BONS MOTIVOS PARA ESTUDAR MATEMÁTICA DISCRETA

ANTÔNIO CÉSAR BALEEIRO ALVES
AUGUSTO SILVA
KÁTIA KELVIS CASSIANO
MAIANA SANTOS LOPES

Resumo: *este artigo trata de fórmulas genéricas para a soma de séries finitas de potências de inteiros e da dificuldade de avaliação da expressão da probabilidade de Erlang-B. Para o primeiro problema, propõe-se uma nova fórmula para séries de potências de inteiros e para o segundo apresenta-se o algoritmo de Qiao como uma elegante solução do problema.*

Palavras-chave: *séries, números de Bernoulli e números de Stirling de segunda espécie, Modelo de Erlang*

Neste trabalho, tratamos da obtenção de fórmulas genéricas para a soma de séries finitas de potências de inteiros e da avaliação da expressão da probabilidade de bloqueio de Erlang-B. Esses problemas ocorrem em áreas aplicadas, como a análise de complexidade de algoritmos e o dimensionamento de sistemas telefônicos, e suas soluções fundamentam-se em teorias estudadas na disciplina Matemática Discreta. Os dois problemas possuem em comum o fato de exigirem a avaliação de séries e o estudo de funções geratrizes. O cálculo de séries de potências de inteiros e a existência de algoritmos eficientes para obtenção de fórmulas para computá-las são relevantes na elaboração de aplicativos em linguagem simbólica, enquanto a aplicação do modelo de Erlang expõe as dificuldades de processamento numérico nos computadores atualmente disponíveis. Para o primeiro problema abordado propõe-se uma fórmula para séries genéricas de potências de inteiros baseada no Teorema das Colunas do Triângulo de Pascal, e, para o segundo problema, sugere-se uma abordagem que expõe as limitações de representação nos modernos computadores. Para contornar as dificuldades numéricas no cálculo direto com o modelo de Erlang, apresentamos um consagrado algoritmo proposto por Qiao em 1998.

Normalmente, é praxe, na primeira aula de uma disciplina, o professor esforçar-se para apresentar aos seus novos alunos bons motivos para estudar o conteúdo que será desenvolvido no período que naquele momento se inicia. Matemática Discreta é uma dessas disciplinas cujo conteúdo tem caráter formal dado os teoremas e princípios que lhe são próprios e que são de fundamental importância em áreas aplicadas, como Pesquisa Operacional, Complexidade de Algoritmos e Criptografia. Um artifício normalmente usado como elemento motivador consiste em discorrer sobre a biografia de alguns personagens que são ícones incontestáveis desse campo da matemática, tais como Cantor (o criador da Teoria dos Conjuntos) e Paul Erdös, entre outros. Sobre Erdös, quando de sua morte em 1996, os jornalistas e biógrafos escreveram: “foi o mais fecundo matemático do século XX, tendo escrito mais de mil artigos em revistas indexadas”. É imprescindível também fazer uma breve preleção sobre a história do famoso Teorema de Fermat, que se manteve por quase 400 anos sem demonstração até ser provado em 1995 pelo inglês A. J. Willes. Todavia, nada é mais motivador quando o estudante enxerga de maneira clara a possibilidade de aplicar os conhecimentos que lhe serão apresentados. Com a preocupação de prover elementos motivadores para o estudo de Matemática Discreta, escrevemos este artigo. Buscamos apresentar alguns aspectos primordiais da Matemática Discreta, enfatizando a aplicação dos temas aqui abordados e o tratamento computacional. São analisados dois problemas de grande interesse em áreas aplicadas: (1) a obtenção de fórmulas para cálculo de séries finitas de potências de inteiros; (2) o modelo de Erlang para dimensionamento de centrais telefônicas com enfoque nas dificuldades de processamento numérico. Para validar os desenvolvimentos apresentados, ao final, são elaboradas situações que exemplificam a aplicação dos temas estudados e que comprovam a eficiência dos métodos.

