda. Os *dumps* mais recentes estão disponíveis em `http://dumps.wikimedia.org/enwiki/`. O usuário precisa estar atento a dois detalhes importantes nesse passo: *i*) nem todos os *dumps* que aparecem na lista estão concluídos (alguns aparecem no início como *dump in progress*); *ii*) como toda a *Wikipedia* será importada, a partição em disco que contém o banco de dados precisa ser cuidadosamente dimensionada, ou a importação causará erro de falta de espaço depois de dias executando. No trabalho, um computador com Linux *Debian* 5 e 500GB

[1] `http://en.wikipedia.org/wiki/List_of_mathematics_categories`

de disco rígido foram suficientes. Foi feita uma instalação minimalista, deixando o maior espaço possível para a partição Var, onde os dados da importação foram armazenados pelo SGBD (*MySQL*).

Os arquivos necessários com os respectivos tamanhos são listados a seguir:

- pages-articles.xml.bz2 (9, 2 GB);
- category.sql.gz (21, 7 MB);
- categorylinks.sql.gz; (1, 2 GB)
- pagelinks.sql.gz (4, 8 GB);

O esquema completo das tabelas da *Wikipedia* com explicação detalhada sobre cada uma delas pode ser acessado em `http://www.mediawiki.org/wiki/Manual:Database_layout`. Conforme pode-se verificar, a biblioteca toda utiliza algumas dezenas de tabelas. Ao se extrair e importar os arquivos acima citados, algumas dessas tabelas serão preenchidas. Na importação da parte matemática é necessária a manipulação das seguintes (os conteúdos e informações pertinentes estão detalhados):

- *Page*: informações como o título e o *namespace*[2] das páginas. O campo *namespace* é o que permite diferenciar páginas com o mesmo nome, mas diferentes funções. Por exemplo, existe a página *Mathematics* de conteúdo, e a página *Mathematics* portal. Elas possuirão o mesmo valor no Banco de Dados no campo *page_title*, mas valores diferentes em *page_namespace*;
- *Revision*: metadados para as edições feitas na *wiki*, nesse caso, estão apenas os metadados da última edição em cada caso (uma vez que os arquivos com as edições em si não foram baixados). Essa tabela é a ligação entre a tabela *Page* e a tabela *Text*, próxima a ser detalhada;
- *Text*: Contém os textos em si das páginas, armazenados no campo *old_text*;
- *Category*: todas as categorias da biblioteca estão aqui;
- *Categorylinks*: links entre as categorias e as páginas, nos campos *cl_to* e *cl_from* respectivamente;
- *Pagelinks*: links entre as páginas;

[2] `http://en.wikipedia.org/wiki/wikipedia:Namespace`

Com base nessas explicações os algoritmos foram então desenvolvidos. Em alto nível, os passos seguidos são os seguintes:

A extração da parte matemática começa com a seleção da página com título List_of_mathematics_categories e *namespace* 0 na tabela *Page*. À partir daí obtém-se o id da revisão no campo *page_latest* (chave estrangeira para a tabela *Revision*). Depois, na tabela *Revision* obtém-se o campo *rev_text_id* (chave estrangeira para a tabela *Text*). Na tabela *Text* finalmente se consegue o texto da página de categorias. Nesse momento então, extrai-se do *HTML* todos os títulos das categorias. Para cada título de categoria, através da manipulação das tabelas *Category* e *Categorylinks* obtém-se os ids das páginas contidas nela.

No trabalho foi armazenada também a relação entre categorias, isso é, quais as subcategorias estão presentes dentro de uma determinada categoria, montando assim uma árvore de relações. A obtenção dessa relação não está presente diretamente em uma tabela, mas no *HTML* das páginas de categoria. Exemplificando, se uma categoria B está contida dentro de uma outra A, então, dentro do *HTML* de B terá links da forma: [[Category:A]] ou [[Category:A|sortkey]]. Mais um detalhe importante aqui: na tabela *Page*, o espaço em um título é preenchido com _. Por exemplo, *Number Theory* aparecerá como *Number_Theory*. No link dentro do *HTML* das páginas de categoria no entanto, os espaços não são preenchidos. Assim, uma categoria que esteja dentro de *Number Theory* terá o link da forma: [[Category:Number Theory]].

Uma visão mais detalhada incluindo pseudocódigo e estrutura de tabelas proposta na importação foi colocada no Apêndice A.

### 3.1.2 Organização do conteúdo

O conteúdo obtido da *Wikipedia* aparece portanto organizado em categorias, a partir das quais é possível navegar em níveis de forma semelhante a uma árvore. Ao todo existem armazenados no Banco de Dados 1 388 categorias, com a maior profundidade igual a 6. Uma curiosidade que foi observada no *dump* extraído, e que ainda continua presente na atualidade está na página *List_of_mathematics_categories*, que é dividida em três seções: *i*) *Mathematics categories*; *ii*) *Mathematicians categories*; *iii*) *Mathematics-related categories*. Quatro categorias aparecem tanto em *Mathematics categories* quanto em *Mathematics-related categories*. É interessante porque pode levar ao questionamento se essas categorias são Matemáticas, ou se são relacionadas à Matemática. As categorias são:

- *Descriptive complexity*: Possui dentro 9 páginas e está contida dentro da categoria *Computational complexity theory*;

- *Laymen and Statistics*: Possui dentro uma subcategoria (*Statistics education*), 8 páginas e está contida dentro da categoria *Statistics*;

- *Mathematics templates*: Possui dentro 10 subcategorias, 133 páginas e está contida dentro de duas categorias (*Mathematics and abstraction templates* e *Mathematics Wikipedia administration*);

- *Statistical dependence*: Possui dentro duas subcategorias (*Covariance and correlation* e *Inter-rater reliability*), 42 páginas e está contida dentro da categoria *Statistics*;

Das quatro categorias relacionadas, *Descriptive complexity* parece ser categoria relacionada à Matemática, por possuir páginas de conteúdo ligados à Computação. *Laymen and Statistics* e *Statistical dependence* possuem a maioria das páginas de fato dentro da Estatística, e poderiam portanto aparecer como categorias Matemáticas. A *Mathematics templates* por sua vez traz *templates* (ou modelos) relacionados à navegação nas páginas da Matemática e páginas relacionadas. Talvez devesse ser a única presente em ambas as seções (*Mathematics categories* e *Mathematics-related categories*) da página *List_of_mathematics_categories*.

### 3.1.3 Representação das Expressões Matemáticas

As fórmulas mostradas nas páginas em formato *png* possuem representação textual dentro das mesmas. As equações no *dump* obtido aparecem dentro do texto em *tags* <math></math> e em formato TEX[16]. Assim, foi desenvolvido um algoritmo que percorre o código *HTML* recuperando esse conteúdo através de expressão regular. Ao final foram extraídas 208 604 equações.

## 3.2 *MathWorld*

Desenvolvida por Eric Weisstein e lançada no ano de 1995, a *MathWorld* se tornou referência tanto na área de educação quanto entre os matemáticos. A versão utilizada no trabalho foi obtida em Agosto de 2009 através do software *HTTrack*[3][17].

Um detalhe interessante é que a *MathWorld* detecta e bloqueia *WebCrawler's* que façam o download a uma taxa muito elevada. Durante o processo de download, caso ocorra essa detecção, o site continua enviando as páginas sem nenhum problema aparente (o código HTTP de retorno continua sendo 200 OK [18]). No entanto, quando a página é aberta, o conteúdo dela apresenta o texto *"Access Denied for IP..."* indicando que o IP utilizado foi bloqueado. À partir desse ponto, qualquer

[3] `http://www.httrack.com`

página da biblioteca que se tente o acesso retornará dessa forma. Mesmo com o passar de dias o IP permanece bloqueado. A obtenção da biblioteca então precisou ser feita com o *WebCrawler* executando a taxas muito baixas. No trabalho [19] o autor apresenta um estudo sobre a estrutura da *MathWorld*, onde ele cita ter obtido a biblioteca em Dezembro de 2008 (8 meses antes da versão obtida no presente trabalho), com 12 000 páginas (aproximadamente 3 000 páginas a menos quando comparada com o base desse trabalho). Pode-se imaginar que a biblioteca não tenha tido um crescimento tão alto em tão pouco tempo, indicando talvez o mesmo problema com a obtenção das páginas citadas acima.

Pelo menos duas configurações precisam ser feitas no *HTTrack*: definir a quantidade de conexões igual a cerca de uma a cada 20 segundos (tentativas com uma conexão a cada 10 segundos já resultam em bloqueio), e mudar a identidade do navegador, que por padrão o identifica como sendo um *WebCrawler*. A biblioteca após ter sido obtida ficou com 15 095 páginas de conteúdo, e seus arquivos *HTML* compactados com aproximadamente de 230 MB.

### 3.2.1 Organização do conteúdo

Na *MathWorld* as páginas de conteúdo matemático são organizadas e classificadas dentre uma das grandes categorias mostradas na Figura 3.1.

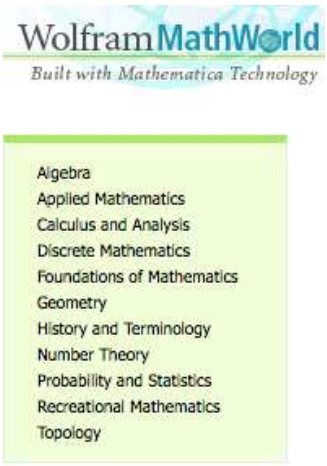



Figura 3.1: Menu contendo as grandes categorias.

Internamente em cada uma das grandes categorias existe uma estrutura em árvore, até que nos nós folha obtém-se as páginas de conteúdo. Ao todo existem armazenadas no Banco de Dados 864 categorias A Figura 3.2 mostra as subcategorias contidas em *Algebra*. Os números dentro dos parênteses mostram a quantidade de itens presentes dentro da categoria. Os itens podem ser subcategorias ou páginas de conteúdo, sendo que não acontece de dentro de uma categoria ocorrer ambas as

possibilidades (páginas de conteúdo e subcategorias). Assim, o conteúdo de cada categoria é um *ou exclusivo* entre essas duas possibilidades.


Figura 3.2: Subcategorias em *Algebra*.

As páginas de conteúdo trazem no seu cabeçalho os caminhos nos quais as mesmas são classificadas à partir do menu das grandes categorias. Conforme pode-se observar na figura 3.3, a página *Sum* pode ser alcançada a partir das grandes categorias *Algebra* e *History and Terminology*.


Figura 3.3: Classificações da página *Sum*.

Um detalhe importante é que algumas podem possuir caminhos a serem percorridos nas categorias, mas que não são listados no cabeçalho das páginas de conteúdo. Como exemplo:

*Algebra*>*Coding Theory*@ >*Code*

Nesse exemplo, *Code* é a página final de conteúdo. Apesar de ser possível acessá-la à partir de *Algebra*, o único caminho exibido no cabeçalho da página é:

*Discrete Mathematics*>*Coding Theory*>

A questão está justamente no @ apresentado no menu de *Algebra*, que pode ser visualizado na figura 3.2. Sempre que uma subcategoria for listada com o símbolo @, isso indica que aquela subcategoria está mais contida em uma outra categoria, mas possuindo uma menor influência na categoria onde a mesma é listada com @. No exemplo, *Coding Theory* é uma subcategoria que está mais relacionada com a categoria *Discrete Mathematics* do que com a *Algebra*.

A Figura 3.4 exibe uma pequena parte da estrutura da *MathWorld*, mas que já permite a análise de algumas características inerentes à biblioteca. Os círculos representam as páginas de categorias e subcategorias. Os quadrados representam páginas

finais. Analisando a estrutura das categorias e subcategorias, pode-se perceber o formato de árvore com as páginas de conteúdo estando nos nós folha. Ao se observar os links entre as categorias e subcategorias na versão utilizada no trabalho constatou-se que eles ocorrem sempre de um nível menor para um nível maior ou igual. Por exemplo, *Abstract Algebraic Curves* está no nível 3 da árvore, e ela só possui links para categorias de nível maior ou igual a 3. Outro detalhe é que a profundidade da árvore não é a mesma para todas as categorias, sendo a maior profundidade de tamanho igual a mostrada na figura, com a grande categoria *Geometry* descendo em mais 4 níveis em subcategorias até chegar na página *Antipedal Triangle*, que é uma página de conteúdo. Observando a página *Trilinear Coordinates*, observa-se que uma página final pode ser alcançada à partir de caminhos com tamanhos diferentes, estando inclusive dentro de uma mesma categoria. Os nós folha da árvore (páginas de conteúdo) por sua vez, também possuem links entre si. E, por último, o link de *Geometry* para *Algebraic Geometry* ilustra a ocorrência de um link do tipo @, explicado no parágrafo anterior.

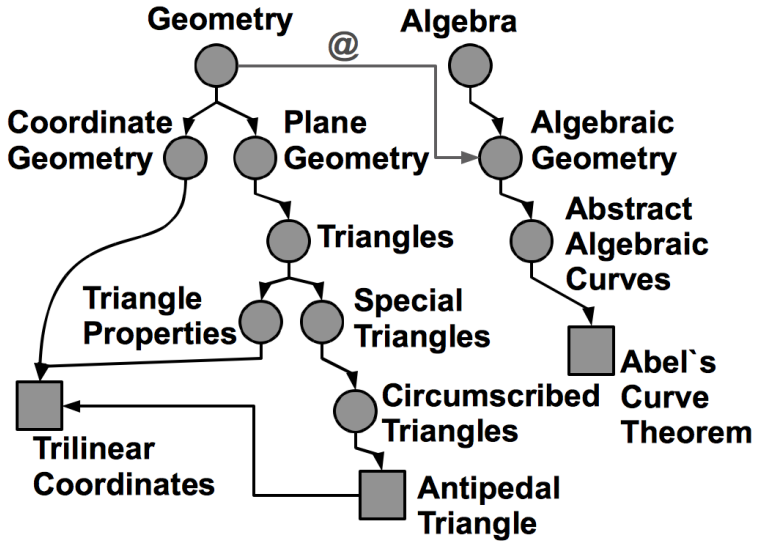



Figura 3.4: Resumo da estrutura da *MathWorld*.

### 3.2.2 Representação das Expressões Matemáticas

Na *MathWorld* as expressões são representadas em imagens no formato *gif*. Porém, todas possuem uma representação alternativa em forma textual, que pode ser obtida no código *HTML* da página. A Figura 3.5 mostra uma parte do código *HTML* da página contendo a representação textual (dentro do atributo *alt* da *tag img*) da equação $P(A \cap B_j) = P(A)P(B_j|A)$ obtida da página do Teorema de Bayes. Um detalhe nessa figura e que acontece com bastante frequência também nas outras bibliotecas é a vírgula ao final da equação. Parte do texto da página, mas que

aparece colocada dentro da *tag* da equação. É comum encontrar no final das equações extraídas vírgulas e pontos finais. Esses elementos textuais foram retirados das equações na busca.

```
<tr><td align="left"><img src="/images/equations/BayesTheorem/NumberedEquation1.gif"
class="numberedequation" width="166" height="20" border="0" alt=" P(A intersection B_j)=P(A)P(B_j|A),
" /></td><td
```

Figura 3.5: Representação da equação no código *HTML* na *MathWorld*.

A ferramenta trabalha então com a representação textual das expressões. Na obtenção das mesmas à partir do código *HTML*, o algoritmo desenvolvido procura por *tags* que possuam o atributo *class* contendo algum dos seguintes valores: *numberedequation*, *inlineformula* ou *displayformula*. Analisando os documentos percebeu-se que sempre que uma fórmula acontece, a mesma está representada dentro de *tags* que contém algum desses três valores no atributo *class*. Ao final foram extraídas 118 377 equações.

## 3.3 *DLMF*

A *NIST Digital Library of Mathematical Functions* foi disponibilizada em sua versão completa em 2010. A versão utilizada no trabalho foi obtida em Setembro de 2010 também através do software *HTTrack*. Os arquivos *HTML*, TEX e *MathML* compactados ficam com cerca 13, 5 MB.

### 3.3.1 Organização do conteúdo

O conteúdo da *DLMF* é organizado de forma semelhante aos capítulos de um livro, com foco na classificação de funções. A Figura 3.6 mostra a exibição dos tópicos iniciais da biblioteca.

A profundidade da *DLMF* também é menor em comparação com a *MathWorld*, estando as páginas de conteúdo acessíveis com profundidade menor ou igual a dois.

### 3.3.2 Representação das Expressões Matemáticas

A *DLMF* também possui representação textual das suas equações representadas em formato *png*. Pode-se perceber equações presentes tanto no decorrer do texto quanto em posições de destaque na página. As equações que aparecem em destaque possuem a representação textual em arquivos separados nos formatos TEX [16] e *MathML presentation* [20]. A Figura 3.7 mostra os formatos de representação para a equação na opção *Encodings*.

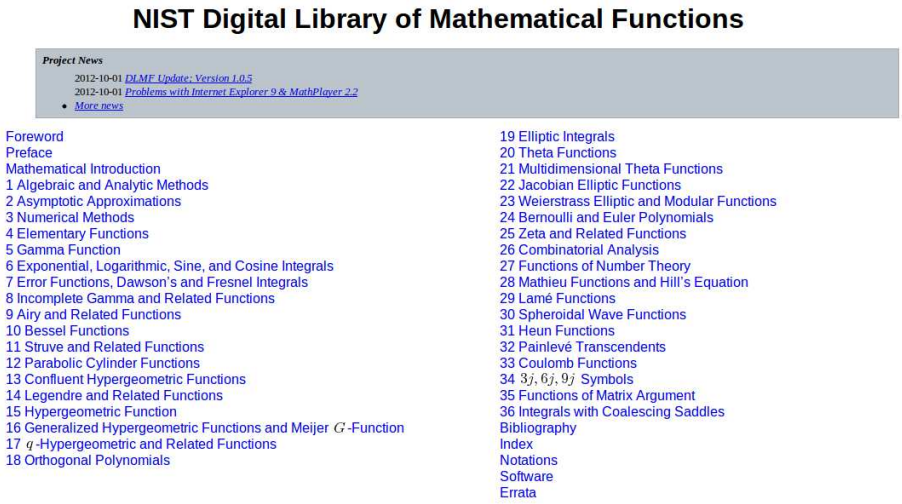



Figura 3.6: Tela inicial exibindo a divisão das seções.

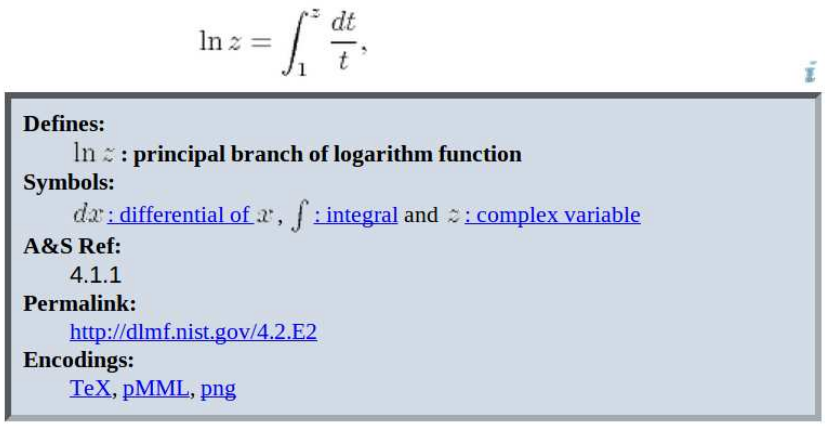



Figura 3.7: Opções de exibição em TEX e *MathML* na *DLMF*.

As equações que aparecem no decorrer do texto possuem seu código em TEX incorporado ao *HTML* da página em um atributo *alt*, e são precedidas pelo atributo *class="math"*. A Figura 3.8 exibe em destaque o atributo.

```
<p class="p">The <em class="emph">general logarithm function</em> <img
src="./4/2/m33.png" width="37" height="16"
alt="\mathop{\mathrm{Ln}\/}\nolimits z" class="math" style="vertical-
align:-2px;"> is defined by</p>
```

Figura 3.8: Representação da equação no código *HTML* na *DLMF*.

É muito importante que uma ferramenta de busca ao oferecer suporte à *DLMF* não ignore as equações no decorrer do texto. Pois foram extraídas da *DLMF* 46 411 equações, dentre as quais 26 718 estavam no texto da página, e o restante (19 693) nos arquivos TEX e *MathML* referenciados. Outro detalhe, constata-se nas equações vírgulas ou ponto no final das mesmas (elementos textuais), além de colchetes \[ \] e no meio \/, que não são renderizados, e foram retirados.

### *MathML*

Das três bibliotecas que a busca abrange, a *DLMF* é a única a oferecer a representação em *MathML*, linguagem essa criada com a finalidade de facilitar a publicação de conteúdo científico e matemático na Internet. A *MathML* foi lançada na sua versão 3.0 em Outubro de 2010, e apresenta como vantagem o fato de permitir a representação das expressões de duas formas: *presentation* e *content*.

A representação em *presentation* foi criada com o intuito de permitir a visualização nos navegadores das expressões em uma elevada qualidade gráfica. Quando escrita em *content*, os aspectos mais importantes da expressão estão em questões semânticas, e não na sua visualização. Vale frisar que apesar do tempo que já se passou desde a primeira formalização presente no site do *W3C* sobre a *MathML* (datada de Abril de 2007), o padrão ainda não é suportado de forma plena nos navegadores mais populares. Esse fato possivelmente é um dos aspectos que faz com que bibliotecas adotem ainda a representação de equações na forma de figuras, com representação textual alternativa nas *tags*.

A seguir estão duas representações em *MathML* para a expressão $b^2 - 4ac$, sendo o código 3.1 escrito em formato *presentation*, onde pode-se verificar por exemplo a *tag* <mrow>, que indica a organização em grupo horizontalizado. As *tags* em *presentation* geralmente começam com a letra *'m'*, e utilizam de *'o'* para operadores, *'i'* para identificadores e *'n'* para números. O código apresentado em 3.2 por sua vez está no formato *content*. Nesse formato, *tags* de aspecto semântico são a principal diferença, como por exemplo <power>.

Código 3.1: Exemplo de *MathML* em *presentation*

```
<mrow>
        <msup>
                <mi>b</mi>
                <mn>2</mn>
        </msup>
        <mo>−</mo>
        <mrow>
                <mn>4</mn>
                <mo>&#x2062;<!−−INVISIBLETIMES−−></mo>
                <mi>a</mi>
                <mo>&#x2062;<!−−INVISIBLETIMES−−></mo>
                <mi>c</mi>
        </mrow>
</mrow>
```

Código 3.2: Exemplo de *MathML* em *content*

```
<apply>
        <minus/>
```

```
3           <apply>
4                   <power/>
5                   <ci>b</ci>
6                   <cn>2</cn>
7           </apply>
8           <apply>
9                   <times/>
10                  <cn>4</cn>
11                  <ci>a</ci>
12                  <ci>c</ci>
13          </apply>
14  </apply>
```

Um último detalhe a ser tratado quando se trabalha na recuperação de equações em formato *MathML presentation* pode ser visto na linha 9 do código 3.1. O operador representado pela codificação *HTML* ⁢ denominado *invisibletimes* faz parte de um grupo que geralmente não é renderizado, mas que agrega informação sobre a fórmula, conforme pode-se perceber. Assim, ao se abrir uma equação que tenha esse código em um editor de texto por exemplo, ele não será exibido. Existem 4 códigos desses não renderizáveis em *MathML*[4]: ⁡, ⁢, ⁣ e &InvisiblePlus;.

[4] `http://www.w3.org/TR/mathml3/mathml.pdf`, página 29. Acesso em 13/02/2013

# Capítulo 4

# Estrutura de Rede das Bibliotecas

Neste capítulo serão apresentadas a metodologia e os resultados obtidos sobre a estrutura da rede das bibliotecas. O estudo aqui exibido antecede o desenvolvimento da ferramenta de busca (explicado no Capítulo 5). O entendimento da organização do conhecimento matemático, e sua caracterização segundo medidas clássicas de redes (graus e centralidade), e de ranqueamento (*HITS* e *PageRank*) fornece um bom norte depois na definição de qual critério adotar na ordenação dos resultados da busca.

## 4.1 Metodologia

No estudo do conhecimento matemático foi obtido para cada uma das bibliotecas um grafo dirigido. Nos grafos então, cada página de conteúdo deu origem a um nó, e arestas representam os links existentes entre as páginas. Em todos os casos, links que tinham como origem e destino a mesma página foram desconsiderados. Logo, não existe nó com aresta para si próprio. De forma semelhante, se uma determinada página $a$ possui dois ou mais links para uma mesma página $b$, apenas uma aresta é considerada.

No caso da *Wikipedia* e da *MathWorld*, links podem ser classificados de duas maneiras quanto à sua localização no texto da página. Eles podem aparecer no corpo do texto, ou na seção *See-also* quando ela existir. Intuitivamente percebe-se que esses links tendem a ter propósitos diferentes. Enquanto que os links no corpo do texto geralmente possuem como objetivo explicar melhor algum termo usado na escrita, sendo portanto referências de acesso rápido antes que a leitura na página continue sendo feita; os links na seção *See-also* indicam de fato páginas onde material relacionado pode ser encontrado. Por essa razão, foram construídos dois grafos distintos tanto para *Wikipedia* quanto para a *MathWorld*. Em um deles se considera todos os links das páginas (links contidos no corpo do texto e na seção *See-also*), enquanto que no outro são considerados apenas os links da seção *See-also*.

No caso da *DLMF* não foi feita nenhuma diferenciação em relação ao local onde os links aparecem nas páginas. Porém, as páginas da *DLMF* possuem seções no texto denominadas *Referenced-by*. Isto é, caso uma página $a$ possua um link referenciando uma seção em $b$, a seção referenciada em $b$ terá um link na forma *Referenced-by* para $a$. Assim, links dessa natureza são redundantes e foram desconsiderados.

Sobre a notação a ser usada nesse e nos próximos capítulos, formalizou-se $n$ como sendo o número de nós e $m$ o número de arestas. Para um nó $i$, $I_i$ é o seu conjunto de vizinhos de entrada (nós nos quais as arestas se originam, e apontam para $i$), e $O_i$ é o conjunto de vizinhos de saída (nós nos quais as arestas originadas em $i$ apontam para). O grau de entrada é representado por $\delta_i^+ = |I_i|$, e o grau de saída por $\delta_i^- = |O_i|$. O número de vizinhos desconsiderando-se a direção das arestas é dado por $\delta_i = |I_i \cup O_i| \leq \delta_i^+ + \delta_i^-$. Fica claro portanto que $\max\{\delta_i^+, \delta_i^-\} \leq \delta_i$. Para quaisquer dois nós $i$ e $j$, $d_{ij}$ é a distância de $i$ até $j$, que é o número de arestas no menor caminho dirigido que conecta os dois nós. Caso não exista caminho entre eles, $d_{ij=\infty}$. Definiu-se ainda $R_i$ como sendo o conjunto de nós $j$ alcançáveis a partir de $i$, de modo que $0 < d_{ij} < \infty$. Observe que $R_i = 0$ se e somente se o nó $i$ for um *sink*, ou seja, $O_i = 0$. A Tabela 4.1 exibe o período em que as bibliotecas foram obtidas, bem como a notação a ser usada para referenciá-las.

Tabela 4.1: Bibliotecas, período de download e notação

| Biblioteca | Período de download | Notação |
|---|---|---|
| *Wikipedia* | Setembro 2010 | $W$ |
| *Wikipedia*, links *See-also* | Setembro 2010 | $W'$ |
| *MathWorld* | Agosto 2009 | $M$ |
| *MathWorld*, links *See-also* | Agosto 2009 | $M'$ |
| *DLMF* | Setembro 2010 | $D$ |

### 4.1.1 Medidas Globais

Foram exploradas seis medidas globais para cada grafo. As duas primeiras são relacionadas à quantidade de nós $n$ e de arestas $m$. A primeira é simplesmente o grau médio de entrada, definido como $\delta^+$ e dado por

$$\delta^+ = \frac{1}{n}\sum_i \delta_i^+ = \frac{m}{n} \tag{4.1}$$

(sendo necessariamente igual ao grau médio de saída). A segunda medida é o grau médio do grafo (desconsiderando-se a direção das arestas). Definido como $\delta$, tem se

$$\delta^+ \leq \delta = \frac{1}{n}\sum_i \delta_i \leq \frac{1}{n}\sum_i \left(\delta_i^+ + \delta_i^-\right) = 2\delta^+ \ . \tag{4.2}$$

Ambos $\delta^+$ e $\delta$ são indicadores da densidade de arestas em relação ao número de nós. O valor de $\delta$, em particular, pode variar dentro dos limites, $\delta^+$ e $2\delta^+$, indicando no primeiro caso que para toda aresta $a \to b$ existe também uma outra $b \to a$ (antiparalela), e no segundo caso onde não ocorre nenhuma aresta antiparalela. Em média então, a fração de $\delta$ que corresponde a pares de arestas antiparalelas é dada por $(2\delta^+ - \delta)/\delta = 2\delta^+/\delta - 1$.

A próxima medida global estudada é a fração $S$ de $n$ correspondente aos nós dentro da maior componente fortemente conexa do grafo (denominada de GSCC, onde o $G$ foi usado para expressar a ideia da maior componente ser gigante). Uma componente fortemente conexa pode ser então composta por único nó, por exemplo $i$, de modo que $i \notin R_j$, para todo nó $j \in R_i$ (não existe caminho de volta que saia de quaisquer dos nós que podem ser alcançados por $i$ no grafo dirigido); ou é um conjunto maximal de nós de acordo com a propriedade que $j \in R_i$ para quaisquer dois elementos $i$ e $j$ tal que $j \neq i$. No segundo caso então, um caminho dirigido existe entre dois nós distintos quaisquer dentro da componente fortemente conexa. Informalmente, o valor de $S$ pode ser considerado como um indicador do "grau de aciclicidade" da rede. Se o grafo for acíclico, então todas as componentes fortemente conexas são compostas por um único nó e $S = 1/n$. O outro extremo corresponde ao caso onde todos os nós estão dentro da $GSCC$, tendo portanto $S = 1$.

A quarta e a quinta medidas globais são ambas relacionadas a classificação do grafo frente ao assim chamado efeito de mundo-pequeno [21, 22]. A primeira delas consiste em se calcular a distância média entre quaisquer dois nós distintos, de modo que somente distâncias finitas são consideradas. Esta medida é definida então como $l$, de modo que

$$l = \frac{1}{N} \sum_i \sum_{j \in R_i} d_{ij}, \tag{4.3}$$

onde $N$ é o número de pares $i, j$ que contribuem no somatório duplo. A segunda delas segue a tendência usual de se desconsiderar a direção das arestas e calcular o coeficiente de clusterização do grafo resultante com base na sua definição mais comum [23]. Sendo $C$ o coeficiente de clusterização, então a sua formulação segue $C = 3t/T$, onde ambos $t$ e $T$ se referem a conjuntos de três nós no grafo, por exemplo, $i, j, k$. O valor de $t$ destina-se a medir o número de triângulos no grafo, isto é, são conjuntos onde uma aresta conecta $i$ e $j$, outra conecta $j$ e $k$, e uma terceira conecta $i$ e $k$. O valor de $T$, por outro lado, conta os conjuntos de três vértices conectados por duas arestas pelo menos (podendo a terceira existir ou não). O fator de três usualmente visto no numerador de $C$ reflete o fato de que existem três conjuntos de tamanho três do primeiro tipo para cada triângulo no grafo. Assim sendo, $0 \leq C \leq 1$ (variando de nenhuma transitividade até transitividade total).

Na análise dos grafos, cada coeficiente de clusterização será apresentado lado-a-lado com o valor que ele teria caso todo nó $i$ continuasse a ter o mesmo grau $\delta_i$, mas as ligações fossem criadas de forma aleatória[23]. Esse valor, chamado de $C'$ é dado por

$$C' = \frac{\left(\delta^{(2)} - \delta\right)^2}{n\delta^3} \tag{4.4}$$

onde $\delta^{(2)} = (1/n)\sum_i \delta_i^2$.

A última medida global consiste na verdade de um conjunto de quatro coeficientes de assortatividade. Cada um é um coeficiente de correlação de Pearson de duas sequências de números de tamanho $m$. Sendo $\alpha_1, \alpha_2, \ldots, \alpha_m$ e $\beta_1, \beta_2, \ldots, \beta_m$ as sequências; $\mu_\alpha$ e $\mu_\beta$ são as médias correspondentes; e $\sigma_\alpha$ e $\sigma_\beta$ os desvios padrão. Esse coeficiente é dado por

$$r_{\alpha,\beta} = \frac{(1/m)\sum_e \alpha_e\beta_e - \mu_\alpha\mu_\beta}{\sigma_\alpha\sigma_\beta}. \tag{4.5}$$

O coeficiente de assortatividade original é obtido fazendo-se $\alpha_e = \delta_i^-$ e $\beta_e = \delta_j^+$ para a aresta dirigida $e$ de $i$ para $j$ [24, 25]. Isto é, mede a relação dos graus de saída dos nós que estão nas origens das arestas com os graus de entrada dos nós que se encontram nos destinos das mesmas. Em resumo, consiste em usar $out, in$ no lugar de $\alpha, \beta$. No presente trabalho foram consideradas ainda outras três variações com base nas outras combinações possíveis $(in, out;\ out, out;\ in, in)$[26, 27]. Os resultados das medidas aqui apresentadas estão na Seção 4.2.1. A Figura 4.1 mostra as variações possíveis para $\alpha$ e $\beta$ no coeficiente.

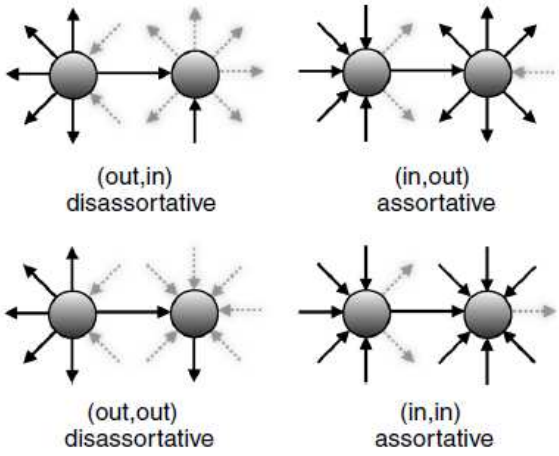



Figura 4.1: Variações de assortatividade [26, 27].

### 4.1.2 Medidas Locais

Apresentar as medidas locais de um grafo requer que cada medida de interesse seja avaliada por nó e apresentada na forma de distribuição de probabilidade em relação ao grafo todo. Nesta seção serão apresentadas as medidas de interesse utilizadas no trabalho, cujos respectivos resultados serão exibidos depois na Seção 4.2.2 em gráficos mostrando a distribuição acumulada complementar (referenciada como CCD), denotadas por $F(z)$ para um valor possível $z$, que é a probabilidade de que um nó escolhido aleatoriamente tenha um valor de medida que ultrapasse $z$. O cálculo de $F(z)$ foi feito considerando-se a fração de $n$ que representa os nós nos quais o valor da medida está acima de $z$.

As medidas locais mais amplamente estudadas são o grau (não dirigido), grau de entrada e grau de saída de um nó. Não apenas pelo fato de já terem sido calculadas em um grande variedade de domínios, mas também pelo motivo de que conhecer suas respectivas distribuições contribui no estudo de muitas outras propriedades de grafos [28]. Logo, essas medidas são as três primeiras estudadas, uma vez que a caracterização dos graus de entrada e saída tem com o passar dos anos conduzido a importantes descobertas em relação à Web (será chamada no texto por $W^*$) e à *Wikipedia* considerando-se todas as páginas em inglês (será chamada no texto por $W^+$). Diferentes estudos baseados em dados distintos mostram que a distribuição dos graus de entrada do grafo da Web segue uma distribuição de lei de potência [21, 29, 30]. Ou seja, a probabilidade de que um nó escolhido ao acaso tenha um grau de entrada $k > 0$ é proporcional a $k^{-\alpha}$ (então, a CCD correspondente é aproximadamente proporcional a $k^{1-\alpha}$) para $\alpha \approx 2,1$. Leis de potência semelhantes foram também reportadas para o grau de saída dos grafos, mas nesses casos, parece não haver tanta concordância [30]. Alguns trabalhos mostram ainda que para o grafo da *Wikipedia* ($W^+$), suas distribuições de grau, grau de entrada e de saída seguem leis de potência, com expoentes $-2,21$, entre $-2,65$ e $-2$, e $-2,37$ respectivamente [31, 32]. Leis de potência são inerentes a redes livre de escala [33], e o seu aparecimento em grafos como $W^*$ e $W^+$ têm sido frequentemente explicados pelo mecanismo de conexão preferencial (*preferential attachment*) [31, 34–36].

As outras medidas locais analisadas são apresentadas na Tabela 4.3. Quatro delas ($B_i$, $S_i$, $C_i$ e $G_i$) são medidas de centralidade do nó $i$ no grafo, sendo portanto relacionadas a menores caminhos dirigidos nos quais $i$ participa de alguma forma. As três medidas restantes são relacionadas a mecanismos de busca na Web. Duas delas mostram o quanto um nó $i$ se classifica como um *hub* ($y_i$) ou como um *authority* ($x_i$). A Figura 4.2 mostra o conceito, onde o valor de *hub* de um nó depende dos valores de *authority* dos nós que ele aponta, e de forma semelhante, o valor de *authority* de um nó depende dos valores de *hub* dos nós que apontam para ele. A terceira medida

calculada para as redes é o *PageRank* ($\rho(i)$), relacionada ao *Google*.

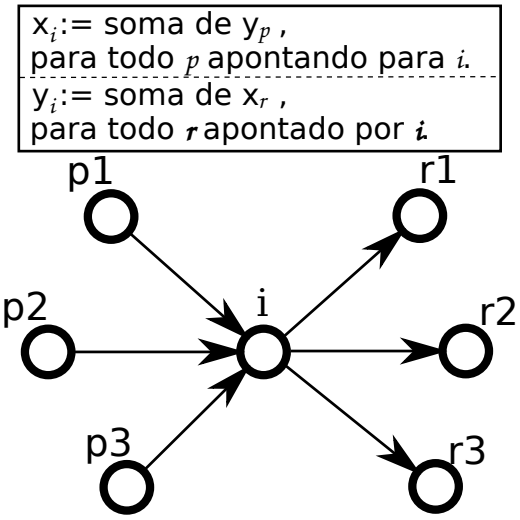



Figura 4.2: Regras para o cálculo de *authorities* e *hubs*.

As medidas de centralidade podem ser computadas através de variações de um algoritmo já bem conhecido [37]. No caso das medidas relacionadas ao *HITS*, primeiramente todo $x_i$ e $y_i$ são iniciados com 1. Então, $x_i$'s e $y_i$'s são alternadamente atualizados através das regras mostradas na Tabela 4.3. As atualizações de $x_i$ são precedidas pela normalização dos valores resultantes de modo que $\sum_i x_i^2 = 1$. Os valores de $y_i$ também são normalizados de forma semelhante. Após os resultados convergirem, todas as medidas são atualizadas de modo a $\sum_i x_i = \sum_i y_i = 1$. Para a medida de *PageRank*, novamente todo $\rho_i$ é iniciado com 1, e a regra de atualização mostrada na Tabela 4.3 é aplicada iterativamente até estarem estáveis. Nesse momento, todos os valores $\rho_i$'s são normalizados de modo que $\sum_i \rho_i = 1$. Para as medidas relacionadas ao *HITS* e *PageRank* o algoritmo desenvolvido foi executado enquanto os valores calculados nas duas últimas iterações para todos os nós apresentassem uma diferença contida no intervalo $[-10^{-16}, 10^{-16}]$.

Ainda com relação ao *PageRank*, foi feito também um estudo em relação à quantidade da medida que cada uma das bibliotecas preserva. Para melhor entendimento, considere os dois exemplos simples a seguir. No algoritmo, conforme já comentado, inicialmente atribui-se 1 como valor de *PageRank* de todas as páginas. Então, inicia-se a execução do mesmo, que convergirá com as iterações, resultando no valor de *PageRank* calculado para cada uma das páginas. Fica claro portanto que o valor total para um site (somando o valor de todas as páginas que o compõem), se analisado isoladamente (isto é, desconsiderando links externos que cheguem ou saiam dele), será no máximo igual a $n$. O primeiro exemplo onde esse resultado seria alcançado é exibido na Figura 4.3.

A Tabela 4.2 mostra a convergência do algoritmo nas dez primeiras iterações.

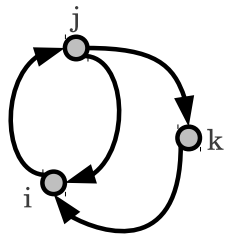



Figura 4.3: Exemplo 1 para cálculo do *PageRank*.

Tabela 4.2: Valores de *PageRank* obtidos a cada iteração para o Exemplo 1.

| Iteração | $i$ | $j$ | $k$ |
|---|---|---|---|
| 0 | 1,0000000000 | 1,0000000000 | 1,0000000000 |
| 1 | 1,4250000000 | 1,0000000000 | 0,5750000000 |
| 2 | 1,0637500000 | 1,3612500000 | 0,5750000000 |
| 3 | 1,2172812500 | 1,0541875000 | 0,7285312500 |
| 4 | 1,2172812500 | 1,1846890625 | 0,5980296875 |
| 5 | 1,1618180859 | 1,1846890625 | 0,6534928516 |
| 6 | 1,2089617754 | 1,1375453730 | 0,6534928516 |
| 7 | 1,1889257074 | 1,1776175091 | 0,6334567835 |
| 8 | 1,1889257074 | 1,1605868513 | 0,6504874414 |
| 9 | 1,1961637369 | 1,1605868513 | 0,6432494118 |
| 10 | 1,1900114118 | 1,1667391764 | 0,6432494118 |

O valor final obtido para cada uma das páginas do exemplo é de: $i = 1,1921989825$, $j = 1,1633691351$ e $k = 0,6444318824$, ficando claro que o valor máximo de *PageRank* para esse site foi mantido, com a soma dos valores resultando em 3. Considere agora o Exemplo 2, que consiste na retirada do link de $k$ para $i$. O grafo resultante é exibido na Figura 4.4.

No Exemplo 2, a página $k$ receberá um valor de *PageRank* de $j$, mas o seu próprio valor não será propagado adiante, cenário típico onde a soma dos valores finais será inferior a 3. Nesse caso, os valores de *PageRank* obtidos serão: $i = 0,3346379648$, $j = 0,4344422701$, $k = 0,3346379648$. A soma dos valores é então igual a $1,1037181997$.

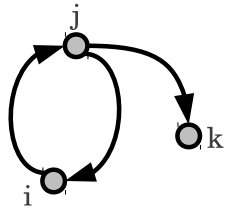



Figura 4.4: Exemplo 2 para cálculo do *PageRank*.

Pelos exemplos percebe-se que quanto mais nós *sink*, mais perda do valor inicial de *PageRank* ocorre. Na Seção 4.2.2 serão apresentados para cada biblioteca o valor

de *PageRank* total considerando-se todas as páginas, em relação ao total possível, que é igual a $n$.

Sobre as medidas de centralidade e algumas das ferramentas para cálculos em grafos que foram utilizadas, como a *Gephi*[1] e a *Guess*[2] observou-se que elas não apresentam como possibilidade de cálculo todas as medidas aqui apresentadas, se limitando geralmente à *Betweenness centrality* dos nós ($B_i$). O motivo talvez seja porque no artigo que apresenta o algoritmo é mostrada apenas a implementação para o cálculo de $B_i$, e o autor cita que as demais medidas podem ser obtidas com adaptações no algoritmo, preservando-se a mesma complexidade, mas não apresenta as adaptações. Assim, os algoritmos para as medidas foram desenvolvidos e extensivamente testados. Para os casos em que encontrou-se ferramentas capazes de calcular as mesmas medidas, as ferramentas serviram como base na validação. Além das já citadas *Gephi* e *Guess*, utilizou-se também a ferramenta *Network Workbench*[3]. Durante a implementação aconteceu também de encontrar erro no algoritmo do *HITS* na ferramenta *Gephi* quando se usava grafo dirigido. Ao pesquisar no fórum, descobriu-se que o problema já havia sido relatado, e que um usuário já havia informado como corrigi-lo, mas a aplicação da correção ainda não havia sido implementada na data da realização do trabalho.

[1] `https://gephi.org/`
[2] `http://graphexploration.cond.org/`
[3] `http://nwb.cns.iu.edu/`

Tabela 4.3: Medidas locais adicionais para o nó $i$. $\sigma_{jk}$ é o número de menores caminhos existentes de $j$ a $k$, enquanto que $\sigma_{jk}(i)$ conta somente aqueles que passam por $i$.

| Designação | Fórmula | Referência(s) |
|---|---|---|
| Betweenness centrality | $B_i = \sum_{j \neq i} \sum_{\substack{k \neq i \\ k \in R_j}} \frac{\sigma_{jk}(i)}{\sigma_{jk}}$ | [38, 39] |
| Stress centrality | $S_i = \sum_{j \neq i} \sum_{\substack{k \neq i \\ k \in R_j}} \sigma_{jk}(i)$ | [40] |
| Closeness centrality | $C_i = \begin{cases} \frac{1}{\sum_{j \in R_i} d_{ij}}, & \text{se } R_i \neq \emptyset \\ 0, & \text{caso contrário} \end{cases}$ | [41] |
| Graph centrality | $G_i = \begin{cases} \frac{1}{\max_{j \in R_i} d_{ij}}, & \text{se } R_i \neq \emptyset \\ 0, & \text{caso contrário} \end{cases}$ | [42] |
| Regra de atualização do *HITS* para *hubs* | $y_i := \sum_{j \in O_i} x_j$ | [43] |
| Regra de atualização do *HITS* para *authorities* | $x_i := \sum_{j \in I_i} y_j$ | [43] |
| Regra de atualização do *PageRank* com fator de amortecimento $0,85$ | $\rho_i := 0.15 + 0.85 \sum_{j \in I_i} \frac{\rho_j}{\delta_j^-}$ | [4] |

### Medidas Locais e Robustez da GSCC

Nos grafos estudados, bem como em todos os grafos que refletem redes do mundo real, a existência de componente fortemente conexa gigante (GSCC) é uma mera questão de observação: Analisa-se o grafo das componentes fortemente conexas e seleciona-se a maior. No entanto, em um sentido mais abstrato, modelos de grafos aleatórios de redes têm sido estudados sobre a existência de tal componente relacionado a um cenário de crescimento (isto é, a quantidade de nós e/ou arestas aumenta fazendo com que o grafo se torne mais denso). Tais estudos se iniciaram com os grafos aleatórios de Erdős-Rényi (ER) [44], que são não dirigidos e cujos graus se caracterizam por uma distribuição de Poisson. Uma vez que as arestas não possuem direção no modelo ER, observa-se portanto componentes fracamente (ao invés de fortemente) conexas, ou simplesmente componentes conexas, e a componente conexa gigante (GCC). Verifica-se que com o incremento de $\delta$ (grau médio) de 1 à medida que o grafo se torna mais denso, surge uma GCC como uma componente conexa que em um primeiro momento aparece separada das demais em virtude do seu tamanho [45]. Fenômenos semelhantes ocorrem também em muitos outros modelos de grafos aleatórios, incluindo variações dirigidas no que diz respeito ao surgimento de uma GSCC [28, 46–49].

Outro fenômeno consiste na quebra da GCC ou da GSCC quando nós são continuamente isolados do restante do grafo através da remoção de todas as arestas que incidem sobre eles. No caso dos grafos ER, por exemplo, é esperada que a quebra da GCC ocorra após a fração $1 - 1/\delta$ dos nós ter sida aleatoriamente isolada [50], considerando que inicialmente $\delta > 1$ (ou seja, que realmente existe uma GCC inicialmente). Resultados semelhantes também têm sido obtidos para grafos não dirigidos com a distribuição de graus seguindo uma lei de potência. Contudo, ao contrário dos grafos ER, cujos graus dos vértices se concentram em torno da média, agora podem existir nós com alto grau, fazendo sentido portanto analisar tanto nós específicos quanto aleatoriamente. Conforme pode-se perceber, para $\alpha = 2,5$ (que se considera descrever o grafo da Internet) só se espera que a GCC deixe de existir após pelo menos 99% dos nós terem sido isolados aleatoriamente. Contudo, para grafos relativamente pequenos, esse valor pode ser baixo como cerca de 80% [51]. Isolando os nós de maior grau no entanto, espera-se que cerca de 20% já seja o suficiente [52]. O estudo da robustez da GSCC foi feito no trabalho mais como efeito ilustrativo, não tendo relação com os propósitos da ferramenta de busca em si.

## 4.2 Resultados

Após a apresentação da metodologia então na Seção 4.1, serão apresentados nessa seção os resultados dos estudos. De forma semelhante, os resultados estão divididos em duas subseções: Medidas globais; e Medidas locais.

### 4.2.1 Medidas Globais

As medidas globais dos grafos da Tabela 4.1 são apresentadas nas Tabelas 4.4 e 4.5, que incluem uma linha adicional para o já citado grafo da *Wikipedia* em inglês considerando-se todas as páginas ($W^+$), obtido em um passado relativamente recente [31, 32]. A Tabela 4.4 contém ainda mais uma linha com medidas do grafo da Web ($W^*$), agora baseadas em dados de um trabalho mais antigo [29][4]. Nem todas as medidas calculadas foram encontradas para os grafos $W^+$ e $W^*$, conforme indicado nos espaços deixados em branco na Tabela 4.4. Os grafos estão organizados nas Tabelas 4.4 e 4.5 em ordem não crescente de $n$, e portanto em ordem decrescente de $m$.

Os resultados mostrados na Tabela 4.4 indicam que a densidade relativa entre arestas e nós, conforme determinado por $\delta^+$, possui a mesma ordem de grandeza para a maioria dos grafos, sendo a exceção o $W'$, baseado no grafo da *Wikipedia* considerando somente links *See-also*, cujo valor de $\delta^+$ apresenta uma ordem a menos de grandeza. Os contribuidores para as páginas matemáticas da *Wikipedia* portanto parecem acrescentar consideravelmente menos links *See-also* do que aqueles que contribuem para a *MathWorld*. É importante notar ainda que os cinco grafos matemáticos apresentados possuem valores bastante diferentes para a taxa $2\delta^+/\delta - 1$, indicando o $W'$ como o grafo com a menor quantidade de arestas antiparalelas contribuindo para os graus em média; e o $M'$, o grafo baseado nos links *See-also* da *MathWorld*, como tendo a maior quantidade. Novamente, os contribuidores da *MathWorld* parecem ser mais cuidadosos ao acrescentar referências cruzadas na forma *See-also*.

[4]Referências um pouco mais recentes parecem indicar valores de $S$ próximos a 0,33 para um grafo Web de tamanho semelhante [30], mas sem estimativas para $l$.

Tabela 4.4: Medidas globais: grau médio de entrada ou saída ($\delta^+$), grau médio ($\delta$) e valor resultante de $2\delta^+/\delta - 1$, fração de $n$ compondo a GSCC $S$, distância média entre nós distintos ($l$) e coeficiente de clusterização ($C$, juntamente com o valor $C'$ que ele teria caso as ligações fossem aleatórias).

| Graph | $n$ | $m$ | $\delta^+$ | $\delta$ | $2\delta^+/\delta - 1$ | $S$ | $\ell$ | $C$ | $C'$ |
|---|---|---|---|---|---|---|---|---|---|
| $W^*$ | 203 549 046 | 1 466 000 000 | 7,20 | | | 0,28 | 16,18 | | |
| $W^+$ | 339 834 | 5 278 037 | 15,53 | | | 0,82 | 4,90 | | |
| $W$ | 37 723 | 688 589 | 18,25 | 30,62 | 0,19 | 0,80 | 4,11 | 0,055 | $7,59 \times 10^{-4}$ |
| $W'$ | 37 723 | 21 503 | 0,57 | 1,04 | 0,09 | 0,02 | 15,27 | 0,061 | $4,96 \times 10^{-8}$ |
| $M$ | 15 095 | 92 648 | 6,14 | 9,72 | 0,26 | 0,78 | 5,32 | 0,048 | $5,18 \times 10^{-4}$ |
| $M'$ | 15 095 | 46 965 | 3,11 | 4,45 | 0,40 | 0,62 | 7,45 | 0,093 | $1,77 \times 10^{-4}$ |
| $D$ | 908 | 7 527 | 8,29 | 12,81 | 0,29 | 0,81 | 3,79 | 0,062 | 0,011 |

Tabela 4.5: Medidas globais: coeficiente de assortatividade.

| Graph | $r_{\text{out,in}}$ | $r_{\text{in,out}}$ | $r_{\text{out,out}}$ | $r_{\text{in,in}}$ |
|---|---|---|---|---|
| $W^+$ | $-0,150$ | | | |
| $W$ | $-0,071$ | $0,075$ | $-0,074$ | $-0,022$ |
| $W'$ | $0,041$ | $0,094$ | $0,070$ | $0,028$ |
| $M$ | $-0,037$ | $-0,018$ | $-0,015$ | $-0,019$ |
| $M'$ | $-0,054$ | $-0,031$ | $-0,058$ | $-0,036$ |
| $D$ | $-0,169$ | $0,006$ | $-0,053$ | $-0,043$ |

Um dos maiores contrastes que pode ser visto na Tabela 4.4 é com relação ao valor de $S$, o tamanho da GSCC em relação a $n$. Enquanto que para o grafo da Web $W^*$ a melhor estimativa fica em torno de 28% dos nós compondo a GSCC, para o grafo da *Wikipedia* $W^+$, bem como para a maioria dos grafos das bibliotecas matemáticas consideradas, a GSCC engloba um percentual substancialmente maior de nós (entre 62 e 82%). A exceção novamente ocorre no grafo $W'$, cuja GSCC possui apenas 2% dos nós, e na qual poderá se perceber ser conectada de forma bastante esparsa pelos links *See-also*. Esse baixo percentual ocorre justamente em virtude da pequena relação de arestas por vértice no grafo, resultando em muitos vértices isolados, o que gera por consequência várias componentes de tamanho igual a 1.