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

FUNÇÃO GERATRIZ

Função geratriz é uma função da qual retiramos uma seqüência de números. Ao obtermos a função geratriz de uma seqüência, diversas informações são extraídas dos números que a compõem, até a de encontrar uma fórmula para a seqüência a provar identidades. O estudo de propriedades analíticas da função leva ao conhecimento de como a seqüência se comporta. Geratrizes podem também ser de grande auxílio quando desejamos demonstrar teoremas em Teoria dos Números, permitindo, por exemplo, inferir delas temas tão áridos quanto 'números primos' (WILF, 1994). Neste trabalho, estamos particularmente interessados no emprego de funções geratrizes para provar identidades, e o estudo se limita a obter séries de potências, sem nos preocuparmos com aspectos analíticos da função ou da série, tais como convergência da série e existência da função.

Seguem as definições extraídas da referência (WILF, 1994).

Definição 1. Designamos como série de potências formal uma expressão da forma $a_0 + a_1 x + a_2 x^2 + \cdots$, em que a seqüência $\{a_n\}_0^\infty$ é chamada seqüência de coeficientes.

Definição 2. O símbolo $f \overset{\text{spo}}{\leftrightarrow} \{a_n\}_0^\infty$ significa que a série f é a função geratriz da série de potência ordinária para a seqüência $\{a_n\}_0^\infty$. Quer dizer que $f = \sum_n a_n x^n$.

A título de exemplos, as funções e^x, $\operatorname{sen} x$ e $(1+x)^n$ possuem as seguintes séries ordinárias de potências, respectivamente, $\sum_{k\geq 0} \frac{x^k}{k!}$, $\sum_{k\geq 0} (-1)^k \frac{x^{2k+1}}{(2k+1)}$ e $\sum_{k=0}^{n} \binom{n}{k} x^k$. Chamamos a atenção do leitor para a sutil diferença de notação dos limites do somatório: nos dois primeiros exemplos, o limite superior é ∞ e, no terceiro, o limite superior é o inteiro n. Adotaremos esta notação ao longo do texto.

SÉRIES FINITAS DE POTÊNCIAS DE INTEIROS
E NÚMEROS DE BERNOULLI

A obtenção de uma fórmula para séries do tipo $\sum_{k=0}^{n} k^m$, com $m = 1, 2, 3, \cdots$, para um dado inteiro positivo n, é um desses problemas recorrentes em análise de complexidade e surge muitas vezes em meio a certos desenvolvimentos. Posteriormente mostraremos alguns exemplos de aplicações. Este problema pode ser solucionado com o uso de funções geratrizes. O desafio que se apresenta aqui é o seguinte: "como obter uma expressão geral para o resultado desse somatório?" Embora este seja um problema intensivamente estudado, ele desperta grande interesse em programadores da área de matemática computacional e, às vezes, reserva novidades.

A Proposição 1 é extraída de (WILF, 1994).

Proposição 1. Seja a função geratriz $\frac{x(e^{nx}-1)}{e^x-1}$, com n um número inteiro positivo, e seja $S(n,m) = \sum_{k=0}^{n} k^m$ a série genérica de $m-$ésimas potências de inteiros para um especificado inteiro positivo m, então $S(n,m) = \frac{1}{m+1}\sum_{j=1}^{m+1}\begin{pmatrix} m+1 \\ j \end{pmatrix} B_{m+1-j}(n+1)^j$, em que B_{m+1-j} é o $m+1-j-$ésimo número da seqüência de Bernoulli, e $\begin{pmatrix} m+1 \\ j \end{pmatrix}$ é o número binomial $\frac{(m+1)!}{j!(m+1-j)!}$.

Prova: Partiremos do pressuposto de que, se x é um número real, $\frac{x}{e^x-1}$ é a função geratriz da soma $\sum_{r\geq 0}\frac{B_r x^r}{r!}$, em que B_r é o $r-$ésimo número de Bernoulli de ordem zero. Então, $\frac{x}{e^x-1} = \sum_{r\geq 0}\frac{B_r x^r}{r!}$.

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

O desenvolvimento em série de Taylor de $e^{nx}-1$ é a soma $\sum_{j\geq 1}\frac{n^j x^j}{j!}$. Portanto,

$$\begin{aligned}
\frac{x(e^{nx}-1)}{e^x-1} &= \left(\frac{x}{e^x-1}\right)(e^{nx}-1)\\
&= \left(\sum_{r\geq 0}\frac{B_r x^r}{r!}\right)\left(\sum_{j\geq 1}\frac{n^j x^j}{j!}\right)\\
&= \sum_{r\geq 0}\frac{x^r}{r!}\left\{\sum_{j\geq 1}\binom{r}{j}B_{r-j}n^j\right\}
\end{aligned}$$