O restante dos dados na Tabela 4.4 se referem a $l$ e a $C$, uma média do tamanho dos caminhos no grafo (de forma dirigida) e ao coeficiente de clusterização (de forma não dirigida), respectivamente. Primeiro observou-se que em seis dos sete grafos, $l$ é proporcional a $\ln n$ com uma constante da ordem de $10^{-1}$, sendo a exceção em $W'$, para o qual a constante de proporcionalidade é aproximadamente $1,45$ (isso acontece em função das distâncias consideravelmente maiores em comparação com $W$, conforme esperado em virtude do valor $\delta^+$ substancialmente menor). Em todos os casos no entanto, as distâncias são em média bastante pequenas dado o valor de $n$, e os coeficientes de clusterização ($C$) visivelmente maiores do que o apresentado em ($C'$) por um fator de pelo menos duas ordens de magnitude, podendo dessa forma se classificar todos os grafos como estruturas de mundo-pequeno. A única exceção na ordem de grandeza da diferença entre $C$ e $C'$ ocorre em $D$, onde $C \approx 5,46C'$. Uma possível explicação para o cenário visto em $D$ pode ser pelo fato de que cada página da *DLMF* contém substancialmente mais material, o que acaba refletindo na baixa quantidade de nós no grafo. A considerável diferença vista nos grafos de $C$ para $C'$, embora usual, não acontece sempre [23].

A Tabela 4.5 contém todos os quatro coeficientes de assortatividade para os cinco grafos relacionados, e para $W^+$, o grafo na integra da *Wikipedia* em inglês. A grande maioria dos valores é da ordem de $10^{-2}$ no máximo, sendo portanto suficientemente próximos de zero para serem considerados como não relacionados. Em geral isso ou

é um indicativo de um padrão de conexões aleatórias (que não é o caso), ou que o critério para a inserção de arestas não apresentam relação a graus de entrada ou saída (o que parece ser mais plausível). Curiosamente, embora, o mesmo se aplique para as suas únicas exceções, $W^+$ e $D$, no qual um valor moderadamente negativo sugere que nesses dois grafos conexões são feitas de tal forma que produz um pequeno, mas perceptível grau de desassortatividade. Ou seja, existe uma pequena tendência de nós com alto (pequeno) grau de saída a se conectarem a outros com pequeno (alto) grau de entrada. Essa tendência é exibida de forma bastante semelhante para $r_{out,in}$ em ambos $W^+$ e $D$ ($-0,150$ no primeiro caso, $-0,169$ no segundo). Talvez o fato supracitado de que páginas da *DLMF* possuam mais material do que as páginas das outras bibliotecas de alguma maneira torne $D$ semelhante a $W^+$ nesse caso.

### 4.2.2 Medidas Locais

Os gráficos para as medidas locais são apresentados na Figura 4.6 (grau não dirigido, grau de entrada, e saída), Figura 4.7 (centralidades), e Figura 4.10 (*hub*, *authority* e *PageRank*). Uma característica interessante em todos eles é que nenhuma das medidas parece expressar claramente uma lei de potência para nenhum número significante de ordens de grandeza. Por exemplo, embora tenha-se verificado que a CCD do grau de entrada da *DLMF* se aproxima de uma lei de potência com $\alpha = 2,47$, esse fato parece considerável apenas para uma ordem de grandeza (aproximadamente entre 10 e 100). No caso da Figura 4.6, em particular, a ausência de uma lei de potência parece confirmar o esperado, de que em domínios específicos, como são as cinco bibliotecas, o conhecimento é o que guia o estabelecimento dos links, ao invés de algum critério baseado na popularidade, como a conexão preferencial (*preferential attachment*).

Ainda analisando o gráfico do grau de entrada ((*a*) da Figura 4.6), é possível ver que na *DLMF* algumas páginas possuem valores bastante elevados em comparação com as demais. Fato semelhante acontece no gráfico dos Graus de Saída (*b*), onde a biblioteca *Wikipedia* Original apresenta algumas páginas com valores altos, discrepantes em comparação com as demais.

Nos graus de entrada, as páginas da *DLMF* que apresentam valores de entrada bastante elevados são exibidos na Tabela 4.6. Na *Wikipedia* Original, dentre essas páginas com valor elevado de grau de saída, encontram-se em menor quantidade páginas de conteúdo (ex: Probability_distribution[5]), e de biografia de grandes nomes na matemática (ex: Aristotle). A maioria das páginas são do tipo "List_of..." (ex: List_of_statistics_articles), bastante comuns na *Wikipedia*, e que formam uma espécie de índice de páginas de um determinado assunto. O destaque fica para as 6 páginas

[5]Acrescente aos nomes `http://en.wikipedia.org/wiki/`

de maior valor de grau de saída, que não são páginas de conteúdo, mas páginas de *namespace Wikipedia*, que possuem informação ou discussão sobre a biblioteca (ex: *Wikipedia: Pages_needing_attention/Mathematical_and_Natural_Sciencies*).

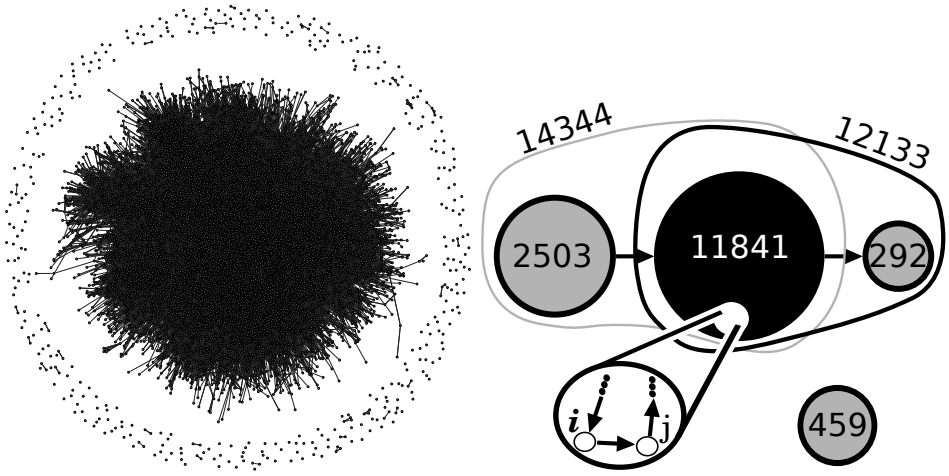
Tabela 4.6: Páginas de alto grau de entrada na *DLMF*.

| Página | Título da página | Título do capítulo da página |
|---|---|---|
| 2.1 | Definitions and Elementary Properties | Asymptotic Approximations |
| 1.9 | Calculus of a Complex Variable | Algebraic and Analytic Methods |
| 4.14 | Definitions and Periodicity | Elementary Functions |
| 5.2 | Definitions | Gamma Function |
| 1.4 | Calculus of One Variable | Algebraic and Analytic Methods |
| 4.2 | Definitions | Elementary Functions |

Nas medidas *Betweenness* e *Stress centrality* acontece ainda um cenário interessante em torno da GSCC. Na Figura 4.5 é mostrado o grafo da *MathWorld* ($M$) original em ($a$), apenas como efeito ilustrativo. Em ($b$) é uma representação do mesmo, sendo o vértice preto de tamanho igual a 11 841 a GSCC. Os vértices cinza antes e depois indicam a quantidade de vértices que alcançam e são alcançados pela componente respectivamente. O último vértice cinza abaixo, com valor igual a 459 são os nós que não possuem links para a componente, e nem são alcançados por ela. Assim, ao analisar os vértices da componente, mais os vértices que chegam até ela, essa soma resulta em 14 344 (acima da marcação à esquerda). Ao se fazer análise semelhante, mas olhando agora os vértices alcançáveis à partir da componente gigante, o valor obtido é o 12 133 (exibido na marcação à direita). Considere agora o zoom dado na GSCC, onde foram destacados dois nós ($i$ e $j$). O nó $i$ é o único que possui link para $j$. Ou seja, $I_j = \{i\}$. Mesmo que $i$ possua também uma única aresta de saída, $B_i$ já terá valor igual a = 14 342, uma vez que ele está no caminho entre todos os vértices que compõem ou alcançam a GSCC. De forma semelhante, caso $O_i = \{j\}$, o nó $j$ terá $B_j = 12\,131$. Esses cenários acabam colocando páginas irrelevantes com valores relativamente altos tanto para $B_i$ quanto para $S_i$. Uma página que tenha um valor de $B_i = 14\,342$ por exemplo, do total de $n = 15\,095$ para o grafo $M$, terá 9 854 páginas com valor de $B_i$ inferior ao dela, colocando portanto essa página à frente de quase 2/3 do grafo. É importante notar que geralmente são páginas de pouco ou nenhum conteúdo. Um exemplo pode ser visto em $M$, onde a página *Anamorphogram.html*[6] é a única a apontar para *AnamorphicArt.html*, sendo que nenhuma delas possui conteúdo. A Tabela 4.7 mostra os valores de vértices que alcançam, e são alcançados pela GSCC para as cinco bibliotecas.

No artigo [19] é feito um estudo sobre a estrutura da *MathWorld*. Os resultados obtidos são semelhantes ao caracterizar essa rede, concluindo ter propriedade de

[6]Acrescente aos nomes `http://mathworld.wolfram.com/`




(a) Grafo original. (b) GSCC e demais vértices.

Figura 4.5: *MathWorld*.

Tabela 4.7: Vértices que alcançam ou são alcançáveis à partir da componente.

| Biblioteca | Vértices que alcançam | Vértices alcançáveis |
|---|---|---|
| *DLMF* | 817 | 792 |
| MathOriginal | 14 344 | 12 133 |
| MathSeeAlso | 10 524 | 10 728 |
| WikiOriginal | 34 846 | 31 380 |
| WikiSeeAlso | 2 321 | 3 997 |

mundo-pequeno, e que as conexões não são criadas segundo o critério de conexão preferencial (*preferential attachment*). Vale frisar novamente, conforme já citado no Capítulo 3, que o grafo usado no artigo possui $n = 12\,000$. A Tabela 4.8 mostra as dez páginas de maior valor recuperadas pelo presente trabalho, e pelo outro.

Tabela 4.8: Top 10 de páginas considerando-se $B_i$ dos trabalhos.

| Ordem | Título | Título [19] |
|---|---|---|
| 1 | Circle | Circle |
| 2 | Polynomial | Polynomial |
| 3 | Binomial Coefficient | Binomial Coefficient |
| 4 | Prime Number | Prime Number |
| 5 | Integer | Integer |
| 6 | Set | Set |
| 7 | Matrix | Matrix |
| 8 | Group | Group |
| 9 | Triangle | Power |
| 10 | Power | Graph |

Percebe-se como única diferença a página na posição 9 *Triangle*, que aparentemente não foi obtida pelo outro trabalho, talvez por um problema com o *WebCrawler* (já citado no Capítulo 3).

Na Figura 4.7, os valores da CCD para $C_i$ e $G_i$ compartilham uma propriedade

peculiar, onde os nós aparecem concentrados dentro de três intervalos relativamente pequenos. Para ajudar no entendimento, os três intervalos foram destacados no gráfico de *Closeness Centrality* (marcados como $i$, $ii$ e $iii$). Em cada uma das cinco bibliotecas, no ponto $i$ estão os nós que não possuem links de saída para nenhum outro (*sink*). Tendo portanto $R_i = O_i = 0$, e os valores de $C_i = G_i = 0$. No ponto $ii$ aparece o intervalo que na maioria das vezes é o mais denso em quantidade de nós. Nós que aparecem nesse intervalo possuem valor relativamente baixo de centralidade tipicamente associados com distâncias relativamente grandes em $R_i$. Esses são os nós que ou estão na GSCC ou possuem um caminho que alcança ela. Isso explica novamente a única exceção, que é condizente com o pequeno tamanho da GSCC da *Wikipedia* com links *See-also* ($W'$). O terceiro intervalo, marcado como $iii$ contém o restante dos nós, onde na maioria dos casos se caracterizam por pequenas distâncias dos nós em $R_i$. Esses nós se localizam portanto fora da maior componente do grafo, e também não possuem caminho que conduza a ela. Observe que no caso do grafo ($W'$) é justamente onde se concentram a maior parte dos vértices, condizente com o pequeno tamanho da sua GSCC, e pela predominância de vértices pouco conectados.

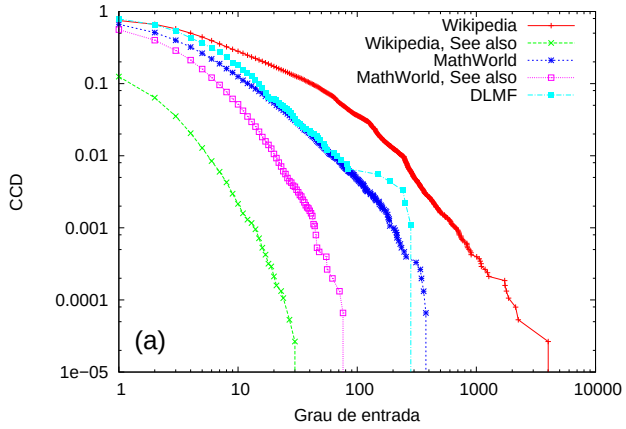



Figura 4.6: Gráficos CCD para os valores $\delta_i^+$ (a), $\delta_i^-$ (b), e $\delta_i$ (c) da $W$ (*Wikipedia*), $W'$ (*Wikipedia*, *See-also*), $M$ (*MathWorld*), $M'$ (*MathWorld*, *See-also*), e $D$ (*DLMF*).

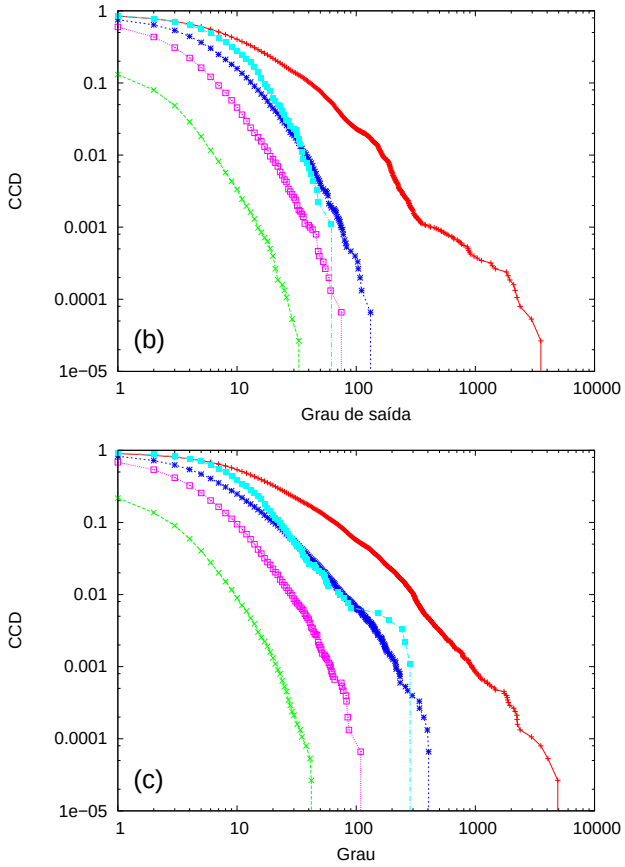



Figura 4.6: Continuação.

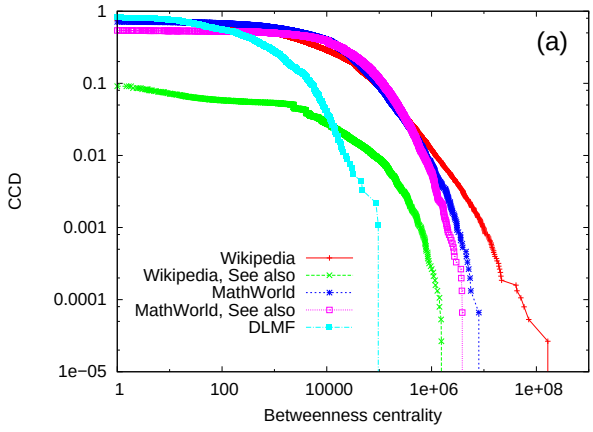



Figura 4.7: Gráficos CCD para os valores $B_i$ (a), $S_i$ (b), $C_i$ (c), e $G_i$ (d) da $W$ (*Wikipedia*), $W'$ (*Wikipedia*, *See-also*), $M$ (*MathWorld*), $M'$ (*MathWorld*, *See-also*), e $D$ (*DLMF*)

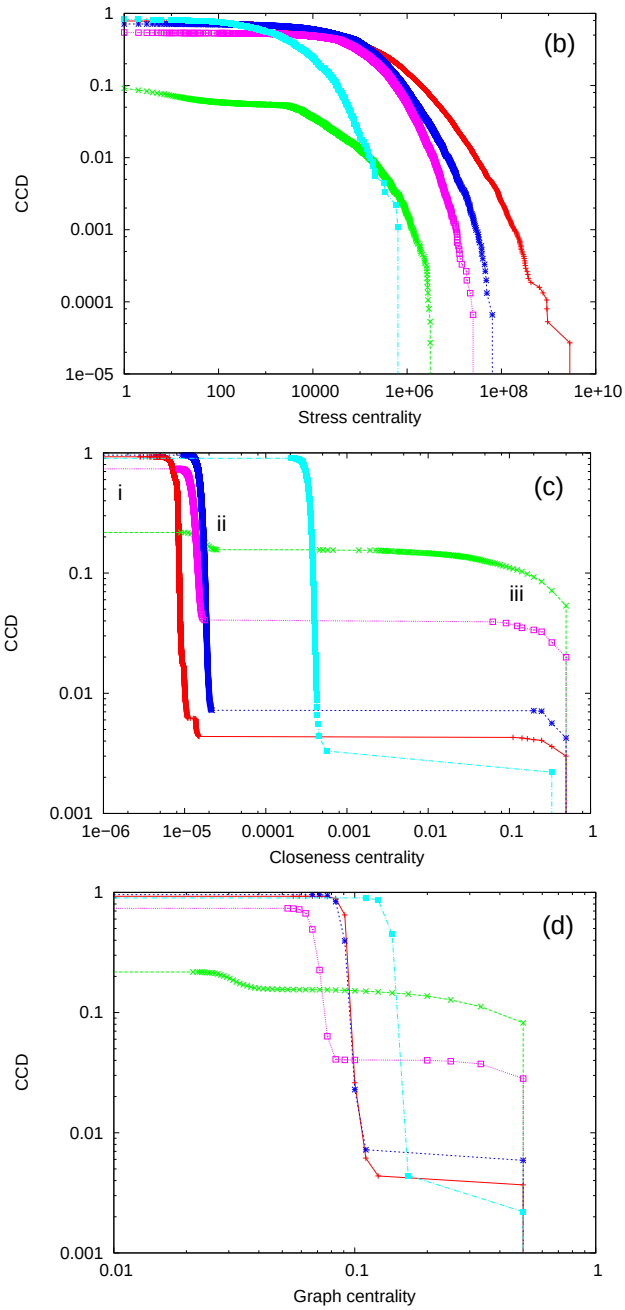



Figura 4.7: Continuação.

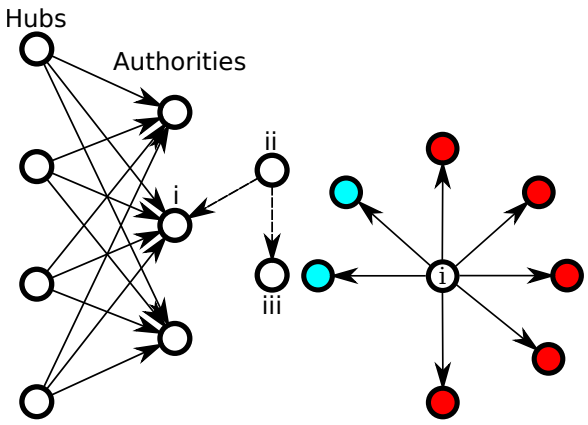
A Figura 4.10 apresenta os gráficos para as medidas relacionadas à busca na Web. Dois cenários interessantes podem ser destacados nos gráficos relacionados à *hub* e *authority* ($(a)$ e $(b)$ respectivamente). Na biblioteca *M′* é possível observar um formato semelhante a degraus. Analisando-se os nós que aparecem com valores nas extremidades desses degraus, constatou-se que eles participam de conexões semelhantes à mostrada na Figura 4.8 $(a)$. Os nós marcados à esquerda demonstram um cenário típico de bons *hubs*, enquanto que os nós no meio aparecem como bons *authorities*. Considere agora o nó $i$, que recebe um link do nó $ii$. O $ii$ por ter apontado um nó importante, receberá um valor relativamente bom como *hub*. Por consequência, o nó $iii$ que recebe apenas o link de $ii$ receberá a propagação se tornando um nó relativamente bom em *authority*. Esse cenário e pequenas variações dele compõem as extremidades dos degraus observados. Um exemplo real é mostrado na Figura 4.8 $(b)$, que representa as seguintes páginas:

- nó $i$: *Neighborhood*[7];
- nós azuis: são as páginas *vonNeumannNeighborhood* e *MooreNeighborhood*, que possuem alto valor de *authority*;
- nós vermelhos: páginas que possuem um valor semelhante de *authority* por receberem esse link de $i$. Das cinco páginas em vermelho, uma delas recebe apenas esse link, enquanto as outras possuem outras arestas de entrada, no entanto, as demais arestas contribuem com valores baixos de *authority* por não se originarem em páginas com valores altos de *hubs*. As páginas em vermelho são:
  - *NeighborhoodComplex*: possui apenas uma aresta de entrada, vinda justamente da página $i$;
  - *GraphNeighborhood*: possui duas arestas de entrada;
  - *OpenNeighborhood*: possui 4 arestas de entrada;
  - *OpenSet*: possui 13 arestas de entrada;
  - *Ball*: possui 21 arestas de entrada;

Os nós em vermelho estão entre a maioria das páginas que compõem o primeiro "degrau" no gráfico de *authority*, com valores entre $6,51\text{x}10^{-6}$ até $7,27\text{x}10^{-6}$.

No grafo da *Wikipedia* $(W)$ pode-se constatar a presença de um conjunto de páginas com valores elevados tanto de *hub* quanto de *authority*. Esse comportamento ocorre nas páginas de Estatística. Isso porque todas as páginas nesse grupo possuem um quadro "*Statistics*" ao seu final, composto por links para todas as demais. Assim,

[7]considere a URL como `http://mathworld.wolfram.com/Neighborhood.html`




(a) Exemplo. (b) Cenário real.

Figura 4.8: *MathWorld*.

essas páginas se tornam *hubs* e *authorities* de destaque em relação ao restante. Um exemplo pode ser visto em *Variance*. O quadro é mostrado na Figura 4.9. Os links estão ocultos dentro de cada aba, e podem ser exibidos clicando em "*show*".

| v · t · e | Statistics | [hide] |
|---|---|---|
| | Descriptive statistics | [show] |
| | Data collection | [show] |
| | Statistical inference | [show] |
| | Correlation and regression analysis | [show] |
| | Categorical, multivariate, time-series, or survival analysis | [show] |
| | Applications | [show] |
| | Category · Portal · Outline · Index | |

Figura 4.9: Quadro *Statistics*.

No *PageRank* é interessante observar que para o grafo $W\prime$ o gráfico aparece descontinuado em dois pontos (valores iguais a $3,524\text{x}10^{-5}$ e $1,27\text{x}10^{-4}$). Observa-se uma alta concentração de páginas nesses valores, que ocorre em função de dois cenários: *i*) havendo dois vértices $a$ e $b$, onde existe uma aresta no sentido $a \rightarrow b$, com $I_a = 0$, $O_a = 1$ e $I_b = 1$, o valor de *PageRank* normalizado ao final será igual a $3,524\text{x}10^{-5}$ (ocorrem 728 páginas nessa situação); *ii*) formação de subgrafos completos de forma isolada dos demais, terminando todas as páginas com valor de *PageRank* igual a 1 antes da normalização, e $1,27\text{x}10^{-4}$ após normalizado (com 163 páginas nessa situação).

A Tabela 4.9 mostra então a ideia já explicada na Seção de Metodologia. Foram calculados o *PageRank* para as cinco bibliotecas, e verificou-se a porcentagem do *PageRank* total que cada uma delas poderia ter. Conforme já era de se esperar, os grafos $M$ e $W$ por terem maior quantidade de arestas, chegam mais perto do

potencial total do grafo em *PageRank*. No extremo oposto se encontra o grafo $W'$ com apenas $20,87\%$ do valor de *PageRank* preservado.

Tabela 4.9: Total de *PageRank* por biblioteca, e porcentagem do máximo possível.

| Biblioteca | Total do *PageRank* | % do máximo |
|---|---|---|
| *DLMF* | $735,192767$ | $80,9684\%$ |
| *MathWorld* Original | $14\,349,229188$ | $95,0595\%$ |
| *MathWorld See-also* | $10\,469,525287$ | $69,3576\%$ |
| *Wikipedia* Original | $34\,909,567277$ | $92,5419\%$ |
| *Wikipedia See-also* | $7\,873,762168$ | $20,8726\%$ |

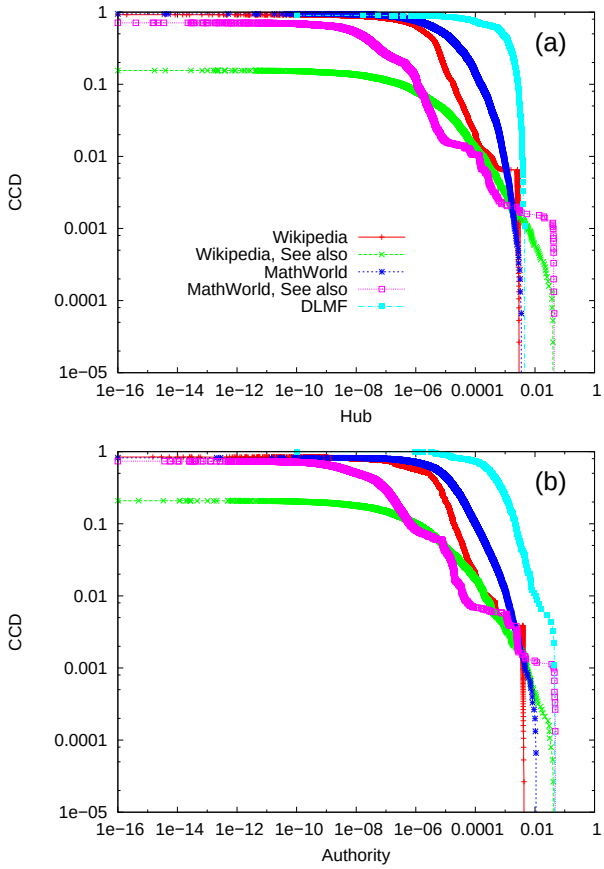



Figura 4.10: Gráficos CCD para os valores $y_i$ (a), $x_i$ (b), e $\rho_i$ (c) da $W$ (*Wikipedia*), $W'$ (*Wikipedia*, *See-also*), $M$ (*MathWorld*), $M'$ (*MathWorld*, See also), e $D$ (*DLMF*).

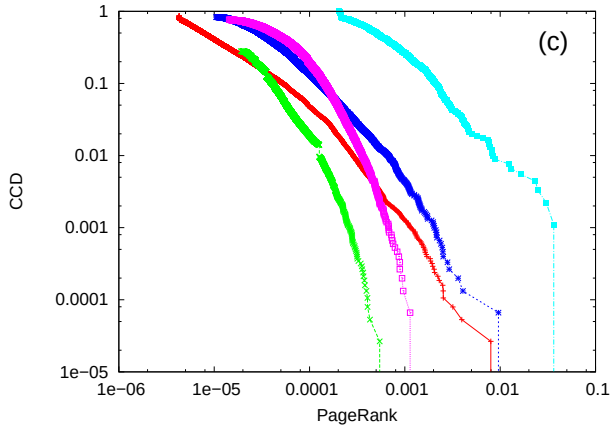



Figura 4.10: Continuação.

### Medidas Locais e Robustez da GSCC

Os resultados apresentados nessa seção descrevem a evolução de $S$, a fração de $n$ dentro da GSCC, à medida que nós são removidos, seja de maneira aleatória ou escolhendo-se o de maior valor segundo alguma das medidas locais calculadas. Na remoção aleatória, é apresentado o valor médio obtido de dez execuções independentes. Na remoção baseada nas medidas locais, considerou-se todas as dez discutidas na Seção 4.1.2. Em todos os casos o isolamento de nós é feito até que não haja mais componentes fortemente conexas contendo mais do que um nó. Quando o isolamento termina portanto, todos os nós restantes então ou estão isolados (sem vizinhos de entrada ou saída) ou são parte de uma porção acíclica do grafo atual.

Os resultados são exibidos na Figura 4.11, onde a fração de nós retirados até a condição de parada do algoritmo aparece contida no intervalo $[0,4; 0,6]$ para grafos não *See-also*, próxima de $0,3$ para $M'$ e menos de $0,01$ para $W'$. Se novamente se desconsiderar o grafo $W'$, cuja GSCC se mostrou bastante frágil, e talvez também o $M'$, observa-se nos três grafos restantes ($W$, $M$ e $D$) o que parecem ser GSCC's bastante resistentes. Quando se observa a retirada de nós seguindo alguma das medidas locais, os dados da Figura 4.11 mostram que nenhuma das medidas específicas em questão se sobressai em relação às outras, com exceção para a *Graph centrality* e *Closeness centrality* que se mostram bem menos efetivas em todos os casos, exceto para $W'$. Essas duas medidas são respectivamente a segunda e terceira menos efetivas na quebra da GSCC (seguidas pela retirada aleatória, que é a pior). Assim, exceto para essas, os dados revelam que com a retirada de aproximadamente $0,2$ para $W$ e $D$, e um pouco mais de $0,1$ para $M$, e então um pouco menos de $0,1$ para $M'$ e, finalmente, menos de $0,001$ para $W'$, já ocorre a quebra das componentes fortemente conexas de tamanho $\geq$ 2. A retirada de nós com base em alguma das medidas locais então, colocou os grafos não *See-also* novamente em um patamar de

medida separado dos demais, com $M'$ ficando no meio termo, e novamente o $W'$ com os resultados menores.

No grafo $W$ (Figura 4.11, $a$) é possível perceber que as medidas relacionadas ao *HITS* (*authority* e *hub*) e o *PageRank* começam com uma diferença de desempenho em relação às demais (até por volta de 5%) dos vértices retirados, onde as demais medidas se mostram mais eficientes na quebra da GSCC. Essa diferença ocorre porque essas três medidas demoram a retirar do grafo a página *List_of_mathematicians*. Quando essa página é retirada então, ocorre a desconexão de aproximadamente 2 500 páginas de dentro da componente, fazendo com que a eficiência dessas medidas se torne então semelhante às demais a partir desse ponto.

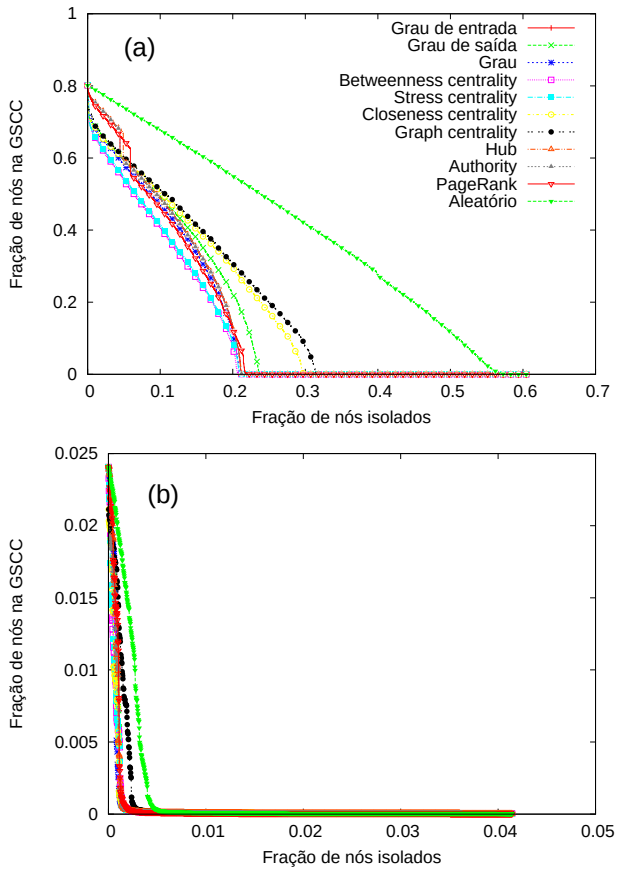



Figura 4.11: Evolução de $S$ em relação ao isolamento de nós em $W$ (*Wikipedia*; a), $W'$ (*Wikipedia*, *See-also*; b), $M$ (*MathWorld*; c), $M'$ (*MathWorld*, *See-also*; d), e $D$ (*DLMF*; e).

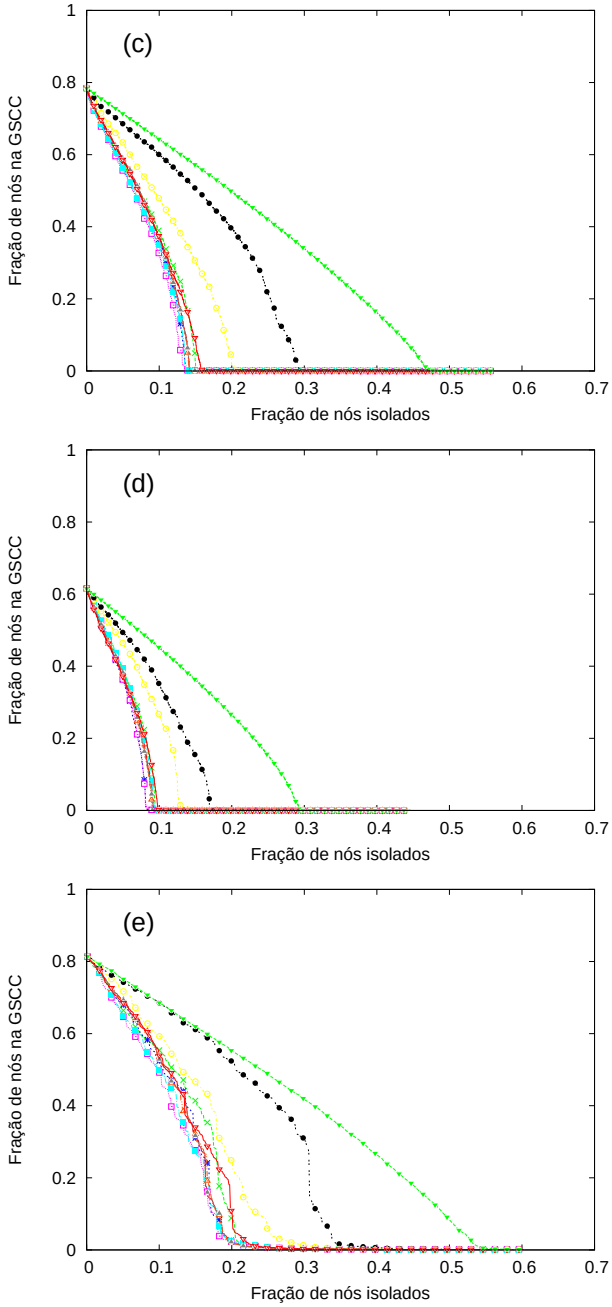



Figura 4.11: Continuação.

## 4.3 Conclusões do Capítulo

Foram estudadas nesse capítulo então três bibliotecas online matemáticas: porção matemática da *Wikipedia*, *MathWorld* e *DLMF*, sob a perspectiva de teoria de grafos. Para essa finalidade, foram considerados cinco grafos dirigidos nos quais os nós são as páginas, e as arestas representam as ligações diretas entre essas páginas através de links que apontam de uma para a outra. No caso da *Wikipedia* e do *MathWorld*, esses links acontecem em duas categorias identificáveis (aqueles que estão no corpo do texto e aqueles que aparecem nas seções *See-also*). Dessa forma, foram considerados dois grafos separados para cada uma dessas bibliotecas. Assim, propriedades globais e locais dos grafos foram medidas, com os objetivos de caracterizá-los e estudar a resistência à perda acidental ou intencional de material.

O Capítulo 5, conforme já mencionado, descreve a construção da ferramenta de busca por fórmulas matemáticas. As medidas globais e locais exibidas até aqui serão então testadas com o intuito de verificar como é o desempenho das mesmas como critério na ranqueamento de resultados com base em alguns estudos preliminares da ferramenta de busca desenvolvida.

# Capítulo 5

# Ferramenta de Busca

Este capítulo é separado em duas seções principais: Busca Textual, e Busca Matemática. Na primeira é desenvolvida uma busca textual por termos matemáticos, onde o objetivo é medir qual (ou quais) das dez medidas calculadas no capítulo anterior se sai melhor na ordenação de resultados, dado o domínio da matemática. Na segunda seção é detalhado o desenvolvimento da ferramenta de busca por fórmulas matemáticas, com a metodologia aplicada e os resultados obtidos em um teste feito com usuários.

## 5.1 Busca Textual

A busca textual foi desenvolvida, e fornece ao trabalho uma boa intuição relacionada ao ranqueamento dos resultados. A principal contribuição aqui é comparar se as medidas clássicas de ranqueamento (*HITS* e *PageRank*) são também adequadas, dado o domínio bastante específico.

### 5.1.1 Metodologia

A busca textual foi desempenhada para cada um dos cinco grafos dirigidos $W$, $W'$, $M$, $M'$ e $D$, buscando no texto dos nós por um uma determinada quantidade de palavras-chave em matemática, obtidas no site da *Microsoft Academic Search*[1] (*MAS*) em 1 de Novembro de 2012.

Foi utilizado o método padrão delineado em [3]. Considerando cada grafo e medida local nas consultas (feitas à partir das palavras-chave). O primeiro passo consiste na construção de uma lista $A$ de respostas (ordenadas em ordem não crescente da medida local em análise), bem como por um conjunto $R$ de nós relevantes.

[1] `http://academic.research.microsoft.com/RankList?entitytype=8&topDomainID=15&subDomainID=0.`

Então, inicia-se o cálculo das métricas do já conhecido método Precisão e Revocação (*Precision and Recall*). O conceito é exibido na Figura 5.1.

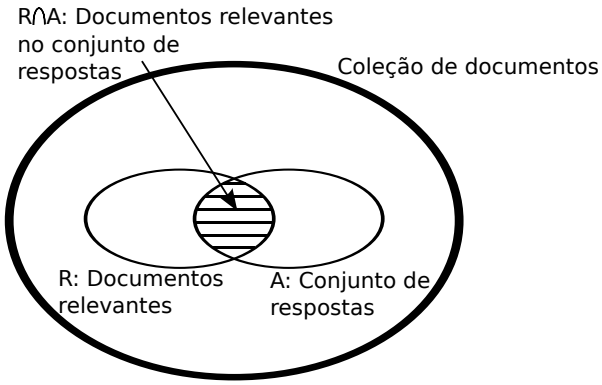



Figura 5.1: Conceito de *Precision and Recall* [3].

A curva de *Precision X Recall* é usualmente representada em onze níveis de *Recall* $(0\%, 10\%, 20\%, \ldots, 100\%)$. Para compreender a obtenção dos onze níveis, considere o seguinte exemplo: $R = \{d3, d56, d129\}$ e o conjunto de 15 documentos recuperados para a consulta exibidos na Tabela 5.1. Os documentos que pertencem ao conjunto $R$ foram marcados em $A$ com um * na respectiva ordem em que foram recuperados.

Tabela 5.1: Conjunto $A$.

| | | |
|---|---|---|
| $1 - d425$ | $6 - d615$ | $11 - d193$ |
| $2 - d87$ | $7 - d512$ | $12 - d715$ |
| $3 - d56*$ | $8 - d129*$ | $13 - d810$ |
| $4 - d32$ | $9 - d4$ | $14 - d5$ |
| $5 - d124$ | $10 - d130$ | $15 - d3*$ |

A Tabela 5.2 mostra o resultado de *Recall* e *Precision* para o exemplo. Para facilitar o entendimento, a análise da tabela deve ser feita olhando as 15 respostas na ordem em que ocorrem. Considere inicialmente os três primeiros resultados ($d425$, $d87$ e $d56$), onde apenas o $d56$ é um documento relevante. O valor de *Recall* nesse momento é igual a $33,3\%$, pois das três respostas olhadas, apenas uma foi relevante. O valor de *Precision* é também $33,3\%$, mas o motivo é porque de três documentos do conjunto dos relevantes (lembrando que $R = \{d3, d56, d129\}$), apenas um foi recuperado. Agora, expanda a análise até o resultado de número 8. De forma semelhante ao raciocínio feito para os três primeiros resultados, o valor de *Recall* será de $25,0\%$, pois dos 8 resultados, 2 são relevantes. O valor de *Precision* por sua vez aumenta para $66,6\%$, pois dos três documentos relevantes para essa busca, dois foram recuperados.

Tabela 5.2: Valores de Recall X Precision.

| Recall | Precision |
|---|---|
| $33,3\%$ | $33,3\%$ |
| $25,0\%$ | $66,6\%$ |
| $20,0\%$ | $100,0\%$ |

Para se representar esses resultados em onze níveis de *Recall* utiliza-se da seguinte expressão:

$$P(r_j) = \max_{\forall r | r_j \leq r} P(r) \tag{5.1}$$

Onde:

- $r$ indica os valores de Recall da Tabela 5.2;
- $j \in \{0, 1, 2, ..., 10\}$ e é a referência a um dos onze níveis de Recall na representação nova, de modo que $r_5$ representa um nível de *Recall* igual a $50,0\%$;
- $P(r_j)$ é o valor de *Precision* correspondente ao nível de *Recall* $j$;
- $P(r)$ é o valor de *Precision* correspondente ao nível de *Recall* original, exibido na Tabela 5.2.

A Tabela 5.3 mostra os resultados para os onze níveis considerando o exemplo dado.

Tabela 5.3: Valores de Recall X Precision.

| Recall | Precision |
|---|---|
| $0\%$ | $33,3\%$ |
| $10\%$ | $33,3\%$ |
| $20\%$ | $33,3\%$ |
| $30\%$ | $33,3\%$ |
| $40\%$ | $25,0\%$ |
| $50\%$ | $25,0\%$ |
| $60\%$ | $25,0\%$ |
| $70\%$ | $20,0\%$ |
| $80\%$ | $20,0\%$ |
| $90\%$ | $20,0\%$ |
| $100\%$ | $20,0\%$ |

Aplicando então os conceitos explicados ao teste desenvolvido, o conjunto $A$ construído para uma palavra-chave pesquisada é formado pelos nós que possuem no texto a palavra em questão. Este conjunto é então ordenado de forma não crescente para cada uma das dez medidas analisadas. Tomou-se o cuidado de não identificar trechos de palavras como falso-positivo. Por exemplo, suponha a busca da palavra

*Algebra*. Caso um documento possua *Algebraic*, ele não é retornado. De forma semelhante, foi feito um tratamento para pesquisas compostas por mais de uma palavra, como *Maximum Likelihood Estimate*. Sobre o conjunto $R$, ele normalmente seria identificado por um grupo de especialistas. Na falta de um grupo que identifique grupos $R$ para as palavras-chave buscadas, obteve-se $R$ recorrendo às dez medidas locais, não apenas à medida que está sendo analisada, e que foi usada na ordenação de $A$. Foi dada a cada medida um "voto" à favor ou contra um potencial nó candidato a compôr $R$. Os passos a seguir explicam a construção de $R$:

1. Considere $X$ sendo o conjunto dos nós nos quais a palavra-chave pesquisada aparece no texto. Se $|X| \leq 10$, vá para o Passo 5.

2. Crie listas em ordem não crescente dos nós contidos em $X$, considerando como critério de ordenação cada uma das medidas locais (que são 10 no total).

3. Considere $Y$ sendo o conjunto de nós que aparecem entre as dez primeiras posições na maioria das dez listas ordenadas (por exemplo, em pelo menos seis) de acordo com as medidas.

4. Faça $R := Y$ e pare.

5. Faça $R := 0$ e pare.

Observe que a exigência de $|X| > 10$ é necessária para evitar o caso trivial onde $R = X$, que não permite diferenciar o desempenho das medidas locais entre si. Quando essa condição não é alcançada e o Passo 5 é executado, a consulta é desconsiderada.

Na obtenção dos resultados, que serão apresentados na Seção 5.2.2, optou-se por testar 100 palavras-chave na busca textual. O algoritmo recebeu como entrada as palavras obtidas do site da *MAS*. Para que uma palavra fosse computada no teste, a terminação precisava ocorrer no Passo 4 descrito. Desta forma, foram obtidas 100 pesquisas para todas as bibliotecas, com exceção da $D$, cuja execução terminou com apenas 14 palavras-chave consultadas. A execução da busca na $D$ ficou com essa pequena quantidade porque após entrar com as 300 primeiras palavras da *MAS*, apenas 14 estiveram presentes em mais de dez nós, sendo que para as outras bibliotecas, a lista das 300 primeiras já foi suficiente para conseguir as 100 palavras desejadas para o teste.

### 5.1.2 Resultados

A Figura 5.2 contém então os gráficos resultantes de *Precision-Recall*. Os valores são organizados em onze níveis, conforme explicado na Expressão 5.1.

De acordo com os dados na Figura 5.2, para se fazer a busca na porção matemática da *Wikipedia* ranqueando com das medidas locais calculadas no grafo $W$, a melhor escolha é o *PageRank*, seguido muito de perto pelas medidas *hub* e *authority*. Se a busca for feita no grafo $W'$ no entanto, a medida a ser escolhida é a *hub*. Observe contudo que o uso de $W'$ resulta em uma perda de precisão de cerca de 10% em comparação com $W$, não sendo portanto recomendado. Ainda analisando a *Wikipedia*, os dados mostram que se um usuário estiver disposto a examinar a lista de resposta dos nós $A$ passando o ponto no qual cerca de 70% do conjunto de documentos relevantes $R$ já terem sido recuperados, então o grau dos nós e a medida *Stress centrality* se tornam as preferidas na ordenação de $A$.

Voltando o foco para a *MathWorld* observa-se um cenário completamente diferente nos dados, uma vez que agora a melhor medida local para a ordenação de $A$ é a *Stress centrality* calculada no grafo $M$, independente do quanto de $R$ um usuário esteja disposto a recuperar. Se o usuário não desejar analisar mais do que 50% de $R$ no entanto, a medida *Betweenness centrality* se mostra igualmente eficiente. Ao se usar o grafo $M'$, e ao contrário da *Wikipedia*, apenas uma pequena perda de precisão ocorre em comparação com $M$ (cerca de 1 ou 2%), mas agora, a medida preferida de ranqueamento de $A$ foi a *Betweenness centrality*, seguida muito de perto pelos graus dos nós (ao se examinar até cerca de 60%).

Para a *DLMF*, a medida a ser escolhida é novamente a *Betweenness centrality* (para até cerca de 40% de $R$), embora os valores de *authority* se mostrem tão eficientes quanto (para até 20% de $R$), da mesma forma que a *Stress centrality* e o grau de entrada (para até 10% de $R$). Caso um usuário esteja disposto a examinar cerca de 50% ou mais de $R$, então a medida *Stress centrality* se torna a medida local a ser escolhida.

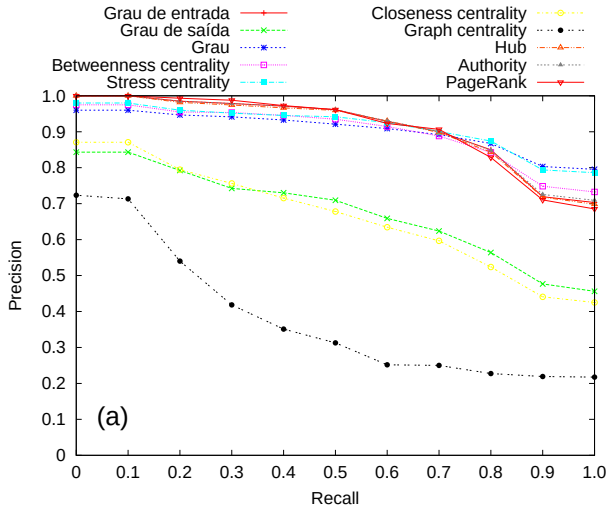



Figura 5.2: Gráficos de *Precision-Recall* para $W$ (*Wikipedia*; a), $W'$ (*Wikipedia*, *See-also*; b), $M$ (*MathWorld*; c), $M'$ (MathWorld, *See-also*; d), e $D$ (*DLMF*; e).

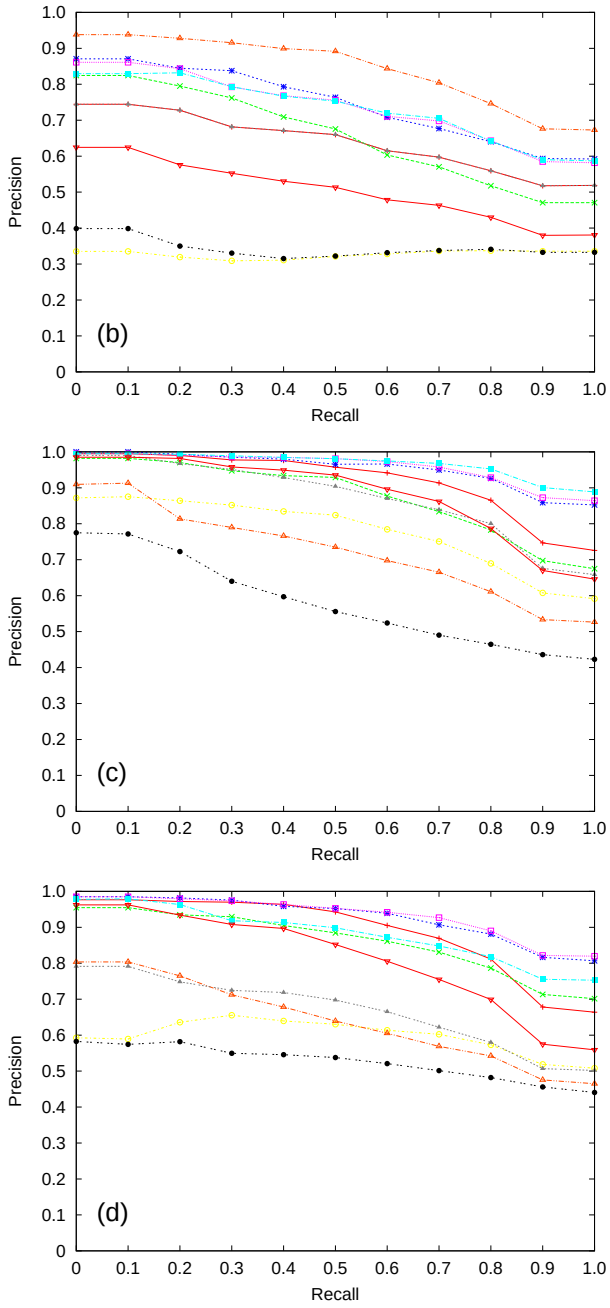



Figura 5.2: Continuação.

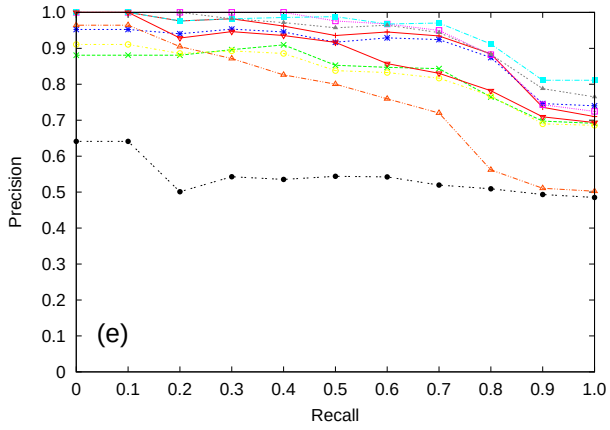



Figura 5.2: Continuação.