Olhando a função geratriz $\frac{x(e^{nx}-1)}{e^x-1}$ de outro modo, temos

$$\begin{aligned}
\frac{x(e^{nx}-1)}{e^x-1} &= x\left(\frac{e^{nx}-1}{e^x-1}\right)\\
&= x(1+e^x+e^{2x}+\cdots+e^{(n-1)x})\\
&= x\sum_{k=0}^{n-1}\sum_{m\geq 0}\frac{k^m x^m}{m!}\\
&= \sum_{m\geq 0}\frac{x^{m+1}}{m!}\sum_{k=0}^{n-1}k^m\\
&= \sum_{m\geq 0}\frac{x^{m+1}}{m!}S(n-1,m)
\end{aligned}$$

em que, $S(n-1,m)$ é a soma das $m-$ésimas potências dos inteiros $0,1,2,\cdots,n-1$. Finalmente, comparamos os coeficientes de x^r da expressão do primeiro desenvolvimento e de x^{m+1}

do último desenvolvimento e, simultaneamente, ajustamos os índices fazendo $r = m+1$ e trocando $n-1$ por n. Isto nos permite escrever a expressão (1) concluindo, assim, a demonstração.

$$S(n,m) = \sum_{k=0}^{n} k^m = \frac{1}{m+1} \sum_{j=1}^{m+1} \binom{m+1}{j} B_{m+1-j} (n+1)^j \quad (1)$$

De acordo com Wilf (1994), os polinômios de Bernoulli são, no caso geral, obtidos da função geratriz conforme é mostrado a seguir:

$$\frac{xe^{zx}}{e^x - 1} = \sum_{r \geq 0} \frac{B_r(z)x^r}{r!}$$

Do caso específico da Proposição 1 temos que os números de Bernoulli, que aparecem na fórmula (1), são os números $B_0(0), B_1(0), \cdots, B_m(0)$, ou seja, $z = 0$. Cumpre-nos apresentar então uma maneira eficiente do ponto de vista computacional de obter seqüências de números de Bernoulli. Uma forma possível é o emprego do algoritmo de Akiyama-Tanigawa (KANEKO, 2000), que permite representar os números de Bernoulli na primeira coluna de um triângulo que lembra o Triângulo de Pascal, cujos passos são:

Algoritmo 1. Geração de uma seqüência de Bernoulli.

Dado o número inteiro m

Para $k = 0,1,2,3,\cdots,m$

$a(0,k) = 1/(k+1)$

Para $r = 1,2,3,\cdots,m$

Para $j = 0,1,2,3,\cdots,m$

$a(r,j) = (j+1)[a(r-1,j) - a(r-1,j+1)]$

Fim.{ $a(0,0), a(1,0), a(2,0), \cdots, a(m,0)$ são $B_0(1), B_1(1), \cdots, B_m(1)$ }.

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

Esta versão do algoritmo de Akiyama-Tanigawa é referenciada na literatura como algoritmo-A, e os números da coluna zero da matriz $a(.)$, $a(r,0)$ com $r \geq 1$ são os números da seqüência de Bernoulli, $B_0(1)$, $B_1(1)$, $\cdots$, $B_m(1)$, ou seja, $z = 1$. Precisamos de um método para obter os números que aparecem na fórmula (1).

Para resolver tal problema, a referência (CHEN, 2001) sugere uma modificação no Algoritmo 1 em que propõe uma diferente relação de recorrência partindo da mesma seqüência inicial. Daí resulta o Algoritmo 2, referido na literatura como algoritmo-B.

Algoritmo 2. Geração de uma seqüência de Bernoulli.

Dado o número inteiro m

Para $k = 0,1,2,3,\cdots,m$

$a(0,k) = 1/(k+1)$

Para $r = 1,2,3,\cdots,m$

Para $j = 0,1,2,3,\cdots,m$

$a(r,j) = j\,a(r-1,j) - (j+1)a(r-1,j+1)$

Fim. { $a(0,0)$, $a(1,0)$, $a(2,0)$, $\cdots, a(m,0)$ são $B_0(0)$, $B_1(0)$, $\cdots, B_m(0)$ }.

Ao aplicarmos os Algoritmos 1 e 2, obtemos as seguintes seqüências de Bernoulli, respectivamente, que curiosamente diferem entre si apenas pelo elemento B_1,

$$1,\ 1/2, 1/6, 0, -1/30, 0, 1/42, 0, -1/30, 0, 5/66, \cdots$$

$$1,\ -1/2, 1/6, 0, -1/30, 0, 1/42, 0, -1/30, 0, 5/66, \cdots$$

O Algoritmo 1 permite escrever a fórmula (2) com sensível diferença em relação à primeira fórmula apresentada, n^j em lugar de $(n+1)^j$.

$$S(n,m) = \sum_{k=0}^{n} k^m = \frac{1}{m+1} \sum_{j=1}^{m+1} \binom{m+1}{j} B_{m+1-j}\, n^j \qquad (2)$$

em que, B_{m+1-j} é o $m+1-j$ – ésimo número da seqüência de Bernoulli, $B_0(1)$, $B_1(1)$, $\cdots, B_m(1)$.

As fórmulas (1) e (2) não são as únicas soluções possíveis para a série genérica finita de potências de inteiros. Demonstramos a seguir, nesta seção, uma nova fórmula para o cálculo da série $\sum_{k=0}^{n} k^m$. Esta nova fórmula para o cálculo da série $\sum_{k=0}^{n} k^m$, com $m = 1, 2, 3, \cdots$, pode ser deduzida com o o Teorema das Colunas do Triângulo de Pascal, que é expresso da seguinte forma:

$$\sum_{i=0}^{r} \binom{p}{p+i} = \binom{p+1}{p+r+1}, \text{ para } p = 0,1,2,\cdots.$$

Tal teorema pode ser encontrado na referência (MORGADO, 2004) em que é facilmente demonstrado aplicando-se sucessivamente a relação de Stifel. Consideremos então a Proposição 2 para a qual apresentamos a prova dada em seguida.

Proposição 2. Sejam n e m números inteiros positivos e consideremos $S(n,m) = \sum_{k=0}^{n} k^m$ a série genérica de $m-$potências de inteiros. Então $S(n,m) = \sum_{j=1}^{m} (-1)^{m-j}\, j! \left\{ \begin{matrix} m \\ j \end{matrix} \right\} \binom{n+j}{j+1}$, em que $\left\{ \begin{matrix} m \\ j \end{matrix} \right\}$ é um número de Stirling de segunda espécie e $\binom{n+j}{j+1}$ é o número binomial $\dfrac{(n+j)!}{(j+1)!\,(n-1)!}$.

Prova: Iniciamos a demonstração procurando expressar k, k^2, k^3, etc., em termos de números binomiais em $k, k+1$, $k+2$, etc., como a seguir:

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

$$k = \binom{k}{1},\ k^2 = 2\binom{k+1}{2} - \binom{k}{1},\ k^3 = 6\binom{k+2}{3} - 6\binom{k+1}{2} + \binom{k}{1}.$$

Olhando mais de perto as expressões, observamos que os números de Stirling emergem naturalmente:

$$k^3 = 1\times 3!\binom{k+2}{3} - 3\times 2!\binom{k+1}{2} + 1\times 1!\binom{k}{1} \to \left\{ {3 \atop 1} \right\} 3!\binom{k+2}{3} - \left\{ {3 \atop 2} \right\} 2!\binom{k+1}{2} + \left\{ {3 \atop 3} \right\} 1!\binom{k}{1}$$

Conseqüentemente, por Indução Matemática, para a $m-$ésima potência podemos escrever a seguinte expressão generalizada:

$$k^m = \sum_{j=1}^{m} (-1)^{m-j}\, j! \left\{ {m \atop j} \right\} \binom{k+j-1}{j}.$$

Introduzimos o somatório em ambos os lados desde $k=1$ até $k=n$,

$$\sum_{k=1}^{n} k^m = \sum_{k=1}^{n} \sum_{j=1}^{m} (-1)^{m-j}\, j! \left\{ {m \atop j} \right\} \binom{k+j-1}{j},$$

donde, no segundo membro, é possível transpor $\sum_{k=1}^{n}$ para a posição imediatamente à esquerda do número binomial $\binom{k+j-1}{j}$, assim:

$$\sum_{k=1}^{n} k^m = \sum_{j=1}^{m} (-1)^{m-j}\, j! \left\{ {m \atop j} \right\} \sum_{k=1}^{n} \binom{k+j-1}{j}.$$