## 5.2 Busca Matemática

A seguir então serão mostrados os detalhes da implementação da ferramenta de busca matemática denominada *SearchOnMath* bem como os resultados do teste feito com usuários.

### 5.2.1 Metodologia

A busca matemática se baseia inicialmente em uma gramática para análise léxica. O objetivo é evitar situações como o exemplo citado na *Symbolab* no Capítulo 2, onde $x^2$ retorna resultados diferentes do $y^2$. No entendimento do trabalho, deseja-se que ambas as expressões acima retornem algo do tipo *VAR POT CONST*, indicando ser formada por variável, operador de potência e constante. Dessa forma, diferentes notações podem ser usadas com uma certa liberdade (no sentido de que há um consenso onde $x$, $y$ e $z$ sejam variáveis "sempre", $a$, $b$, e $c$ constantes, e assim por diante), e a qualidade da busca ainda será mantida.

A primeira gramática a ser criada foi para a *MathWorld*, e constituiu um grande desafio, uma vez que não existe disponível nenhum tipo de formalização da linguagem usada. O caminho então foi de fato olhar nas fórmulas da biblioteca para obter palavras reservadas comumente usadas e dessa forma construir a gramática. Primeiro criou-se regras para palavras chave conhecidas, e outras que se percebeu aparecerem bastante. Apesar da sintaxe ser parecida com TEX, muitas palavras reservadas do TEX não aparecem na *MathWorld*. Por exemplo, em TEX a representação de coeficiente binomial é dada por \binom ou em versões mais antigas por \choose. Na *MathWorld* a representação não ocorre com palavras reservadas nesse caso.

Depois de uma versão inicial ter sido construída, observou-se um problema: variáveis em multiplicações implícitas. Suponha o seguinte exemplo bastante simplificado de gramática, mostrado no Código 5.1.

Código 5.1: Problema da multiplicação implícita.

```
1 TRIG          ''cos''|''tan''
2
3 VAR           [a-z]
```

A regra *TRIG* é retornada caso uma função trigonométrica seja encontrada. Propositalmente não foi incluída a possibilidade do seno ("*sin*"). O que acontecerá então é que "*sin*" retornará como sendo três variáveis na forma *VAR VAR VAR*, ficando obviamente errada a interpretação. O problema então está no fato de não se usar operador para representar multiplicação, fazendo com que palavras não previstas sejam reconhecidas como sequências de variáveis.

O problema foi resolvido da seguinte forma: a gramática inicial criada então possuía todas as palavras reservadas encontradas agrupadas nas regras iniciais, e por último uma regra *VAR*, onde uma sequência de caracteres não reconhecida pelas regras iniciais casaria com ela (de forma semelhante ao que aconteceu no exemplo). Sempre que uma sequência de *VAR* de tamanho maior do que 2 aconteceu, a gramática gravou essa saída em um arquivo texto para ser analisado depois. Concluída a análise, o arquivo em questão possuía 8 000 linhas com repetições (por exemplo, $x_i$ em várias linhas), dentre as quais tinha variáveis em multiplicação, palavras reservadas, e pedaços de palavras reservadas. Os pedaços de palavras aconteciam no seguinte cenário. Ainda usando do exemplo do código 5.1, caso acontecesse um *"arccos"*, *"arc"* iria para o arquivo de saída, enquanto que *"cos"* teria casado com a regra *TRIG*. Os critérios adotados foram os seguintes: caso fosse uma palavra reservada na forma de função ou operador, ela era incluída na regra adequada da gramática, ou uma nova regra era criada, caso nenhuma delas fosse adequada. Comentários e rótulos de variáveis foram acrescentados em uma regra de *stop-words*, que são ignoradas na análise léxica. Por exemplo, uma variável encontrada como $x_{input}$ foi considerada apenas como $x$, e o *input* sendo incluído na regra de *stop-words*.

Com base no arquivo obtido então, cada uma das linhas foi checada para que se verificasse se a sequência de caracteres era de fato de variáveis em multiplicação (caso onde a interpretação de *"VAR"* ocorreu corretamente), ou se era de palavras reservadas, na forma de operadores ou nomes de função, ou ainda se era de comentários ou rótulos de variáveis. Na etapa de verificação muitas vezes as páginas que continham as palavras precisavam ser acessadas, para que o contexto fosse verificado e essa classificação pudesse ser feita.

Outro detalhe curioso que acontece na representação é a mistura de texto com codificação *HTML* nas fórmulas. Por exemplo, o alfabeto grego ainda possui interpretação textual na forma &*alpha*;, mas o alfabeto polaco é todo representado em códigos do tipo Ą. Ver o contexto da página então continuou sendo necessário. Outra ajuda na descoberta de possíveis símbolos veio da lista de caracteres encontrada no *FileFormat*[2].

Conforme pode-se perceber, a formalização da gramática incluiu a criação de diversas regras de modo a buscar a classificação das palavras reservadas. O critério adotado na criação das regras será discutido na subseção a seguir.

### Critério na Criação das Regras

Na criação das regras, muitas ocorreram de forma trivial, como por exemplo a regra *"SUM"* que engloba operadores relacionados a soma e subtração. Contudo, precisou-se definir um critério para a classificação de funções específicas, e portanto, menos conhecidas. O critério adotado foi o seguinte, sempre que uma função ou operador específico foi encontrado, foi executado um algoritmo que contava a ocorrência desse termo nas grandes áreas da Matemática definidas na *MathWorld* (Figura 3.1). Ao término da execução, verificou-se em qual área o termo tinha maior ocorrência em relação ao tamanho da área, na seguinte forma: $\max\left(\frac{t_i}{|Area_i|}\right), i \in Areas$, onde $t_i$ é a quantidade de vezes que o termo ocorre na área $i$, sendo $i$ verificado para todas as áreas. A relação que retorna maior valor fica definida então como sendo a área onde o termo encontrado mais influencia no conhecimento. Dessa forma, foram criadas regras para cada uma das grandes áreas. Suponha a regra $ALG$ como sendo a referencia para *Algebra*. Se uma função específica aparece mais em *Algebra*, ela será portanto incluída nessa regra.

### Gramática para TEX

Após o desenvolvimento da gramática para a *MathWorld*, foi iniciada então a construção de gramática semelhante para a linguagem TEX, usada nas bibliotecas *Wikipedia* e *DLMF*. Inicialmente pensou-se que seria mais fácil o desenvolvimento em virtude das palavras reservadas em TEX iniciarem com \, diferente da *MathWorld* onde nenhum caractere especial delimita o início de palavras reservadas. No entanto, constatou-se que os problemas se repetiram pelo seguinte, nem todas as funções são definidas na linguagem, como pode-se esperar. Assim, sempre que uma função ou palavra reservada não prevista é usada, utiliza-se mudança no estilo da fonte (cujo padrão é itálico) a fim de que seja diferenciada de sequência de caracteres. Foram observados nomes de função dentro dos

[2] `http://www.fileformat.info/info/unicode/category/Sm/list.htm`

mais variados delimitadores de mudança de estilo, podendo-se citar: \mathop, \operatorname, \mathbb, \mathbf, \boldsymbol, \mathit, \mathrm, \mathsf, \mathcal, \mathfrak, \scriptstyle, \displaystyle, \textstyle, \text, \mbox e \hbox.

Esta página[3] da *Wikipedia* ajudou bastante nos padrões usados na biblioteca. Apesar de sugerir o uso de alguns delimitadores de estilo acima citados para usos específicos, como \operatorname para funções, na prática se percebeu um uso bastante indiscriminado de todos eles, apresentando inclusive comentários e rótulos de variáveis, se tornando um problema semelhante ao observado na *MathWorld*. Encontrou-se inclusive o uso de funções já definidas de forma inadequada dentro de delimitadores, como por exemplo *sin* sendo usado sem \ e dentro de \mathrm, como pode ser visto na página[4]. Um trecho do código com o uso citado é mostrado na Figura 5.3.

```
alt=" = R \frac {d \theta} {dt} \left( -\mathrm{sin}\ \theta \ \mathbf{i} + \mathrm{cos}
```

Figura 5.3: Trecho do código *HTML* contendo função seno e cosseno dentro de delimitador \mathrm.

A partir então do exemplo acima citado, pode-se concluir o seguinte: Não se pode esperar nem que as expressões reservadas previstas na linguagem tenham sido iniciadas adequadamente com \. Assim, precisou-se de definir um critério para recuperação e tratamento de textos que estejam dentro desses delimitadores. Os tratamentos feitos foram os seguintes nas equações:

- Ao encontrar um delimitador sem {}, ele foi retirado do código: exemplo \mathbf $S$ originou $S$;
- Ao encontrar um delimitador com {}:
  - Se no seu conteúdo tiver comandos TEX reconhecidos: exemplo \mathbf{$x$ \in \Omega} originou $x$ \in \Omega;
  - Se não tiver comandos TEX reconhecidos:
    - Tamanho menor ou igual a 3: exemplo \text{var} originou var
    - Senão: exemplo \text{such that} originou “” (ou seja, nada).

Depois desse tratamento, as fórmulas passam pela análise léxica, de forma semelhante ao que se fez com a *MathWorld*. Toda sequência de caracteres identificada como sequência de variável foi gravada em um arquivo texto, que depois foi todo olhado linha-a-linha a fim de se constatar se nenhuma palavra reservada havia sido interpretada de forma errada (o tamanho final foi de 2 300 linhas aproximadamente sem repetições).

[3] http://en.wikipedia.org/wiki/Help:Displaying_a_formula
[4] http://en.wikipedia.org/wiki/Centripetal_force

### Unificação das Gramáticas

Após perceber que não se podia esperar nem que palavras reservadas em TEX tivessem sido corretamente representadas começando com \, verificou-se então que as gramáticas inicialmente desenvolvidas separadamente para *MathWorld* e TEX poderiam ser unificadas. As regras para palavras reservadas ficam da seguinte forma, vista no Código 5.2:

Código 5.2: Exemplo de regra unificada.

```
1 TRIG  ‘‘\\’’?[Cc]‘‘os’’|‘‘\\’’?[Tt]‘‘an’’|‘‘\\’’?[Ss]‘‘in’’
```

A sequência "\\" indica a existência de uma barra apenas, precedida por ?, indicando que pode ocorrer zero ou uma vez. Depois a sequência [Cc] cobre a possibilidade de começar com maiúscula ou minúscula, encerrando-se finalmente pelo restante da palavra. Um último detalhe a ser citado está no fato de que considera-se a possibilidade de palavras reservadas começarem com maiúscula ou minúscula, conforme citado, mas sequências de letras são diferenciadas de maiúsculo e minúsculo. Observa-se por exemplo que $a$ é comumente encontrada representando constante, enquanto que $A$ muitas vezes representa lado de figura geométrica. Então, essa diferenciação se mantém com regras diferentes.

Após a unificação das gramáticas o resultado foi cerca de 900 palavras reservadas organizadas em 63 regras, e quase 800 *stop-words* a serem ignoradas. Apenas a título de curiosidade, se encontram *stop-words* como *milk*, *beer*, dentre outras. A classificação resultante para as palavras reservadas pode ser vista no Apêndice D.

### Implementação da Busca

Após obtidos os *tokens* corretamente com a análise léxica descrita na subseção anterior, a busca foi implementada usando-se o tradicional conceito de distância de *Levenshtein* (ou distância de edição), que consiste no número mínimo de operações necessárias (considerando-se inserção, remoção ou edição de letras) para transformar uma *String* em outra [53].

Como exemplo, considere as palavras *kitten* e *sitting*, cuja distância de edição é igual a 3. As operações são as seguintes:

1. *kitten*
2. *sitten* (substituição de $k$ por $s$)
3. *sittin* (substituição de $e$ por $i$)
4. *sitting* (inserção de $g$)

Dessa forma, considerando que o analisador léxico já foi executado para as expressões contidas no banco de dados, e que o seu resultado já tenha sido igualmente armazenado, em uma nova busca por fórmula, a expressão informada pelo usuário passa também pelo analisador léxico, e depois, compara-se a *String* resultante de *tokens*, com as existentes no banco da dados das fórmulas. Basicamente, quanto a menor distância de edição, mais semelhantes são as duas fórmulas comparadas.

Apenas como efeito ilustrativo, a fórmula do produto de Euler

$$\sum_{k=1}^{\infty} \frac{1}{k^s} = \prod_{p \text{ prime}} \frac{1}{1 - p^{-s}} \tag{5.2}$$

gera como resultado a seguinte *String*:

*FSUM BOT VAR RELAT CONST POW INF MULT CONST VAR POW SYMBOL RELAT FMULT BOT PROB LIN MULT CONST CONST SUM PROB POW SUM SYMBOL.*

Utilizando-se então dessa técnica, observou-se que dada uma equação na forma *termo* = *termo*, se o usuário entrasse com apenas um dos termos mostrados (à esquerda ou direita da igualdade), e no banco de dados tivesse apenas a equação completa, a distância de edição resultante ainda seria alta. Por exemplo, seria como um usuário informar somente

$$\prod_{p \text{ prime}} \frac{1}{1 - p^{-s}} \tag{5.3}$$

da fórmula do produto de Euler.

Assim, uma forma simples de minimizar esse problema foi inserir no banco de dados também registros contendo os *tokens* dos termos de equações que possuam operador relacional. Um outro problema que pode acontecer está relacionado à ordem dos termos. Por exemplo, uma pessoa pode buscar por $\pi = \pi P$, e no banco pode estar armazenado ao contrário ($\pi P = \pi$). Os passos a seguir explicam as medidas tomadas com o objetivo de melhorar essas questões:

- Ao se encontrar uma fórmula com operador relacional (da forma $A$ *RELACIONAL* $B$ *RELACIONAL* $C$, com $A$, $B$ e $C$ representando os termos): Insere-se no banco os *tokens* relacionados aos termos $A$, $B$ e $C$ separadamente. A Figura 5.4 mostra um exemplo, com $A$, $B$ e $C$ marcados. Observe que como $A$ e $C$ possuem os mesmos *tokens*, a inserção é feita apenas uma vez;

  - Caso a quantidade de operadores relacionais seja menor ou igual a dois, insere-se no banco também os *tokens* referentes às permutações da fórmula:

    1. $A$ *RELACIONAL* $B$ *RELACIONAL* $C$ (original)

2. $A \quad RELACIONAL \ C \ RELACIONAL \ B$
3. $B \quad RELACIONAL \ A \ RELACIONAL \ C$
4. $B \quad RELACIONAL \ C \ RELACIONAL \ A$
5. $C \quad RELACIONAL \ A \ RELACIONAL \ B$
6. $C \quad RELACIONAL \ B \ RELACIONAL \ A$

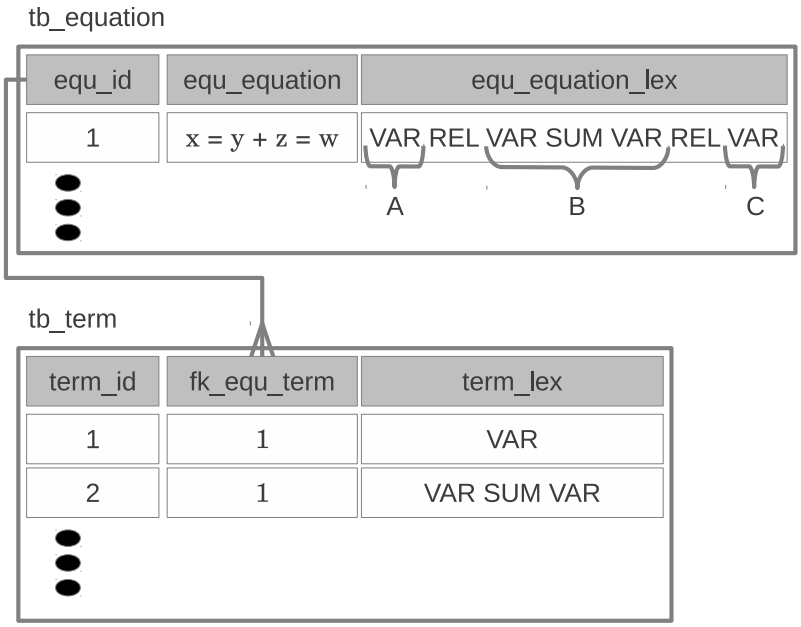



Figura 5.4: Exemplo de inserção de termos na *tb_term*.

Com o acréscimo dos termos e permutações então, a quantidade de expressões aumentou consideravelmente, aumentando também a complexidade da busca. Para tratar esse problema, criou-se um conjunto de 6 características (*features*). Por exemplo, um usuário que informe uma fórmula que contenha a constante "$e$" pode significar para a ferramenta uma característica forte o suficiente para tentar encontrar apenas por fórmulas semelhantes que também contenham essa constante. Dessa forma, reduz-se a quantidade de registros a serem comparados no banco de dados. Os seguintes *tokens* foram agrupados em características:

- *feat1*: *GREEK*, *GAMA*, *HEBREW*, *POLAC*, *DIRAC*, *INF*, *CONST*, *PI*, *UPPER*, *VAR*;
- *feat2*: *LOG*, *POW*, *SQRT*;
- *feat3*: *RELAT*, *ARROW*;
- *feat4*: *E*;
- *feat5*: *TRIG*, *INVTG*, *HYBOLIC*;

- *feat6*: *CALC*, *PART*, *NABLA*, *DERIV*, *INTEGR*, *LIMIT*;

Um exemplo simples da aplicação das características, suponha uma busca por $e^{i\pi}+1=0$, onde os *tokens* retornados são *E VAR PI SUM CONST RELAT CONST*. Esses *tokens* ativam as seguintes características (considerando a ordem em que aparecem na fórmula): *feat4*, *feat1*, *feat3*. Assim, o comando para selecionar fórmulas no banco de dados é direcionado somente àquelas onde essas características ocorrem.

O ranqueamento dos resultados obtidos então segue primeiro a distância de edição. Ou seja, quanto menor a distância, melhor ranqueado estará o resultado. Caso duas páginas empatem nesse critério, o desempate na ordem é feito usando a medida *Stress centrality*, que se mostrou bastante eficiente nos testes de ranqueamento para a busca textual.

### Formalização do Teste

Existem na literatura diversos critérios definidos que podem servir como base em testes de ferramentas de busca [54]:

- relevância: maioria dos testes feitos se concentra nessa categoria. Isso porque se os usuários souberem por exemplo, que o *Google* e demais ferramentas concorrentes estão obtendo resultados bastante semelhantes quanto à relevância, os usuários vão querer cada vez mais experimentar as outras ferramentas. Os resultados de *Precision and Recall* se encaixam nessa categoria;

- ranqueamento: analisa a capacidade da ferramenta de colocar resultados importantes primeiro;

- satisfação do usuário: este tipo de teste tem um escopo bastante abrangente, e por ser feito com usuários, tem característica subjetiva, sendo intimamente ligado ao contexto da consulta, ao estado emocional dos usuários e ao tempo dedicado. No entanto, diversas referências citam que este tipo de teste resulta em uma avaliação mais realista do desempenho do sistema[55–58];

- tamanho/cobertura da Web: indica qual a ferramenta de busca cobre a maior porção da Web. Contudo, não possui relação direta com a qualidade dos resultados;

- dinamismo dos resultados da busca: mede qual ferramenta se adapta melhor ao dinamismo da Web. Também não possui relação direta com a qualidade dos resultados;

- itens pouco conhecidos/relevantes: teste bastante comum que mede a capacidade da ferramenta em exibir nos resultados uma determinada página pessoal;

- tópico/domínio específico: destaca o desempenho da ferramenta em consultas sobre algum tópico ou domínio específico;

- avaliação automática: gera automaticamente os resultados desejados, e tem como vantagem a escalabilidade.

Dentre as classificações acima citadas, o teste desenvolvido encaixa-se nas categorias de relevância, satisfação do usuário, e tópico/domínio específico.

No teste foram selecionadas 30 fórmulas matemáticas clássicas, obtidas em sites relacionados[5][6] e livros [59, 60]. As fórmulas usadas estão listadas no Apêndice B. Depois, cada uma das fórmulas foi buscada em ambas as ferramentas: a *SearchOnMath* aqui proposta, e na *Symbolab*. Foram selecionados os 5 primeiros resultados de cada ferramenta, e agrupados em uma folha contendo então a fórmula buscada, e os resultados retornados por cada uma (colocados fora de ordem, para que o usuário não fosse influenciado a marcar os primeiros resultados como sendo melhores). A escolha por 5 resultados foi com o intuito de não deixar o teste muito extenso e cansativo de ser feito. Cada usuário que realizou o teste recebeu um conjunto de 5 fórmulas selecionadas aleatoriamente dentre as 30. Na seleção aleatória das fórmulas garantiu-se que cada uma das 30 fórmulas estivesse presente em pelo menos 5 testes, e em não mais do que 7. Detalhes sobre os usuários serão apresentados na próxima seção. Para cada resposta retornada pelas ferramentas, o usuário considerou a afirmação
"A página/documento indicada(o) é útil para compreender a fórmula procurada, ou apresenta fórmula semelhante, onde as mudanças ocorrem por manipulações algébricas simples ou troca de notação."
e escolheu uma dentre as seguintes alternativas:

a) Concordo $-->$ (caso a afirmação seja verdadeira para a resposta analisada);

b) Indiferente $-->$ (não é possível se posicionar à favor ou contra com relação à afirmação);

c) Discordo $-->$ (caso a afirmação seja falsa para a resposta analisada).

Observe portanto que sendo 40 usuários, testando 5 fórmulas cada, com 5 resultados para cada fórmula, somando-se tudo, chega-se a 1 000 resultados avaliados para cada ferramenta.

A seguir é apresentada uma folha no formato aplicado no teste.

[5] `http://www.math.utah.edu/~pa/math/equations/equations.html` (acesso em 01/02/2013)
[6] `http://nargaque.com/2011/10/05/10-mind-blowing-mathematical-equations/` (acesso em 01/02/2013)

Fórmula buscada: $\sum_{k=1}^{\infty} \frac{1}{k^s} = \prod_{p \text{ prime}} \frac{1}{1-p^{-s}}$

"A página/documento indicada(o) é útil para compreender a fórmula procurada, ou apresenta fórmula semelhante, onde as mudanças ocorrem por manipulações algébricas simples ou troca de notação".

Tabela 5.4: Resultados da Ferramenta 1.

| Título | Expressão(ões) | URL | Resposta |
|---|---|---|---|
| Honors Diff... | $\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$ | http://bit.ly/VITkFm | a) Concordo; b) Indiferente; c) Discordo; |
| Complex Var... | $\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z}$<br>$\sum_{n=1}^{k} \frac{1}{n^z} = \frac{1}{k^{z-1}} +$<br>$z \sum_{n=1}^{k-1} \int_n^{n+1} (t)\, t^{-z-1} dt$ | http://bit.ly/13bfZPa | a) Concordo; b) Indiferente; c) Discordo; |
| What is this math... | $\zeta(s) = \sum_{k=1}^{\infty} \frac{1}{k^s}$ | http://bit.ly/XartND | a) Concordo; b) Indiferente; c) Discordo; |
| Analytic number... | $\sum_{n=1}^{\infty} \frac{1}{n^s} > 1\, (p \in \ primenumber)$<br>$f(s) = \sum_{n=1}^{\infty} a_n n^{-s}$ | http://bit.ly/Yt2MLG | a) Concordo; b) Indiferente; c) Discordo; |
| Series math... | $\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$<br>$\sum_{n=1}^{\infty} \frac{a_n}{n^s}$ | http://bit.ly/13bgrgk | a) Concordo; b) Indiferente; c) Discordo; |

Tabela 5.5: Resultados da Ferramenta 2.

| Título | Expressão(ões) | URL | Resposta |
|---|---|---|---|
| Group_theory | $\sum_{n \geq 1} \frac{1}{n^s} = \prod_{p \text{ prime}} \frac{1}{1-p^{-s}}$ | http://bit.ly/6KPZtI | a) Concordo; b) Indiferente; c) Discordo; |
| Analytic_number... | $\sum_{n=1}^{\infty} \frac{1}{n^s} = \prod_{p}^{\infty} \frac{1}{1-p^{-s}}$<br>for $s > 1$ (p is prime number) | http://bit.ly/fqIUJW | a) Concordo; b) Indiferente; c) Discordo; |
| Proof_of_the_Euler... | $\sum_{n=1}^{\infty} \frac{1}{n^s} = \prod_{p \text{ prime}} \frac{1}{1-p^{-s}}$ | http://bit.ly/mHASn | a) Concordo; b) Indiferente; c) Discordo; |
| Wallis_product | $\sum_{m=1}^{\infty} \frac{1}{m^2} \sum_{n=1}^{\infty} \frac{1}{n^2}$ | http://bit.ly/g8T4tP | a) Concordo; b) Indiferente; c) Discordo; |
| Riemann_zeta... | $\sum_{n=1}^{\infty} \frac{1}{n^s} = \prod_{p \text{ prime}} \frac{1}{1-p^{-s}}$ | http://bit.ly/3cqDhO | a) Concordo; b) Indiferente; c) Discordo; |

### Instruções, Condução do Teste e Perfil dos Usuários

O teste foi feito com 40 usuários, que ou compareceram presencialmente em um laboratório do Curso de Ciência da Computação da UNIFAL-MG, ou foram conduzidos à distância com as instruções sendo passadas via *Skype*. Todos os usuários receberam antes do início a Folha de Instruções mostrada no Apêndice C.

Os usuários selecionados foram: estudantes de graduação/mestrado de cursos na área de Ciências Exatas; Professores doutores com formação em programas relacionados a Engenharia de Sistemas /Matemática; Profissionais formados em Ciência da Computação ou cursos afins, que atuam no mercado como analistas ou programadores.

## 5.2.2 Resultados

Nesta seção serão exibidos então os resultados obtidos nos testes com usuários, que comparam as ferramentas *SearchOnMath* e *Symbolab*. O capítulo encerra com uma visão da arquitetura da ferramenta desenvolvida, comparando-se com os requisitos de usuários levantados no trabalho [14].

### Testes com Usuário

Os resultados dos testes aplicados estão exibidos na Figura 5.5, mostrando então a porcentagem de escolhas feitas pelos usuários considerando-se os 1 000 itens assinalados para as possibilidades de resposta: a) concordo; b) indiferente; c) discordo;

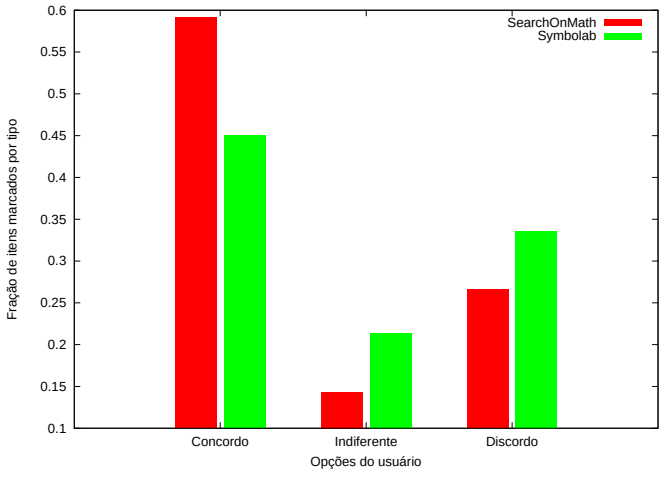



Figura 5.5: Quantidade de itens marcados no total dos testes.

Conforme pode-se observar no gráfico, a ferramenta *SearchOnMath* proposta no trabalho apresentou um índice de aceitação nos itens retornados na busca bem próximo a 60% dentre os 1 000 avaliados. Essa relação se reduz a 45% para a ferramenta *Symbolab*.

Na Figura 5.6 a seguir é mostrada a fração de concordância marcada para cada fórmula do teste. Apenas lembrando, o conjunto de fórmulas está disponível no Apêndice B, e na mesma ordem mostrada nos gráficos.

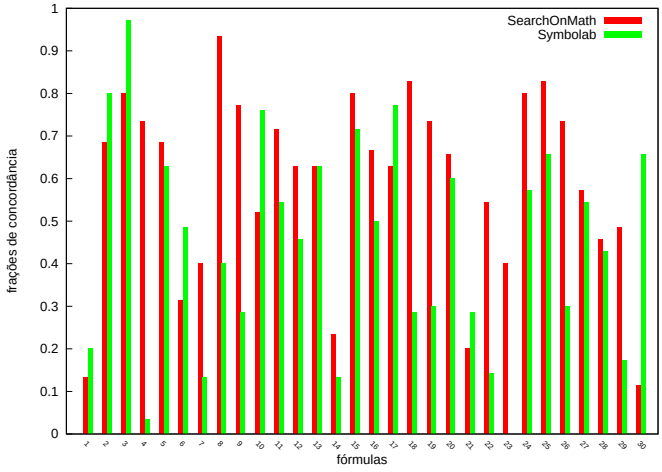



Figura 5.6: Fração marcada de cada opção para a ferramenta *SearchOnMath*.

Conforme pode-se ver, a ferramenta proposta *SearchOnMath* apresentou um desempenho ruim nas fórmulas 1, 6, 7, 14, 21 e 30. Nas fórmulas 6, 7, 14 e 21 aconteceu geralmente de um ou dois documentos retornados entre as 5 respostas conter fórmula semelhante, e os outros três não. Por isso o índice de acerto ficou entre $0,20$ e $0,40$. No caso da fórmula 30, o segundo resultado retornado pela *SearchOnMath* coincide com a fórmula buscada, mas com inversão nos termos da igualdade. Como pode-se ver, nem todos os usuários perceberam, ou o índice de concordância desse caso chegaria em $0,20$, considerando que das 5 respostas para essa busca, todos os usuários marcassem que uma atende.

Uma das menores fórmulas, que é a fórmula 1 ($\pi = \frac{c}{d}$) no entanto teve um índice de acerto muito ruim, e de fato, nenhum dos resultados retornados encontrou a fórmula. Esse caso em especial revela já uma melhoria considerável a ser feita na ferramenta. Conforme já explicado, o algoritmo de distância de edição clássico que foi usado faz a análise caractere a caractere nas *Strings* (no caso, as *Strings* contendo os tokens da fórmula a ser buscada e das fórmulas no banco de dados). Assim, o que ocorre é que a fórmula informada no teste foi com o numerador da fração sendo $c$ em minúsculo, cujo *token* retornado será *CONST*, enquanto que na *Wikipedia*, a fórmula existe, mas representada como $C$ em maiúsculo, que retorna como token *UPPER* (esses *tokens* podem ser vistos no Apêndice D). Com isso, o custo de edição de *CONST* para *UPPER* é igual a 5, um custo bastante elevado para uma fórmula tão pequena, fazendo com que o resultado certo não apareça entre os 5 primeiros. Logo, uma importante contribuição de trabalho futuro a ser implementada é modificar o algoritmo de distância de edição para considerar o custo

de trocar *tokens* ao invés de caracteres, e a troca de *tokens* ter custos diferentes. Por exemplo, trocar *VAR* por *CONST* ou *UPPER* deve ter um custo menor do que trocar por uma função dentro de *CALC*. Essa modificação não somente vai melhorar os resultados de fórmulas pequenas como essa, como também das fórmulas maiores, onde a ferramenta já apresentou bons resultados.

Para a ferramenta *Symbolab*, as fórmulas que ficaram abaixo de $0,20$ de concordância foram 4, 7, 14, 22, 23 e 29. Se considerar ainda as que estão abaixo de $0,40$, acrescenta-se as fórmulas 9, 18, 19, 21 e 26. De forma semelhante, as fórmulas abaixo de $0,20$ representam aquelas onde foram identificados pelos usuários uma ou nenhuma alternativa plausível. O detalhe para a fórmula 23, onde nenhum resultado foi retornado. Ao se analisar o problema, constatou-se que a fórmula está representada de forma um pouco diferente na *Wikipedia*. Então, fez-se um teste na *Symbolab* trocando a fórmula pelo modo como está escrita na *Wikipedia*. Os resultados dessa nova busca retornaram duas expressões, mas nenhuma delas era a encontrada na página. Nos testes não foram feitas mudanças na forma de escrita das fórmulas, se mantendo assim zero resultados para essa. Os usuários foram orientados a considerar a opção *indiferente* nesse caso.

### Arquitetura da Ferramenta

A ferramenta *SearchOnMath* já se encontra em funcionamento. Serão destacados aqui os detalhes referentes à implementação, onde pode-se citar a interface com o usuário, tratamentos feitos na fórmula informada, e o mecanismo de busca em si.

A aplicação Web foi desenvolvida em *JSF - Java Server Faces*, com a lógica de negócio implementada internamente em *Java*, a gramática em *C* gerada com o *Flex* e o banco de dados escolhido foi o *MySQL*. A ferramenta executa em um servidor com processador *Core 2 Quad*, 8 GB de memória RAM, 500 GB de HD, e sistema operacional Linux Debian 6.

No desenvolvimento da primeira versão da interface, considerou-se como referência o estudo [14], onde 13 voluntários foram entrevistados, sendo o grupo composto por dois estudantes universitários, sete estudantes já formados, um professor e três bibliotecários, todos associados ao departamento de matemática local.

Os requisitos levantados para a interface de entrada dos dados para ferramenta matemáticas foram:

- "Eu acho que LaTeX seria uma boa escolha uma vez que todos usamos na escrita de artigos. Às vezes eu uso até para me comunicar com meus amigos no MSN."

- "Seria bom se eu pudesse visualizar a expressão que escrevi."

Com base nesses relatos, a interface de entrada apresentada na Figura 5.7 foi desenvolvida. Conforme pode-se verificar, a mesma é composta por dois campos, sendo o superior onde o usuário informa a fórmula em TEX, enquanto que em baixo a fórmula é renderizada em tempo real. Na renderização foi usada a biblioteca *MathJax*[7], que tem como vantagem produzir o desenho das fórmulas de forma bastante agradável e compatível com os mais populares navegadores existentes, ao contrário da linguagem *MathML*.

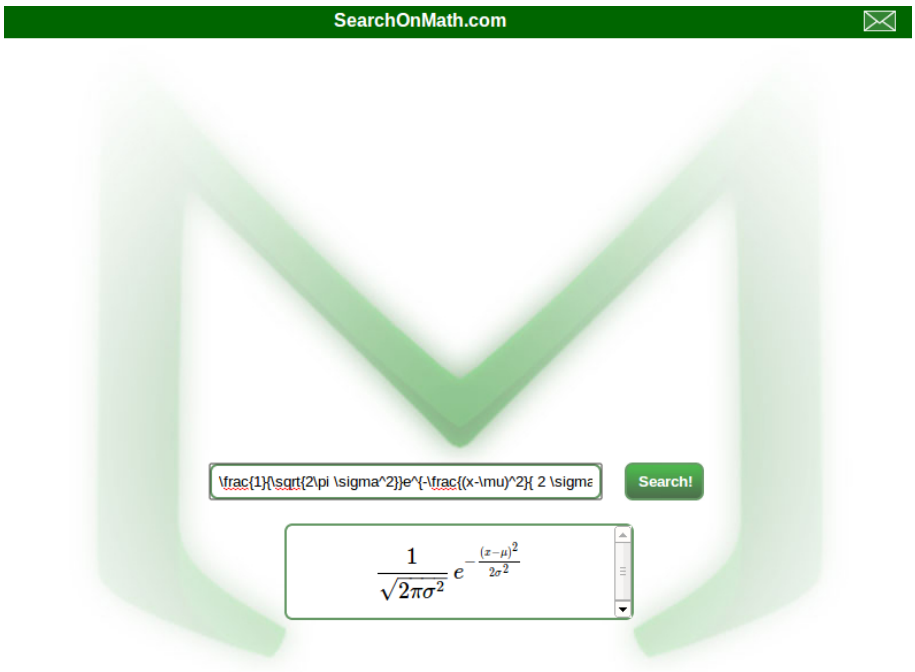



Figura 5.7: Tela inicial da ferramenta *SearchOnMath*.

Pelo grupo reduzido, e extremamente especializado no qual os requisitos foram baseados[14], fica claro que impôr a um usuário não familiarizado com LATEX que ele informe as expressões nessa linguagem tende a ser uma barreira no uso da ferramenta. No entanto, está em andamento um estudo cujo objetivo é identificar quais os botões são necessários a fim de que a vasta coleção de fórmulas contidas no banco possam ser reproduzidas em sua forma original ou semelhante. Essa é uma das prioridades a ser citada em trabalhos futuros.

Quando o usuário informa então a expressão, os mesmos passos citados no tratamento das equações em TEX na Seção 5.2.1 são repetidos aqui. Após esse tratamento, inicia-se o motor de busca, que primeiramente executa o analisador léxico, e em seguida, calcula a distância de *Levenshtein* entre os *tokens* resultantes da fórmula do usuário e os *tokens* das fórmulas do banco de dados. Uma lista ordenada armazena então os resultados de menor distância e os exibe. A exibição ocorre então

[7] `http://www.mathjax.org/`

em ordem não decrescente da distância. Em caso de empate na distância, as páginas são ordenadas segundo a *Stress centrality*, conforme já mencionado.

A execução do analisador léxico (executável obtido à partir do código em *C*) de dentro do código *Java* é feito através do código mostrado em 5.3. Considere que a fórmula informada pelo usuário tenha sido gravada previamente no arquivo *entrada.txt*. Após a execução, os *tokens* foram retornados dentro da variável *line*.

Código 5.3: Execução do analisador de dentro do código em Java.

```
1          String cmd = "./gramatica entrada.txt";
2          Runtime run = Runtime.getRuntime();
3          Process pr = run.exec(cmd);
4          pr.waitFor();
5          BufferedReader buf = new BufferedReader(new InputStreamReader(
               pr.getInputStream()));
6          String line;
7          while ((line = buf.readLine()) != null) {
8                  equacaoBusca = equacaoBusca + line;
9          }
```

Após fazer a busca, a tela de resultados a ser exibida é semelhante à apresentada na Figura 5.8, exibindo portanto o título da página, a fórmula encontrada e um breve texto retirado da própria página.

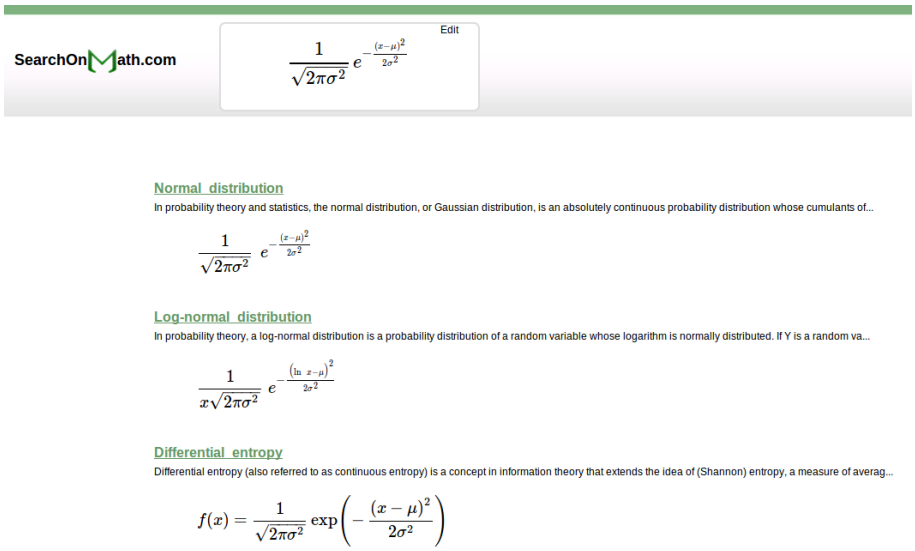



Figura 5.8: Tela exemplo com alguns resultados apenas em caráter ilustrativo.

# Capítulo 6

# Conclusões

No que diz respeito à busca matemática, dois estudos foram conduzidos, sendo o primeiro relacionado a busca textual, onde foi analisada a eficiência das medidas locais calculadas na realização da ordenação dos resultados; e o segundo a busca por fórmulas matemáticas, onde uma gramática para um analisador léxico composta por aproximadamente 900 termos foi desenvolvida e colocada em teste usando como base na obtenção da similaridade entre fórmulas a distância de *Levenshtein*, calculada à partir dos tokens gerados pelo analisador léxico.

Dentre os principais resultados desse trabalho estão a presença de componente fortemente conexa gigante (GSCC), que na maior parte dos casos, apresentam uma fração de nós compondo a componente substancialmente maior do que a apresentada em estudos semelhantes para a Web; indicativos do fenômeno de mundo-pequeno; ausência de sinais claros de assortatividade nos padrões de link, bem como de leis de potência descrevendo as distribuições das medidas locais. Foram mostrados ainda que a maior parte dos grafos estudados são bastante resistentes à perda acidental de material, embora naturalmente menos quando se considerou a destruição intencional das páginas.

Nos resultados relacionados à busca textual sobre a ocorrência de palavras-chave específicas, apenas para a *Wikipedia* as medidas clássicas de *PageRank* e àquelas relacionadas ao *HITS* se saíram melhor. Para as menores *MathWorld* e *DLMF*, os resultados melhores foram alcançados por medidas até então não consideradas para essa finalidade, com destaque para *Stress centrality*, *Betweenness centrality* e o grau dos nós.

Acredita-se que muitos desses resultados possam ser atribuídos a uma propriedade chave em todas as bibliotecas. Ao contrário do que acontece em muitos outros domínios, onde intangíveis como afinidade ou popularidade ditam o estabelecimento de ligações, na construção dessas bibliotecas o que é levado em consideração é o quão bem informado cada contribuidor é sobre: i) o material a ser tratado, e ii) como o material produzido se relaciona a outros tópicos. O fato provável dessa

distinção refletir nas propriedades estruturais medidas do grafo, apesar da grande quantidade de contribuidores, é bastante notável.

Na busca por fórmulas, a ferramenta desenvolvida, *SearchOnMath* teve o seu desempenho comparado com a *Symbolab* segundo critério de relevância avaliado por um grupo formado por 40 usuários, que verificaram a eficiência de ambas na busca por 30 fórmulas clássicas de diversas áreas. Os resultados mostraram que a primeira versão da ferramenta proposta já apresenta um índice de acerto de quase 60% em comparação com o índice da *Symbolab* que foi de 45%. Além disso, os testes permitiram identificar um problema encontrado ao se usar o algoritmo clássico de distância de edição, onde as operações incidem sobre caracteres.

## 6.1 Trabalhos Futuros

Vários trabalhos futuros podem ser citados ainda para a área de busca matemática:

- implementar o algoritmo de distância de edição com pesos e que trabalhe em nível de *tokens*;
- expansão dos estudos e resultados aqui apresentados a outras fontes matemáticas disponíveis na Internet, como a `planetmath.org` e a `mathoverflow.net`, que aumentariam ainda mais o domínio de busca da ferramenta apresentada, e ambas as fontes citadas possuem conteúdo em TEX tendo portanto a vantagem de já se ter o analisador léxico bastante desenvolvido;
- uma busca avançada por áreas na matemática pode ser proposta, usando da classificação já armazenada na tb_category (mostrada no Apêndice A). Por exemplo, um usuário pode ter a opção de buscar por fórmulas só na área de Teoria dos Números, ou Probabilidade;
- As características (*features*) mostradas no Capítulo 5 também podem ser melhoradas, estudando por exemplo quantas fórmulas são cobertas no banco de dados para cada uma, permitindo ajustes que melhorarão o tempo de resposta;
- desenvolvimento de gramática para *MathML*, que é outro padrão usado comumente na Internet;
- algoritmos para padronização de expressões. Exemplificando, fórmulas iguais, mas representadas em TEX e na linguagem do *MathWorld*, podem retornar sequências de tokens diferentes, dada a simplicidade da linguagem adotada no *MathWorld* em comparação com a linguagem TEX;

- desenvolvimento de um analisador sintático a ser atrelado no léxico, de modo a se obter a árvore sintática abstrata das expressões, podendo à partir daí se testar outros métodos na descoberta de fórmulas semelhantes, além da já usada distância de Levenshtein;

- desenvolvimento de interface com botões, com testes de usabilidade, deixando assim a ferramenta independente de linguagem;

- enfrentou-se alguns problemas no uso do *WebCrawler* em virtude de algumas especificidades das bibliotecas. O desenvolvimento de um específico para esse uso é outra proposta, e que já está em desenvolvimento.

- realização de testes para avaliar o mecanismo de ranqueamento com a busca por fórmulas;

- desenvolvimento de estudos para se constatar a periodicidade com que a base de dados deve ser atualizada, uma vez que por se tratarem de bibliotecas, a atualização acontece em uma taxa menor do que em outros tipos de sites cobertos pelas ferramentas de busca textuais de uso geral (por exemplo, um portal de notícias);

# Referências Bibliográficas


[1] SHATNAWI, M., YOUSSEF, A. "Equivalence detection using parse-tree normalization for math search", *2nd International Conference on Digital Information Management*, v. 2, pp. 643–648, 2007.

[2] KOHLHASE, M., SUCAN, I. "A Search Engine for Mathematical Formulae", *Lecture Notes in Computer Science - Artificial Intelligence and Symbolic Computation*, v. 4120, pp. 241–253, 2006.

[3] BAEZA-YATES, R., RIBEIRO-NETO, B. *Modern Information Retrieval*. Second ed. Harlow, UK, Addison Wesley, 2011.

[4] BRIN, S., PAGE, L. "The anatomy of a large-scale hypertextual Web search engine". In: *WWW7: Proceedings of the seventh international conference on World Wide Web 7*, pp. 107–117, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V. doi: http://dx.doi.org/10.1016/S0169-7552(98)00110-X.

[5] ORESKOVIC, A., CHANG, R. "Google widens lead in U.S. searches: comScore". `http://www.reuters.com/article/idUSTRE53E6YT20090415`, 2009.

[6] ASPERTI, A., GUIDI, F., COEN, C. S., et˜al. "A Content Based Mathematical Search Engine: Whelp", *Lecture Notes in Computer Science - Types for Proofs and Programs*, v. 3839, pp. 17–32, 2006.

[7] AVNY, M., ARNON, A., ALYSHAYEV, L. "Symbolab Scientific Equation Search". `http://symbolab.com/`, 2013.

[8] YOUSSEF, A. "Roles of Math Search in Mathematics", *Lecture Notes in Computer Science - Mathematical Knowledge Management*, v. 4108, pp. 2–16, 2006.

[9] SINGH, S. *O último teorema de Fermat*. Record, 2008.

[10] ABRAMOWITZ, M., STEGUN, I. A. *Handbook of Mathematical Functions*. New York, NY, Dover Publications, 1965.

[11] YANG, R., KALNIS, P., TUNG, A. K. H. "Similarity evaluation on tree-structured data". In: *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pp. 754–765, New York, NY, USA, 2005. ACM. ISBN: 1-59593-060-4. doi: http://doi.acm.org/10.1145/1066157.1066243.

[12] MINER, R., MUNAVALLI, R. "An approach to mathematical search through query formulation and data normalization", *Towards Mechanized Mathematical Assistants*, pp. 342–355, 2007.

[13] LIBBRECHT, P., MELIS, E. "Methods to access and retrieve mathematical content in activemath", *Mathematical Software-ICMS 2006*, pp. 331–342, 2006.

[14] ZHAO, J., KAN, M., THENG, Y. L. "Math information retrieval: user requirements and prototype implementation". In: *Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*, pp. 187–196. ACM, 2008.

[15] GRAF, P. "Substitution tree indexing", *Lecture Notes in Computer Science*, v. 914, pp. 117–131, 1995. ISSN: 0302-9743. doi: 10.1007/3-540-59200-8_52.

[16] KNUTH, D. E. *The TeXbook*. 1st ed. Cambridge, Massachusetts, USA, Addison-Wesley Professional, 1984.

[17] ROCHE, X. "HTTrack Website Copier". `http://www.httrack.com/`, 2012.

[18] FIELDING, R., GETTYS, J., MOGUL, J., et˜al. *Request for Comments: 2616*. Relatório técnico, Network Working Group, june 1999.

[19] JIA, Q.-S., GUO, Y. "Discovering the knowledge hierarchy of MathWorld for web intelligence". In: *Proceedings of the Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 535–539, 2009.

[20] W3C. "W3C Math Home". `http://www.w3.org/Math/`, 2010.

[21] ALBERT, R., JEONG, H., BARABÁSI, A.-L. "Diameter of the world-wide web", *Nature*, v. 401, pp. 130–131, 1999.

[22] WATTS, D. J., STROGATZ, S. H. "Collective dynamics of 'small-world' networks", *Nature*, v. 393, pp. 440–442, 1998.

[23] NEWMAN, M. E. J. *Networks*. Oxford, UK, Oxford University Press, 2010.

[24] NEWMAN, M. E. J. “Assortative mixing in networks”, *Phys. Rev. Lett.*, v. 89, pp. 208701, 2002.

[25] NEWMAN, M. E. J. “Mixing patterns in networks”, *Phys. Rev. E*, v. 67, pp. 026126, 2003.

[26] FOSTER, J. G., FOSTER, D. V., GRASSBERGER, P., et˜al. “Edge direction and the structure of networks”, *Proc. Natl. Acad. Sci. USA*, v. 107, pp. 10815–10820, 2010.

[27] PIRAVEENAN, M., PROKOPENKO, M., ZOMAYA, A. “Assortative mixing in directed biological networks”, *IEEE/ACM T. Comput. Biol. Bioinform.*, v. 9, pp. 66–78, 2012.

[28] NEWMAN, M. E. J., STROGATZ, S. H., WATTS, D. J. “Random graphs with arbitrary degree distributions and their applications”, *Phys. Rev. E*, v. 64, pp. 026118, 2001.

[29] BRODER, A., KUMAR, R., MAGHOUL, F., et˜al. “Graph structure in the Web”, *Comput. Netw.*, v. 33, pp. 309–320, 2000.

[30] DONATO, D., LAURA, L., LEONARDI, S., et˜al. “Large scale properties of the Webgraph”, *Eur. Phys. J. B*, v. 38, pp. 239–243, 2004.

[31] CAPOCCI, A., SERVEDIO, V. D. P., COLAIORI, F., et˜al. “Preferential attachment in the growth of social networks: the internet encyclopedia Wikipedia”, *Phys. Rev. E*, v. 74, pp. 036116, 2006.

[32] ZLATIĆ, V., BOŽIČEVIĆ, M., ŠTEFANČIĆ, H., et˜al. “Wikipedias: collaborative web-based encyclopedias as complex networks”, *Phys. Rev. E*, v. 74, pp. 016115, 2006.

[33] NEWMAN, M. E. J. “Power laws, Pareto distributions and Zipf’s law”, *Contemp. Phys.*, v. 46, pp. 323–351, 2005.

[34] PRICE, D. “A general theory of bibliometric and other cumulative advantage processes”, *J. Amer. Soc. Inform. Sci.*, v. 27, pp. 292–306, 1976.

[35] BARABÁSI, A.-L., ALBERT, R. “Emergence of scaling in random networks”, *Science*, v. 286, pp. 509–512, 1999.

[36] BARABÁSI, A.-L., ALBERT, R., JEONG, H. “Scale-free characteristics of random networks: the topology of the world-wide web”, *Physica A*, v. 281, pp. 69–77, 2000.

[37] BRANDES, U. "A Faster Algorithm for Betweenness Centrality", *Journal of Mathematical Sociology*, v. 25, pp. 163–177, 2001.

[38] ANTHONISSE, J. M. "The rush in a directed graph", *Technical Report BN 9/71*, 1971.

[39] FREEMAN, L. C. "A set of measures of centrality based on betweenness", *Sociometry*, v. 40, pp. 35–42, 1977.

[40] SHIMBEL, A. "Structural parameters of communication networks", *Bull. Math. Biophys.*, v. 15, pp. 501–507, 1953.

[41] SABIDUSSI, G. "The centrality index of a graph", *Psychometrika*, v. 31, pp. 581–603, 1966.

[42] HAGE, P., HARARY, F. "Eccentricity and centrality in networks", *Soc. Netw.*, v. 17, pp. 57–63, 1995.

[43] KLEINBERG, J. M. "Authoritative sources in a hyperlinked environment", *J. ACM*, v. 46, pp. 604–632, 1999.

[44] ERDŐS, P., RÉNYI, A. "On random graphs", *Publ. Math. (Debrecen)*, v. 6, pp. 290–297, 1959.

[45] ERDŐS, P., RÉNYI, A. "On the evolution of random graphs", *Publ. Math. Inst. Hung. Acad. Sci. A*, v. 5, pp. 17–61, 1960.

[46] KARP, R. M. "The transitive closure of a random digraph", *Random Struct. Algor.*, v. 1, pp. 73–93, 1990.

[47] MOLLOY, M., REED, B. "A critical point for random graphs with a given degree sequence", *Random Struct. Algor.*, v. 6, pp. 161–180, 1995.

[48] MOLLOY, M., REED, B. "The size of the largest component of a random graph on a fixed degree sequence", *Comb. Probab. Comput.*, v. 7, pp. 295–306, 1998.