O somatório mais à direita é prontamente avaliado com o uso do Teorema das Colunas, conforme a expressão:

$$\sum_{k=1}^{n} \binom{k+j-1}{j} = \binom{n+j}{j+1}.$$

Finalmente, obtemos a expressão geral para $S(n,m) = \sum_{k=0}^{n} k^m$,

$$S(n,m) = \sum_{k=0}^{n} k^m = \sum_{j=1}^{m} (-1)^{m-j}\, j! \begin{Bmatrix} m \\ j \end{Bmatrix} \binom{n+j}{j+1}$$

a qual é simplificada,

$$S(n,m) = \sum_{k=0}^{n} k^m = \sum_{j=1}^{m} \frac{(-1)^{m-j} \begin{Bmatrix} m \\ j \end{Bmatrix} (n+j)^{\underline{j+1}}}{j+1}$$

em que, $(n+j)^{\underline{j+1}}$ é a notação de Knuth.

Fica então concluída a demonstração da Proposição 2, que acreditamos ser uma contribuição original deste trabalho. Uma fórmula alternativa, também baseada nos números de Stirling, que permite encontrar uma expressão geral para $\sum_{k=0}^{n} k^m$ com $m = 1, 2, 3, \cdots$, para um inteiro positivo n, é proposta na referência (MILLER, 2000),

$$S(n,m) = \sum_{k=0}^{n} k^m = \sum_{j=1}^{m} \frac{\begin{Bmatrix} m \\ j \end{Bmatrix} (n+1)^{\underline{j+1}}}{j+1}$$

em que, $(n+1)^{\underline{j+1}}$ é a notação de Knuth.

Nas fórmulas (3) e (4), a notação de Knuth significa o seguinte:

$$(n+j)^{\underline{j+1}} = (n+j)(n+j-1)(n+j-2)\cdots n,$$

$$(n+1)^{\underline{j+1}} = (n+1)n(n-1)\cdots(n-j)$$

Uma vez mais chamamos a atenção do leitor para as sutis diferenças observadas entre as fórmulas (3) e (4), as quais resolvem o mesmo problema proposto. Quanto à fórmula (3), para o caso

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

geral da soma com expoente m, temos:

$$\sum_{k=0}^{n} k^m =$$

$$= (-1)^{m-1}\frac{1}{2}\left\{ {m \atop 1} \right\}(n+1)n + (-1)^{m-2}\frac{1}{3}\left\{ {m \atop 2} \right\}(n+2)(n+1)n + \cdots + \frac{1}{m+1}\left\{ {m \atop m} \right\}(n+m)(n+m-1)\cdots n$$

Já para a fórmula (4), para o caso geral da soma com expoente m, temos:

$$\sum_{k=0}^{n} k^m = \frac{1}{2}\left\{ {m \atop 1} \right\}(n+1) + \frac{1}{3}\left\{ {m \atop 2} \right\}(n+1)n + \cdots + \frac{1}{m+1}\left\{ {m \atop m} \right\}(n+1)n(n-1)\cdots(n-m)$$

UMA REFLEXÃO SOBRE AS FÓRMULAS OBTIDAS

Ao nos debruçarmos sobre este tema, extraímos naturalmente algumas constatações:

- poucos problemas na Matemática apresentam tantas e diversificadas fórmulas para sua solução;

- a comparação das soluções conduz a expansões distintas, mas todas conduzem igualmente à solução do problema;

- notamos uma 'beleza estética' nas soluções apresentadas, uma vez que os números que compõem os coeficientes ou são números combinatórios, como é o caso dos números de Stirling de segunda espécie (partições) e os números binomiais (subconjuntos), ou são números racionais que estão associados ao cálculo de séries infinitas, uma vez que os números de Bernoulli aparecem nos coeficientes nas expansões de Maclaurin de $\operatorname{tg} x$, $\operatorname{cotg} x$, $\operatorname{cosec} x$, $\tanh x$, $\operatorname{cotgh} x$ e $\cosh x$.

A extraordinária característica de exibir tantas soluções surpreenderia ainda mais o leitor se ele fizesse uma pesquisa bibliográfica sobre as inúmeras formas possíveis para o somatório $\sum_{k=0}^{n} k^m$. Para tanto, sugerimos uma pesquisa em (ROSEN, 2000).