[49] DOROGOVTSEV, S. N., MENDES, J. F. F., SAMUKHIN, A. N. "Giant strongly connected component of directed networks", *Phys. Rev. E*, v. 64, pp. 025101, 2001.

[50] BOLLOBÁS, B. *Random Graphs*. Second ed. Cambridge, UK, Cambridge University Press, 2001.

[51] COHEN, R., EREZ, K., BEN AVRAHAM, D., et˜al. "Resilience of the Internet to random breakdowns", *Phys. Rev. Lett.*, v. 85, pp. 4626–4628, 2000.

[52] ALBERT, R., JEONG, H., BARABÁSI, A.-L. "Error and attack tolerance of complex networks", *Nature*, v. 406, pp. 378–382, 2000.

[53] LEVENSHTEIN, V. I. "Binary codes with correction for deletions and insertions of the symbol 1", *Problemy Peredachi Informatsii*, v. 1, n. 1, pp. 12–25, 1965.

[54] ALI, R., BEG, M. M. "An overview of Web search evaluation methods", *Computers & Electrical Engineering*, 2011.

[55] GORDON, M., PATHAK, P. "Finding information on the World Wide Web: the retrieval effectiveness of search engines", *Information Processing & Management*, v. 35, n. 2, pp. 141–180, 1999.

[56] SU, L., CHEN, H., DONG, X. "Evaluation of Web-Based Search Engines from the End-User's Perspective: A Pilot Study." In: *Proceedings of the ASIS Annual Meeting*, v. 35, pp. 348–61. ERIC, 1998.

[57] SU, L. T. "Value of search results as a whole as the best single measure of information retrieval performance", *Information Processing & Management*, v. 34, n. 5, pp. 557–579, 1998.

[58] SU, L. T. "A comprehensive and systematic model of user evaluation of Web search engines: I. Theory and background", *Journal of the American society for information science and technology*, v. 54, n. 13, pp. 1175–1192, 2003.

[59] SALEM, L., TESTARD, F., SALEM, C. *The Most Beautiful Mathematical Formulas*. Wiley, 1997.

[60] STEWART, I. *In Pursuit of the Unknown: 17 Equations That Changed the World*. Basic Books, 2012.

# Apêndice A

# Importação da *Wikipedia*

Conforme mencionado no texto, para extrair apenas o conteúdo Matemático da base de dados da *Wikipedia*, foi necessário o estudo das tabelas, de forma a poder se extrair as páginas, relações de links, categorias e relações de categorias com as páginas e as próprias categorias. Segue então uma descrição mais detalhadas das tabelas necessárias e utilizadas por este trabalho:

- Page:
  - page_id: Atributo inteiro, chave primária da tabela;
  - page_namespace: O nome da página é dividido em namespace e title. Este atributo contém um código do namespace da página. Esse código é o que diferencia duas páginas de mesmo nome, mas hierarquia diferente. Exemplo, pode-se ter o Portal Mathematics, e a página de conteúdo Mathematics. Nesse caso, ambos terão o mesmo atributo title (Mathematics), mas diferentes valores de namespace. Os mais comuns são: 0 para páginas de conteúdo "real" e artigos, 14 para páginas de categorias, 2 para páginas de usuários e 12 para páginas de ajuda;
  - page_title: É o título da página sem o seu namespace.
- Revision:
  - rev_id: Atributo inteiro, chave primaria da tabela;
  - rev_page: Atributo inteiro, chave estrangeira para tabela Page;
  - rev_text_id: Atributo inteiro, chave estrangeira para tabela Text.
- Text (contém o texto das páginas):
  - old_id: Atributo inteiro, chave primaria da tabela;
  - old_text: Fica armazenado o código HTML da página.

- PageLinks (contém as referências de links das páginas):
  - pl_from: Atributo chave estrangeira para page_id em Page. É o id da página origem que possui um link para outras páginas;
  - pl_namespace: Atributo contendo o namespace da página destino, página que está sendo referenciada pelo link;
  - pl_title: Atributo contendo o page_title em Page. É o título da página destino, página que esta sendo referenciada pelo link.
- Category:
  - cat_id: Atributo inteiro, chave primaria;
  - cat_title: Nome da categoria;
  - cat_pages: Número de páginas na categoria;
  - cat_subcats: Número de sub-categorias na categoria.
- CategoryLinks:
  - cl_from: Armazena o page_id da página onde possui o link para a categoria;
  - cl_to: Armazena o nome da categoria indicada.

## A.1 Estrutura proposta

Para a importação somente do conteúdo matemático o Banco de Dados foi modelado de acordo com o diagrama da Figura A.1. A descrição das tabelas é a seguinte:

- tb_html (armazena as páginas):
  - htm_id: Atributo inteiro, chave primária;
  - htm_idold: Id original da página na base da Wikipedia;
  - htm_namespace: O nome da página é dividido em namespace e title. Este atributo contém um código do namespace da página;
  - htm_title: É o título da página sem o seu namespace;
  - htm_title_SW: Título da página sem *stop-words*;
  - htm_title_stem: Título da página sem *stop-words* e reduzidos usando o algoritmo de *Porter Stemming*[1];
  - htm_fonte: Texto da página;

[1] `http://tartarus.org/martin/PorterStemmer/`

- tb_link (links entre as páginas):
  - lin_id: Atributo inteiro, chave primária;
  - lin_idPaginaOrigem: Contém o código da página que onde o link está;
  - lin_idPaginaDestino: Contém o código da página para a qual o link. aponta.
- tb_equation (equações extraídas):
  - equ_id: Atributo inteiro, chave primária;
  - fk_html_equation: chave estrangeira para tb_html;
  - equ_equation: Equação extraída da página;
  - equ_equation_lex: Tokens da equação após análise léxica.
- tb_category (categorias e subcategorias da Wikipedia referentes à Matemática)
  - cat_id: Atributo inteiro, chave primaria;
  - cat_idold: contém o id antigo da categoria no banco de dados original da Wikipedia;
  - cat_title: Nome da categoria;
  - cat_kind: Pode ser 1, 2 ou 3, segundo o seguinte critério: categoria matemática = 1; categoria de matemáticos = 2; categoria relacionada a matemática = 3; cat_level: Nível da categoria considerando a árvore de categorias.
- rl_html_category (relações entre páginas e categorias):
  - htm_id: Código da página;
  - cat_id: Código da categoria.
- rl_category_category (relações entre as categorias):
  - cat_idOrigem: Categoria que referencia outra;
  - cat_idDestino: Categoria referenciada.

## A.2 Pseudocódigos

Os pseudocódigos aqui apresentados manipulam as tabelas originais da *Wikipedia*, importadas do *dump*, e as tabelas criadas na estrutura proposta em A.1. Observe que as tabelas criadas começam com "tb", enquanto que as originais da Wikipedia não. Através dessa diferença na nomenclatura então é possível saber nos códigos quais as tabelas são manipuladas.

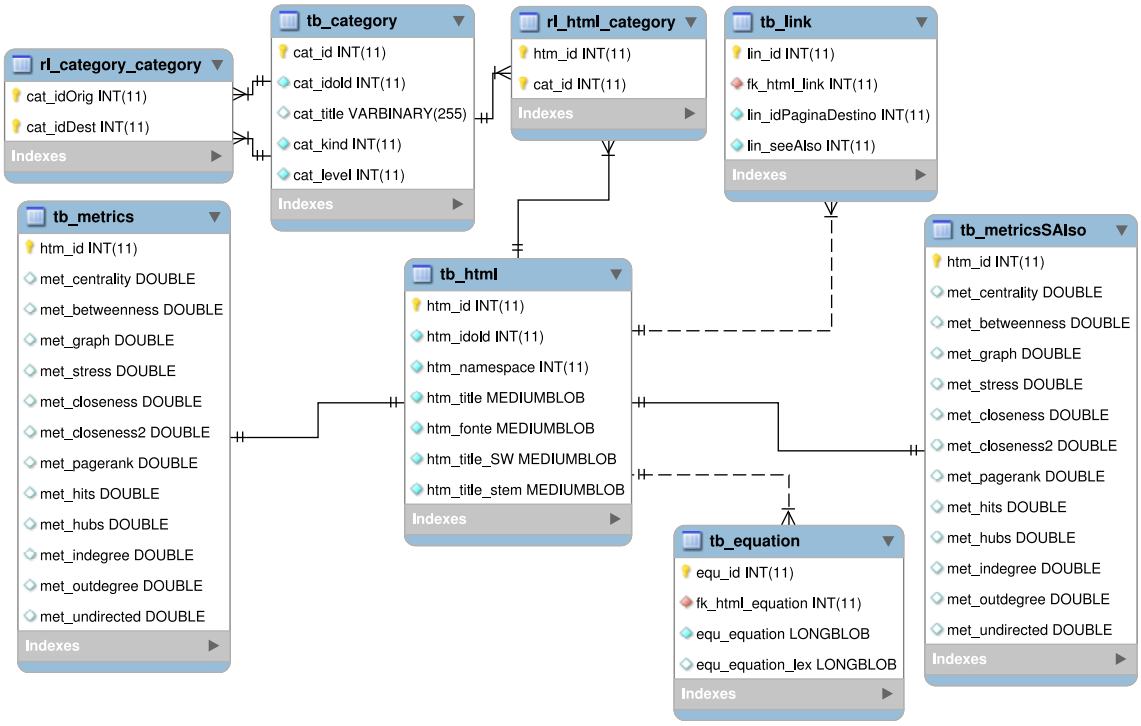



Figura A.1: Estrutura do Banco de Dados.

### A.2.1 Marcação das categorias matemáticas

O pseudocódigo D.1 lê os nomes das categorias da matemática contidas na página[2] e realiza a marcação no banco das categorias que compõem a matemática. Os nomes das categorias estão contidos no texto da página em uma expressão que começa com ':Category:' e termina com '|' ou ']]'. A marcação é feita através da seguinte distribuição no campo temporário denominado math criado na tabela Category:

1. categoria matemática;
2. categoria de matemáticos;
3. categoria relacionada a matemática.

Esse atributo será usado depois pelos outros algoritmos descritos.

Código A.1: Marcação das categorias matemáticas.

```
1 Q <- Selecionar page_id de Page onde
2      page_title = 'List_of_mathematics_categories';
3 P <- Selecionar rev_text_id de Revision onde rev_page = Q.page_id;
4 E <- Selecionar old_text de Text onde old_id = P.rev_text_id;
5 D <- Pesquisar por expressão regular começando com ':Category:'
6      e terminando em '|' ou ']]' em E.old_text;
7 para cada resultado em D faça
```

[2] http://en.wikipedia.org/wiki/List_of_mathematics_categories

```
8           category = D.expressao;
9           V <- verificar se category é do tipo 1, 2 ou 3
10              de acordo com posição na página
11          se V == 1
12                  Atualizar category setar math = 1 onde cat_title =
                        category;
13          se V == 2
14                  Atualizar category setar math = 2 onde cat_title =
                        category;
15          se V == 3
16                  Atualizar category setar math = 3 onde cat_title =
                        category;
```

### A.2.2 Geração da tb_html

O pseudocódigo A.2 recupera as páginas da área matemática do banco de dados da Wikipedia e armazena na tb_html da estrutura proposta. Para isso, é necessário fazer uma junção das tabelas *Page*, *Revision* e *Text* para que se possa recuperar o page_id, page_namespace, page_title e old_text. Porém antes disso é necessário marcar na tabela *Page* quais são as páginas da Matemática. Para essa função, foi criado um atributo, denominado math, e se a página for da matemática é atribuído o valor 1 a ele, se não fica com o valor 0. O seguinte algoritmo é para fazer a marcação nas páginas relacionadas a matemática. Este algoritmo utiliza de uma mesma marcação na tabela *Category* onde foi criado um atributo, denominado math, para se marcar as categorias que são da matemática. Observe que após marcar as páginas de conteúdo, na última linha a página que contém o texto da categoria também é marcada com valor 1.

Código A.2: Marcação das páginas da matemática.

```
1 Q <- Selecionar cat_title de Category onde math >= 1 e math <= 3;
2 para cada resultado em Q faça
3           cat_title = Q.cat_title;
4           P <- Selecionar cl_from de CategoryLinks onde cl_to = cat_title
5           para cada resultado em P faça
6                   page_id = P.cl_from
7                   Atualizar Page P setar P.math = 1 onde P.page_id =
                        page_id
8           Atualizar Page P setar P.math = 1 onde
9               page_title = cat_title e page_namespace = 14;
```

O pseudocódigo para a geração da tb_html é apresentado no código A.3.

Código A.3: Obtenção da tb_html.

```
1 Q <- Selecionar page_id, page_namespace, page_title e old_text
```

```
2       de Page, Text e Revision onde page_id = rev_page
3       e rev_text_id = old_id e math = 1
4   para cada resultado em Q faça
5           page_id = Q.page_id;
6           page_namespace = Q.page_namespace;
7           page_title = Q.page_title;
8           old_text = Q.old_text;
9           Inserir em tb_html valores
10                  page_id, page_namespace, page_title e page_title
```

### A.2.3 Geração da tb_link

O pseudocódigo A.4 recupera todas as páginas matemáticas que possuem links para outras páginas matemáticas e armazena na tb_link. Para isso, foi criado um atributo temporário, denominado lin_math, que recebe 1 quando se atualiza o link de destino, dessa forma, os links que forem de páginas matemáticas para páginas matemáticas terão math = 1. Ao final da execução do algoritmo basta excluir as tuplas com math diferente de 1 para retirar links com referencias para páginas não matemáticas. Observação: Esse código considera ainda na tb_link a existência dos campos lin_namespaceDestino e lin_titleDestino. Ao final esses campos podem ser apagados.

Código A.4: Obtenção da tb_link.

```
1   Q <- Selecionar PL.pl_from, PL.pl_namespace e PL.pl_title de
2       PageLinks PL e Page P onde PL.pl_from = P.page_id e P.math = 1;
3   para cada resultado em Q faça
4           inserir em tb_link (fk_html_link, lin_idPaginaDestino,
5               lin_namespaceDestino, lin_titleDestino) valores
6               (Q.pl_from, 0, Q.pl_namespace, Q.pl_title);
7   Q <- Selecionar htm_id, htm_namespace, htm_title de tb_html
8   para cada resultado em Q faça
9           htm_namespace = Q.htm_namespace;
10          htm_id = Q.htm_id;
11          htm_title = Q.htm_title;
12          atualizar tb_link setando lin_math = 1 e
13              lin_idPagindaDestino = htm_id onde
14              lin_namespaceDestino = htm_namespace e
15              lin_titleDestino = htm_title
16  deletar em tb_link onde lin_math = 0
17  deletar em tb_link campos lin_namespaceDestino e lin_titleDestino
```

### A.2.4 Geração da tb_equation

O pseudocódigo A.5 recupera as equações presentes nas páginas de matemática da Wikipedia. As equações estão entre as tags <math> e </math> no texto das

páginas.

Código A.5: Obtenção da tb_equation.

```
1
2  Q <- Selecionar htm_id, htm_fonte de tb_html
3  para cada resultado de Q faça
4          htm_id = Q.htm_id;
5          htm_fonte = Q.htm_fonte;
6          E <- buscar por expressões regulares entre
7              <math> e </math> em htm_fonte
8          para cada resultado em E faça
9                  equacao = E.proximaEquacao();
10                 se equacao não existir em tb_equation
11                         Inserir em tb_equation valores (htm_id, equacao
                               );
```

### A.2.5 Geração da tb_category

Para se gerar esta tabela basta ler as categorias em Category onde o campo math for diferente de 0. O pseudocódigo é apresentado em A.6.

Código A.6: Obtenção da tb_category.

```
1  Q <- Selecionar cat_id, cat_title e math de Category onde math > 0;
2  para cada resultado em Q faça
3          cat_id = Q.cat_id;
4          cat_title = Q.cat_tile;
5          math = Q.math;
6          Inserir em tb_category valores (cat_id, cat_title, math);
```

### A.2.6 Geração da rl_html_category

O pseudocódigo A.7 escreve nesta tabela a relação entre as páginas e as categorias da matemática. Em CategoryLinks já se encontra a relação de todas as páginas da Wikipedia com as suas categorias. Basta pesquisar nessa tabela pelas páginas da matemática e ver quais são as categorias da matemática relacionadas.

Código A.7: Obtenção da rl_html_category.

```
1  Q <- Selecionar htm_id de tb_html
2  para cada resultado em Q faça
3          htm_id = Q.htm_id;
4          P <- Selecionar * de CategoryLinks onde cl_from = htm_id e math
               = 1
5          para cada resultado em P faça
6                  cl_to = P.cl_to;
```

```
7                   E <- Selecionar cat_id de tb_category onde cat_title =
                        cl_to;
8                   para cada resultado em E faça
9                           cat_id = E.cat_id;
10                          Inserir em rl_html_category valores (htm_id,
                                cat_id);
```

### A.2.7 Geração da rl_category_category

O pseudocódigo A.8 que grava nesta tabela a relação entre categorias, estabelecendo o nível de cada uma. A categoria Mathematics recebe o nível 0. Categorias que se relacionam com ela recebem 1, e assim por diante. Cada página de categoria possui em seu código HTML o nome das categorias que as indicam, e os nomes estão entre a expressão regular começando com '[[Category:' e terminando com ']' ou '|'.

Código A.8: Obtenção da rl_category_category.

```
1  Q <- Selecionar htm_fonte, htm_title de tb_html onde htm_namespace =
       14;
2  para cada resultado em Q faça
3          htm_fonte = Q.htm_fonte;
4          htm_title = Q.htm_title;
5          P <- Pesquisar por expressão regular que começa com
6              '[[Category:' e termina com ']'ou '|';
7          para cada resultado em P faça
8                  cat_idOrigem = Selecionar cat_id de tb_category
9                      onde cat_title = P.expressao;
10                 cat_idDestino =  Selecioanr cat_id de tb_category
11                     onde cat_title = htm_title;
12                 Inserir em rl_category_category
13                     valores (cat_idOrigem, cat_idDestino));
14 Atualizar tb_html h, tb_category c, rl_html_category r setar
15     c.cat_level = 0 onde h.htm_namespace = 100 e
16     h.htm_title = 'Mathematics' e h.htm_id = r.htm_id e
17     r.cat_id = c.cat_id e c.cat_kind = 1;
18 nivel = 0;
19 faça
20         C <- Selecionar cat_idDestino de rl_category_category
21             onde cat_idOrigem contém (Selecionar cat_id de tb_category
22             onde cat_level = nivel)
23         Se (C == vazio)
24                 pare a execução;
25         senão
26                 para cada resultado em C faça
27                         cat_idDest = C.cat_idDestino;
28                         Atualizar tb_category setar
```

```
29                                  cat_level = nivel + 1
30                                  onde cat_id = cat_idDest;
31          nivel = nivel + 1;
```

# Apêndice B

# Fórmulas Usadas no Teste

The definition of Pi

$$\pi = \frac{c}{d} \tag{B.1}$$

Euler's formula for polyhedra

$$F - E + V = 2 \tag{B.2}$$

The binomial theorem

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k} \tag{B.3}$$

The closed-form expression for the Fibonacci number

$$F(k) = \frac{\varphi^k - (-\frac{1}{\varphi})^k}{\sqrt{5}} \tag{B.4}$$

Euler's product formula

$$\sum_{k=1}^{\infty} \frac{1}{k^s} = \prod_{p \text{ prime}} \frac{1}{1 - p^{-s}} \tag{B.5}$$

The fundamental theorem of calculus

$$f(x) = \frac{d}{dx} \int_a^x f(s)ds \tag{B.6}$$

The integral definition of the exponential function

$$\int_1^{e^x} \frac{dy}{y} = x \tag{B.7}$$

The limit definition of the exponential function

$$e^x = \lim_{y \to \infty} \left(1 + \frac{x}{y}\right)^y \tag{B.8}$$

Euler's identity

$$e^{i\pi} + 1 = 0 \tag{B.9}$$

The Gamma function

$$\Gamma(n) = \int_0^\infty e^{-x} x^{n-1} dx \tag{B.10}$$

The Gaussian integral

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi} \tag{B.11}$$

The Laplace transform

$$F(s) = \int_{-\infty}^{\infty} f(x) e^{-sx} dx \tag{B.12}$$

The definition of conditional probability

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \tag{B.13}$$

The definition of total probability

$$P(B) = \sum_k P(A_k \cap B) \tag{B.14}$$

Bayes' theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{B.15}$$

The normal distribution

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{B.16}$$

Shannon's entropy

$$H(X) = -\sum_{x \in \mathbf{X}} p(x) \log p(x) \tag{B.17}$$

The Taylor series

$$f(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(a)}{k!} (x-a)^k \tag{B.18}$$

The series expansion of the exponential function

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!} \tag{B.19}$$

The series expansion of the natural logarithm

$$\ln(1+x) = \sum_{k=1}^{\infty} \frac{(-1)^{k+1} x^k}{k} \tag{B.20}$$

Newton's universal law of gravitation

$$F = G\frac{m_1 m_2}{d^2} \tag{B.21}$$

The wave equation

$$\frac{\partial^2 u}{\partial t^2} = c^2 \nabla^2 u \tag{B.22}$$

The heat equation

$$\frac{\partial \varnothing}{\partial t} = D\nabla^2 \varnothing \tag{B.23}$$

Maxwell's equations for electromagnetic waves

$$\nabla \cdot E = \frac{\rho}{\varepsilon} \tag{B.24}$$

$$\nabla \times E = -\frac{\partial B}{\partial t} \tag{B.25}$$

$$\nabla \cdot B = 0 \tag{B.26}$$

$$\nabla \times B = \mu J + \mu\varepsilon\frac{\partial E}{\partial t} \tag{B.27}$$

A particle's rest energy in special relativity

$$E = mc^2 \tag{B.28}$$

Einstein's field equations in general relativity

$$\mathcal{G} = -\frac{8\pi G}{c^4}\mathcal{T} \tag{B.29}$$

The Schrödinger equation

$$i\hbar\frac{\partial \Psi}{\partial t} = \hat{H}\Psi \tag{B.30}$$

# Apêndice C

# Instruções Passadas aos Usuários

Instruções para a realização do teste

Oi, seja bem vindo! Gostaria primeiramente de agradecê-lo(a) pela disponibilidade em contribuir com o desenvolvimento da nossa ferramenta de busca. :-)

O objetivo do teste é avaliar o desempenho de duas diferentes ferramentas de busca por fórmulas matemáticas.

A realização do teste é bastante simples. São fornecidas cinco fórmulas, cada uma com cinco resultados.

Cada resultado é composto dos seguintes itens: título da página; expressão encontrada; URL.

Por exemplo, suponha que o resultado da busca por $E = mc^2$ retorne como primeiro resultado a página da Wikipedia "Mass-energy_equivalence", e que a página contenha de fato a equação buscada. Assim os campos apresentarão:

Resposta 1:

- título da página: Mass-energy_equivalence;
- expressão encontrada: $E = mc^2$;
- URL: `http://bit.ly/VG90sQ`;

Observações:

- Observe que o link no campo URL é "clicável" no pdf.
- Caso o título da página seja grande demais, ele foi truncado, preservando a parte mais importante do mesmo.
- Os resultados não são ordenados. Ou seja, o primeiro resultado pode não ser o melhor, e de forma semelhante, o último pode não ser o pior.

Para cada resposta retornada pela busca, a seguinte afirmação deve ser considerada:

"A página/documento indicada(o) é útil para compreender a fórmula procurada, ou apresenta fórmula semelhante, onde as mudanças ocorrem por manipulações algébricas simples ou troca de notação".

O usuário deve então marcar na folha apenas UMA dentre as alternativas:
a) Concordo $--> $ (caso a afirmação seja verdadeira para a resposta analisada);
b) Indiferente $--> $ (não é possível se posicionar à favor ou contra com relação à afirmação);
c) Discordo $--> $ (caso a afirmação seja falsa para a resposta analisada).

Muito Obrigado!
Prof. Flavio Barbieri Gonzaga

# Apêndice D

# Descrição dos Tokens

Código D.1: Gramática desenvolvida.

```
1  START    "\\"?[Bb]"egin"|"\\"?[Ee]"nd"
2
3  TEX      "{"|"}"|"\\"?[Ll]"imits"|"\\"?[Nn]"olimits"|
4           "\\"?[Dd]"isplaylimits"
5
6  ENPH     "\\"?[Ll]"eft"|"\\"?[Rr]"ight"|"\\"?[Bb]"ig""g"*
7
8  SUM      "+"|"-"|"\\"?[Pp]"m"|"\\"?[Mm]"p"|"\\"?[Oo]"plus"|
9           "\\"?[Oo]"minus"|"\\"?[Bb]"igoplus"|[Mm]"inus"|
10          [Aa]"ddition"|"\\"?[Pp]"lus"|"\\"?[Bb]"oxplus"|
11          "\\"?[Bb]"oxminus"|"+/-"|"&#8723"";"?|"&#177"";"?|
12          "direct sum"|"&#8853"";"?
13
14 FSUM     "\\"?[Ss]"um"
15
16 MULT     "\\"?(r|l)?"times"|"\\"?[Dd]"iv"|
17          "\\"?[Dd]"ivideontimes"|"/"|"\\"?[Ff]"rac"|
18          "\\"?[Tt]"frac"|"\\"?[Dd]"frac"|"\\"?[Oo]"times"|
19          [Mm]"ultiplication"|"\\"?[Bb]"oxtimes"|
20          [Mm]"ultiply"|[Mm]"ult""t"?|"\\"?[Oo]"slash"|
21          "\\"?[Ll]"eftthreetimes"|"\\"?[Rr]"ightthreetimes"|
22          "/"|"&divide"";"?|"&#8855"";"?|"&#215"";"|"tensor"|
23          "\\"?[Oo]"ver"
24
25 FMULT    "\\"?[Pp]"rod"|"\\"?[Cc]"oprod"
26
27 GREEK    ("\\"|"&")?[Aa]"lpha"";"?|("\\"|"&")?[Bb]"eta"";"?|
28          ("\\"|"&")?[Dd]"elta"";"?|("\\"|"&")?[Ee]"psilon"";"?|
29          ("\\"|"&")?[Zz]"eta"";"?|("\\"|"&")?[Ee]"ta"";"?|
30          ("\\"|"&")?[Tt]"heta"";"?|("\\"|"&")?[Ii]"ota"";"?|
31          ("\\"|"&")?[Kk]"appa"";"?|("\\"|"&")?[Ll]"ambda"";"?|
32          ("\\"|"&")?[Mm]"u"";"?|("\\"|"&")?[Nn]"u"";"?|
```

```
33              ("\\"|"&")?[Xx]"i"";"?|("\\"|"&")?[Rr]"ho"";"?|
34              ("\\"|"&")?[Ss]"igma"";"?|("\\"|"&")?[Tt]"au"";"?|
35              ("\\"|"&")?[Uu]"psilon"";"?|("\\"|"&")?[Pp]"hi"";"?|
36              ("\\"|"&")?[Cc]"hi"";"?|("\\"|"&")?[Pp]"si"";"?|
37              ("\\"|"&")?[Oo]"mega"";"?|("\\"|"&")?[Vv]"arepsilon"";"?|
38              ("\\"|"&")?[Dd]"igamma"";"?|("\\"|"&")?[Vv]"arkappa"";"?|
39              ("\\"|"&")?[Vv]"arpi"";"?|("\\"|"&")?[Vv]"arrho"";"?|
40              ("\\"|"&")?[Vv]"arsigma"";"?|("\\"|"&")?[Vv]"artheta"";"?|
41              ("\\"|"&")?[Vv]"arphi"";"?|("\\"|"&")?[Oo]"micron"";"?
42
43  GAMA        ("\\"|"&")?[Gg]"amma"";"?
44
45  HEBREW      "\\"?[Aa]"leph"|"\\"?[Bb]"eth"|"\\"?[Gg]"imel"|
46              "\\"?[Dd]"aleth"
47
48  POLAC       "&#260";"?|"&#262";"?|"&#280";"?|"&#321";"?|
49              "&#323";"?|"&#346";"?|"&#377";"?|"&#379";"?|
50              "&#261";"?|"&#263";"?|"&#281";"?|"&#322";"?|
51              "&#324";"?|"&#347";"?|"&#378";"?|"&#380";"?
52
53  DOTS        "\\"?[Cc]"dots"|"\\"?[Dd]"ot""s"?|"\\"?[Ll]"dots"|
54              "\\"?[Vv]"dots"|"\\"?[Dd]"dots"("c"|"m")?|"."+|
55              "\\"?[Bb]"oxdot"|"\\"?[Bb]"ullet"|"\\"?[Cc]"enterdot"|
56              "\\"?[Dd]"oteqdot"|"\\"?[Rr]"isingdotseq"|
57              "\\"?[Ff]"allingdotseq"|"&#183";"?
58
59  LOG         "\\"?[Ll]"og"|"\\"?[Ll]"n"|"\\"?[Ll]"g"|[Ll]"ogarithmic"|
60              [Nn]"aplog"|[Ee]"xp"
61
62  LIN         "'"+|("\\"?[Pp]"rime")+|"`"+
63
64  BOT         "_"|"\\"?[Uu]"nderbrace"
65
66  FACT        "!"
67
68  DIRAC       "\\"?[Hh]"bar"
69
70  RELAT       "="|"\\"?[Nn]"e"|"\\"?[Nn]"eq"|"\\"?[Ee]"quiv"|
71              "\\"?[Dd]"oteq"|"\\"?[Dd]"oteqdot"|":="|"\\"?[Ss]"im"|
72              "\\"?[Nn]"sim"|"\\"?[Bb]"acksim"|"\\"?[Tt]"hicksim"|
73              "\\"?[Ss]"imeq"|"\\"?[Bb]"acksimeq"|"\\"?[Ee]"qsim"|
74              "\\"?[Cc]"ong"|"\\"?[Nn]"cong"|"\\"?[Aa]"pprox"|
75              "\\"?[Tt]"hickapprox"|"\\"?[Aa]"pproxeq"|"\\"?[Aa]"symp"|
76              "\\"?[Pp]"ropto"|"\\"?[Vv]"arpropto"|"<"|"\\"?[Nn]"less"|
77              "\\"?[Ll]"l"|"\\"?[Ll]"ll"|"\\"?[Ll]"essdot"|">"|
78              "\\"?[Nn]"gtr"|"\\"?[Gg]"g"|"\\"?[Gg]"gg"|"\\"?[Gg]"trdot"|
79              "\\"?[Ll]"e"|"\\"?[Ll]"eq"|"\\"?[Ll]"neq"|"\\"?[Ll]"eqq"|
```

```
80          "\\"?[Nn]"leqq"|"\\"?[Ll]"neqq"|"\\"?[Ll]"vertneqq"|
81          "\\"?[Gg]"e"|"\\"?[Gg]"eq"|"\\"?[Gg]"neq"|"\\"?[Gg]"eqq"|
82          "\\"?[Nn]"geqq"|"\\"?[Gg]"neqq"|"\\"?[Gg]"vertneqq"|
83          "\\"?[Ll]"essgtr"|"\\"?[Ll]"esseqgtr"|"\\"?[Ll]"esseqqgtr"|
84          "\\"?[Gg]"trless"|"\\"?[Gg]"treqless"|"\\"?[Gg]"treqqless"|
85          "\\"?[Ll]"eqslant"|"\\"?[Nn]"leqslant"|
86          "\\"?[Ee]"qslantless"|"\\"?[Gg]"eqslant"|
87          "\\"?[Nn]"geqslant"|"\\"?[Ee]"qslantgtr"|"\\"?[Ll]"esssim"|
88          "\\"?[Ll]"nsim"|"\\"?[Ll]"essapprox"|"\\"?[Ll]"napprox"|
89          "\\"?[Gg]"trsim"|"\\"?[Gg]"nsim"|"\\"?[Gg]"trapprox"|
90          "\\"?[Gg]"napprox"|"\\"?[Pp]"rec"|"\\"?[Nn]"prec"|
91          "\\"?[Pp]"receq"|"\\"?[Nn]"preceq"|"\\"?[Pp]"recneqq"|
92          "\\"?[Ss]"ucc"|"\\"?[Nn]"succ"|"\\"?[Ss]"ucceq"|
93          "\\"?[Nn]"succeq"|"\\"?[Ss]"uccneqq"|"\\"?[Pp]"reccurlyeq"|
94          "\\"?[Cc]"urlyeqprec"|"\\"?[Ss]"ucccurlyeq"|
95          "\\"?[Cc]"urlyeqsucc"|"\\"?[Pp]"recsim"|"\\"?[Pp]"recnsim"|
96          "\\"?[Pp]"recapprox"|"\\"?[Pp]"recnapprox"|"\\"?[Ss]"uccsim"|
97          "\\"?[Ss]"uccnsim"|"\\"?[Ss]"uccapprox"|"\\"?[Ss]"uccnapprox"|
98          "~"|[Ee]"qual"|"\\"?[Tt]"riangleq"|"!="|"<"|">"|"<="|">="|
99          "&gt"";"?"="?|"&lt"";"?"="?|"&there4"";"?|"&#8757;"|"&#8756;"|
100         ":"|"&sim"";"?|"&#8801"";"?|"&#8802"";"?|"&#8803"";"?|
101         "&#8826"";"?|"&#8827"";"?|"&#8910"";"?|"&#8911"";"?|
102         "&#62417"";"?|"&#8882"";"?|"&#8883"";"?|"&#8884"";"?|
103         "&#8885"";"?|"&amp"";"?|"&#65079"";"?|"same"|"&#8790"";"?|
104         "&#8791"";"?|"&#8792"";"?|"&#8793"";"?|"&#8794"";"?|
105         "&#8795"";"?|"&#8796"";"?|"&#8797"";"?|"&#8798"";"?|
106         "&#8799"";"?|"&#8800"";"?
107
108 ARROW   "\\"?[Rr]"rightarrow"|"\\"?[Ll]"leftarrow"|
109         "\\"?[Rr]"ightarrow"|"\\"?[Nn]"Rightarrow"|
110         "\\"?[Ll]"ongrightarrow"|"\\"?[Ii]"mplies"|
111         "\\"?[Ll]"eftarrow"|"\\"?[Nn]"Leftarrow"|
112         "\\"?[Ll]"ongleftarrow"|"\\"?[Ll]"eftrightarrow"|
113         "\\"?[Nn]"Leftrightarrow"|"\\"?[Ll]"ongleftrightarrow"|
114         "\\"?[Uu]"parrow"|"\\"?[Dd]"ownarrow"|"\\"?[Uu]"pdownarrow"|
115         "\\"?[Nn]"earrow"|"\\"?[Ss]"warrow"|"\\"?[Nn]"warrow"|
116         "\\"?[Ss]"earrow"|"\\"?[Mm]"apsto"|"\\"?[Ll]"ongmapsto"|
117         "\\"?[Rr]"ightharpoonup"|"\\"?[Rr]"ightharpoondown"|
118         "\\"?[Ll]"eftharpoonup"|"\\"?[Ll]"eftharpoondown"|
119         "\\"?[Uu]"pharpoonleft"|"\\"?[Uu]"pharpoonright"|
120         "\\"?[Dd]"ownharpoonleft"|"\\"?[Dd]"ownharpoonright"|
121         "\\"?[Rr]"ightleftharpoons"|"\\"?[Ll]"eftrightharpoons"|
122         "\\"?[Cc]"urvearrowleft"|"\\"?[Cc]"irclearrowleft"|
123         "\\"?[Ll]"sh"|"\\"?[Uu]"puparrows"|"\\"?[Rr]"ightrightarrows"|
124         "\\"?[Rr]"ightleftarrows"|"\\"?[Rr]"ightarrowtail"|
125         "\\"?[Ll]"ooparrowright"|"\\"?[Cc]"urvearrowright"|
126         "\\"?[Cc]"irclearrowright"|"\\"?[Rr]"sh"|
```

```
127             "\\"?[Dd]"owndownarrows"|"\\"?[Ll]"eftleftarrows"|
128             "\\"?[Ll]"eftrightarrows"|"\\"?[Ll]"eftarrowtail"|
129             "\\"?[Ll]"ooparrowleft"|"\\"?[Hh]"ookrightarrow"|
130             "\\"?[Hh]"ookleftarrow"|"\\"?[Mm]"ultimap"|
131             "\\"?[Ll]"eftrightsquigarrow"|"\\"?[Rr]"ightsquigarrow"|
132             "\\"?[Tt]"woheadrightarrow"|"\\"?[Tt]"woheadleftarrow"|
133             "\\"?[Xx]"leftarrow"|"\\"?[Xx]"rightarrow"|
134             "\\"?[Oo]"verrightarrow"|"->"|"-&gt"";"?|"<-"|"&lt"";"?"-"|
135            "<--"|"&lt"";"?"--"|"-->"|"--&gt"";"?|"\\"?[Tt]"o"
136
137 SEP        "//"|";"|","|"&"|"&#9474;"|"&sect;"
138
139 ABS        "|"|"\\|"|"\\"?[Vv]"ert"
140
141 POW        "^"|"\\"?[Oo]"verset"|"\\"?[Uu]"nderset"|"\\"?[Ss]"tackrel"|
142             "\\"?[Oo]"verbrace"|[Pp]"ower"|[Ee]"xponent"
143
144 SQRT       "\\"?[Ss]"qrt"|"\\"?[Ss]"urd"
145
146 INF        "\\"?[Ii]"nfty"|[Ii]"nfinity"
147
148 LPAR       "("|"["|"\\{"|"\\"[Ll]"brace"|[Cc]"ases"|"\\"?[Ll]"ceil"|
149             "\\"?[Ll]"floor"
150
151 RPAR       ")"|"]"|"\\}"|"\\"?[Rr]"brace"|"\\"[Rr]"ceil"|"\\"[Rr]"floor"
152
153 ACCENT     "\\"?[Aa]"cute"|"\\"?[Gg]"rave"|"\\"?[Cc]"heck"|
154             "\\"?[Bb]"reve"|"\\"?[Tt]"ilde"|"\\"?[Hh]"at"|
155             "\\"?[Ww]"idehat"|"\\"?[Vv]"ec"
156
157 BINOM      "\\"?[Bb]"inom"|"\\"?[Tt]"binom"|"\\"?[Dd]"binom"|
158             "\\"?[Cc]"hoose"
159
160 MATRIX     [Mm]"atrix"|[Vv]"matrix"|[Bb]"matrix"|[Pp]"matrix"|
161             [Ss]"mallmatrix"|[Ii]"dentity"
162
163 MLINE      [Aa]"lign"|[Aa]"lignat"|[Aa]"rray"|"{"[\\|lcr]+"}"
164
165 FUNC       [f-h]("^""'"+)?|[Ff]"unc"("tion""s"?)?
166
167 MAXMIN     "\\"?[Mm]"ax"|"\\"?[Mm]"in"|[Mm]"aximum"|[Mm]"inimum"
168
169 CONST      [0-9]+|[0-9]+"."[0-9]+|[a-d]|"\\"?[Ii]"m"|"\\"?[Ii]"math"|
170             "\\"?[Jj]"math"|"\\"?[Bb]"bbk"|"\\"?[Ee]"ll"|"\\"?[Mm]"ho"|
171             "\\"?[Ww]"p"|"\\"?[Rr]"e"|"\\"?[Cc]"ircledS"|[Dd]"ay""s"?|
172             [Mm]"onth""s"?|[Yy]"ear""s"?|[Hh]"our""s"?|
173             [Mm]"in"("ute"("s"?))?|[Mm]"illi"?"second""s"?|
```

```
174             [Cc]"onstant""s"?|[Cc]"onst"|[Aa]"nnual"|[Bb]"it""s"?|
175             [Bb]"yte""s"?|([Mm]"ega"|[Gg]"iga"|[Tt]"era")("byte""s"?)?|
176             [Zz]"ero"|[Ss]"ignificand"|[Hh]"ertz"|[Tt]"hree"|
177             [Mm]"eter""s"?|[Tt]"wo"|[Mm]"ol"|[Cc]"ons"|[Nn]"il"|
178             [Cc]"ent""s"?|[Oo]"dd"|[Ee]"ven"|"EVEN"|[Nn]"umber""s"?|
179             [Nn]"um"|[Ff]"our""s"?|[Mm]"ile""s"?|[MmBb]"illion"|
180             [Tt]"rillion"|"#"|[Hh]"alf"|[Ff]"ive"|[Tt]"wenty"|
181             [Gg]"oogol"|[Oo]"ne"|"1st"|"2nd"|[Mm]"urata"|[Tt]"welve"|
182             [Ee]"ight""een"?|[Ww]"eek""s"?
183
184 PI          ("\\"|"&")?[Pp]"i"";"?|"3.14"[0-9]*
185
186 E           [Ee]|"2.71"[0-9]*
187
188 UPPER       [A-DF-RTV-W]("_"[A-Za-z0-9])?
189
190 TRIG        "\\"?[Ss]"in""c"?|"\\"?[Cc]"os"|"\\"?[Tt]"an"|"\\"?[Cc]"ot"|
191             "\\"?[Ss]"ec"|"\\"?[Cc]"sc"|[Tt]"rigonometric"|[Tt]"rig"|
192             "\\"?[Cc]"osec"
193
194 INVTG       "\\"?[Aa]"rcsin"|"\\"?[Aa]"rccos"|"\\"?[Aa]"rctan"|
195             "\\"?[Aa]"rccosec"|"\\"?[Aa]"rccot"|"\\"?[Aa]"rctg"|
196             "\\"?[Aa]"rcsec"|"\\"?[Aa]"rccsc"|"\\"?[Aa]"rcctg"
197
198 HYBOLIC     "\\"?[Ss]"inh"|"\\"?[Cc]"osh"|"\\"?[Tt]"anh"|
199             "\\"?[Cc]"oth"|"\\"?[Aa]"rccosh"|"\\"?[Aa]"rccoth"|
200             "\\"?[Aa]"rccsch"|"\\"?[Cc]"sch"|"\\"?[Aa]"rcsech"|
201             "\\"?[Ss]"ech"|"\\"?[Aa]"rcsinh"|"\\"?[Aa]"rctanh"|
202             "\\"?[Tt]"anhc"|"\\"?[Ss]"inhc"|"\\"?[Aa]"rccoshlemn"|
203             "\\"?[Aa]"rcsinhlemn"|"\\"?[Cc]"oslemn"|"\\"?[Ss]"inlemn"
204
205 LOGIC       "\\"?[Ff]"orall"|"\\"?[Ee]"xist""s"?|"\\"?[Nn]"exists"|
206             "\\"?[Tt]"herefore"|"\\"?[Bb]"ecause"|"\\"?[Aa]"nd"|
207             "\\"?[Oo]"r"|"\\"?[Ll]"or"|"\\"?[Vv]"ee"|"\\"?[Cc]"urlyvee"|
208             "\\"?[Bb]"igvee"|"\\"?[Aa]"nd"|"\\"?[Ll]"and"|"\\"?[Ww]"edge"|
209             "\\"?[Cc]"urlywedge"|"\\"?[Bb]"igwedge"|"\\"?[Bb]"ar"|
210             "\\"?[Oo]"verline"|"\\"?[Ll]"not"|"\\"?[Nn]"eg"|
211             ("\\"|"&")?[Nn]"ot"";"?|"\\"?[Bb]"ot"|"\\"?[Tt]"op"|
212             "\\"?[Vv]"dash"|"\\"?[Dd]"ashv"|"\\"?[Vv]"Dash"|
213             "\\"?[Mm]"odel""s"?|"\\"?[Vv]"vdash"|"\\"?[Nn]"vdash"|
214             "\\"?[Nn]"Vdash"|"\\"?[Nn]"vDash"|"\\"?[Nn]"VDash"|
215             "\\"?[Uu]"lcorner"|"\\"?[Uu]"rcorner"|"\\"?[Ll]"lcorner"|
216             "\\"?[Ll]"rcorner"|"_|_"|"\\"?[Xx]"or"|"\\"?[Nn]"and"|
217             "\\"?[Nn]"or"|"\\"?[Xx]"nor"|"\\"?[Cc]"onj"
218
219 FLCEIL      "\\"?[rl]"floor"|"\\"?[rl]"ceil"|"|_"|"_|"
220
```

```
221 TF        "\\"?[Tt]"rue"|"\\"?[Ff]"alse"
222
223 PERCENT "\\%"|[Pp]"ercentage"
224
225 PERM      [Pp]"perm"("utation""s"?)?
226
227 ALG       "\\"?[Dd]"et"|"\\"?[Dd]"im"|"\\"?[Dd]"eg"|":"|
228           "\\"?[Cc]"olon"|[Ii]"nv"|[Hh]"om"|[Pp]"gl"|
229           [Ii]"nd"|[Ss]"pec"|[Pp]"gu"|[Pp]"go"|"*"|
230           [Aa]"lg"|"rng"|"\\"?[Aa]"st"|"&#8988;"|
231           "&#8989;"|"&#8990;"|"&#8991;"|"*"
232
233 APP       [Oo]"scillator"|[Mm]"onad"|"MIRR"|[Pp]"oly"|[Ss]"ol"|
234           [Ii]"rrot"|[Ll]"atitude"|[Ll]"ongitude"|[Aa]"ltitude"
235
236 CALC      "\\"?[Kk]"e("r"|"i")|"\\"?[Bb]"e("r"|"i")|"\\"?[Ss]"up"|
237           [Ll]"an"|[Rr]"an"|[Rr]"es"|[Aa]"gm"|[Cc]"odim"|"^+"|
238           [Aa]"iry"
239
240 PART      "\\"?[Pp]"artial"|"&part"";"?
241
242 NABLA     "\\"?[Nn]"abla"
243
244 DISCR     [Tt]"ri"|[Aa]"ut"|[Aa]"dj""acent"?
245
246 DERIV     "d"[dxyuvwst]+|[Dd]"erivative"
247
248 INTEGR    "\\"?[io]+"nt"|"&#8747;"|"&#8748;"|"&#8749;"|
249           "&#8750;"|"&#8751;"|"&#8752;"|"&#8753;"|"&#8754;"|
250           "&#8755;"
251
252 LIMIT     "\\"?[Vv]"arlimsup"|"\\"?[Ll]"im"|"\\"?[Ll]"iminf"|
253           "\\"?[Ll]"imsup"|"\\varinjlim"|"\\"?[Vv]"arprojlim"
254
255 SETOP     "\\"?[Ee]"mpty"|"\\"?[Ee]"mptyset"|"\\"?[Vv]"arnothing"|
256           "\\"?[Ii]"n"|"\\"?[Nn]"otin"|"\\"?[Nn]"i"|"\\"?[Cc]"ap"|
257           "\\"?[Ss]"qcap"|"\\"?[Bb]"igcap"|"\\"?[Cc]"up"|"\\"?[Ss]"qcup"|
258           "\\"?[Bb]"igcup"|"\\"?[Bb]"igsqcup"|"\\"?[Uu]"plus"|
259           "\\"?[Bb]"iguplus"|"\\"?[Ss]"etminus"|"\\"?[Ss]"mallsetminus"|
260           "\\"?[Ss]"ubset"|"\\"?[Ss]"qsubset"|"\\"?[Ss]"upset"|
261           "\\"?[Ss]"qsupset"|"\\"?[Ss]"ubseteq"|"\\"?[Nn]"subseteq"|
262           "\\"?[Ss]"ubsetneq"|"\\"?[Vv]"arsubsetneq"|
263           "\\"?[Ss]"qsubseteq"|"\\"?[Ss]"upseteq"|"\\"?[Nn]"supseteq"|
264           "\\"?[Ss]"upsetneq"|"\\"?[Vv]"arsupsetneq"|
265           "\\"?[Ss]"qsupseteq"|"\\"?[Ss]"ubseteqq"|"\\"?[Nn]"subseteqq"|
266           "\\"?[Ss]"ubsetneqq"|"\\"?[Vv]"arsubsetneqq"|
267           "\\"?[Ss]"upseteqq"|"\\"?[Nn]"supseteqq"|"\\"?[Ss]"upsetneqq"|
```

```
268             "\\"?[Vv]"arsupsetneqq"|[Ii]"ntersection""s"|[Uu]"nion"|
269             "&#8741;"|[Ss]"uperset"
270
271 GEOM        "\\"?[Ss]"mallfrown"|"\\"?[Ff]"rown"|"\\"?[Pp]"arallel"|
272             "\\"?[Nn]"parallel"|"\\"?[Ss]"hortparallel"|
273             "\\"?[Nn]"shortparallel"|"\\"?[Pp]"erp"|"\\"?[Aa]"ngle""s"?|
274             "\\"?[Ss]"phericalangle"|"\\"?[Mm]"easuredangle"|
275             "\\"?[Cc]"irc"("le"|"umscribed")?|"\\"?[Bb]"ox"|
276             "\\"?[Bb]"lacksquare"|"\\"?[Dd]"iamond"|"\\"?[Ll]"ozenge"|
277             "\\"?[Bb]"lacklozenge"|"\\"?[Bb]"igstar"|"\\"?[Bb]"igcirc"|
278             "\\"?[Tt]"riangle"|"\\"?[Bb]"igtriangleup"|
279             "\\"?[Bb]"igtriangledown"|"\\"?[Vv]"artriangle"|
280             "\\"?[Tt]"riangledown"|"\\"?[Bb]"lacktriangle"|
281             "\\"?[Bb]"lacktriangledown"|"\\"?[Bb]"lacktriangleleft"|
282             "\\"?[Bb]"lacktriangleright"|[Ss]"quare"|"\\"?[Ss]"mile"|
283             [Aa]"rc"|"&ang;"|"&#8735;"|"&#8736;"|"&#8737;"|"&#8738;"|
284             [Ff]"illRad"|[Vv]"ol""ume"?|[Dd]"im"|[Aa]"rea"|[Rr]"ef"|
285             [Rr]"ot"|[Tt]"or"|"MCG"|[Aa]"ut"|[Ss]"ym"
286
287 HIST        [Ee]"rf"
288
289 NUMBER      "\\"?[Mm]"od"|"\\"?[Pp]"mod"|"\\"?[Bb]"mod"|[Ii]"rr"|[Cc]"fh"|
290             [Ss]"fh"|[Tt]"Fh"|[Ss]"opf""r"?|[Ll]"pf"|"?"|"\\"?[Mm]"id"|
291             "\\"?[Nn]"mid"|"\\"?[Ss]"hortmid"|"\\"?[Nn]"shortmid"
292
293 PROB        [Mm]"edian"|[Mm]"ean"|"p"|[Ss]"gn"|[Aa]"bs("|[Cc]"ov"|[Vv]"ar"|
294             "S"|"U"|[Uu]"niform"|[Cc]"onditional"|[Ee]"vent""s"?|
295             [Jj]"itter"|[Ff]"requency"|[Aa]"verage""d"?|[Aa]"mplitude"|
296             [Pp]"robabilities"|[Pp]"robability"|[Ww]"eibull"
297
298 RECRAT      [Ll]"cm"|[Gg]"cd"|[Mm]"ex"|"\\$"
299
300 TOPOL      [Cc]"url"|[Aa]"rf"|[Ss]"pan"|[Gg]"rad"|[Dd]"im"|[Hh]"omeo"|"-+"
301
302 COLOR       "\\"?[Cc]"olor{"."}"
303
304 SYMBOL      "\\"?[Dd]"iamondsuit"|"\\"?[Hh]"eartsuit"|"\\"?[Cc]"lubsuit"|
305             "\\"?[Ss]"padesuit"|"\\"?[Tt]"riangleleft"|
306             "\\"?[Tt]"riangleright"|"\\"?[Dd]"iagup"|"\\"?[Dd]"iagdown"|
307             "\\"?[Ee]"qcirc"|"\\"?[Cc]"irceq"|"\\"?[Tt]"riangleq"|
308             "\\"?[Bb]"umpeq"|"\\"?[Ii]"ntercal"|"\\"?[Bb]"arwedge"|
309             "\\"?[Vv]"eebar"|"\\"?[Dd]"oublebarwedge"|"\\"?[Bb]"etween"|
310             "\\"?[Pp]"itchfork"|"\\"?[Vv]"artriangleleft"|
311             "\\"?[Nn]"triangleleft"|"\\"?[Vv]"artriangleright"|
312             "\\"?[Nn]"triangleright"|"\\"?[Tt]"rianglelefteq"|
313             "\\"?[Nn]"trianglelefteq"|"\\"?[Tt]"rianglerighteq"|
314             "\\"?[Nn]"trianglerighteq"|"\\"?[Aa]"malg"|"\\"?[Pp]""|
```

```
315         "\\"?[Ss]""|"\\"?[Dd]"agger"|"\\"?[Dd]"dagger"|"\\"?[Ww]"r"|
316         "\\"?[Gg]"ame""s"?|"\\"?[Ff]"lat"|"\\"?[Nn]"atural"|
317         "\\"?[Ss]"harp"|"\\"?[Bb]"ull"
318
319
320 WSP     (" "|"\\"?[Qq]"quad"|"\\"?[Qq]"uad"|"\\"|"\\;"|"\\>"|
321         "\\,"|"\\!")+
```