Uma preocupação natural do programador, ao comparar as duas fórmulas apresentadas para cálculo de $\sum_{k=0}^{n} k^{m}$, é quanto à eficiência computacional. Qual das expressões apresentadas requer menor esforço de máquina?

Para m inteiro positivo e $m \geq j+1$, os números de Stirling de segunda espécie podem ser obtidos através da relação de recorrência,

$$\left\{ \begin{matrix} m \\ j \end{matrix} \right\} = \left\{ \begin{matrix} m-1 \\ j-1 \end{matrix} \right\} + j \left\{ \begin{matrix} m-1 \\ j \end{matrix} \right\} \text{ para } j = 2, 3, 4, \cdots, \text{com} \left\{ \begin{matrix} m \\ 1 \end{matrix} \right\} = 1$$

e $\left\{ \begin{matrix} m \\ m \end{matrix} \right\} = 1$.

Definindo *iops* (*integers operations*) como uma operação entre inteiros da forma $a \leftarrow a \pm b$, os m números de Stirling para serem obtidos através desta relação de recorrência requerem $O(m^{2})$ *iops*, enquanto a avaliação da soma através das fórmulas (1) e (2) exige claramente maior número de operações, uma vez que ambas necessitam do cálculo de números binomiais também.

No entanto, nem sempre a eficiência computacional é o aspecto mais relevante. Muitas vezes, em atividades de sala de aula, o fator mais atrativo é a possibilidade de avaliar uma expressão de uma forma mais direta, praticamente sem esforço excessivo de memória. Contudo, o aspecto mais importante é que os métodos até então apresentados para obt enção de expressões para avaliação de $\sum_{k=0}^{n} k^{m}$ apresentam-se atraentes para implementação

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

em programação simbólica, tal como faz o programa MATLAB® através das funções simbólicas syms k n e symsum(k^m,0,n), para um dado inteiro m (HANSELMAN, 2003).

O MODELO DE ERLANG

Em sistemas de comunicações, a intensidade de tráfego é uma variável aleatória que é função do tempo, podendo experimentar períodos de ociosidade e de congestionamento. O projeto de um dado sistema telefônico deve garantir que a probabilidade de haver congestionamento seja menor ou no máximo igual a um certo valor considerado razoável. Esta probabilidade é designada como probabilidade de bloqueio. Dentre os modelos estocásticos de tráfego, o modelo conhecido como Erlang-B, é o indicado para medir a taxa de bloqueio de requisições quando o tráfego é aleatório e não existe fila de espera (ERLANG, 1917).

No modelo de Erlang, as chamadas chegam a um enlace conforme um processo de Poisson de taxa conhecida, λ, que é uma medida do número médio de chamadas por unidade de tempo. Os tempos de serviços seguem a distribuição exponencial, com taxa μ. O tráfego é caracterizado pela relação λ/μ, que designaremos por ρ. O enlace é composto por circuitos (troncos ou linhas), e uma chamada é bloqueada e perdida se todos os circuitos estiverem ocupados. Em caso contrário, a chamada é aceita e utiliza um circuito durante o seu tempo de retenção. Estudos realizados sobre a distribuição estatística de Poisson para cálculo da probabilidade de bloqueio podem ser sintetizados na fórmula B de Erlang, ou fórmula de perda de Erlang. Com base no modelo de Erlang-B, obtém-se a fórmula (5) para cálculo da probabilidade estacionária P_b de que todos os circuitos estejam ocupados no enlace sob consideração (ERLANG, 1909).

$$P_b = \frac{\frac{\rho^N}{N!}}{\sum_{k=0}^{N} \frac{\rho^k}{k!}}$$

Cabe notar que o modelo de Erlang é apenas parte de uma área de estudo conhecida como Teoria das Filas, que é capaz de tratar problemas complexos de tráfego em redes que são compartilhadas pelos mais diversos tipos de informação, tais como voz, dados e multimídia. No caso do tráfego de chamadas telefônicas, o modelo de Erlang-B se aplica sem restrições.

A avaliação numérica da probabilidade P_b, através da aplicação direta da fórmula (5) oferece dificuldades uma vez que o número N de linhas pode assumir valores muito altos. O problema advém tanto do cálculo do fatorial quanto do cálculo das potências de ρ à medida que N cresce.

PROCESSAMENTO NUMÉRICO DO MODELO DE ERLANG-B E O ALGORITMO DE QIAO

A necessidade do uso de aritmética em ponto flutuante e da portabilidade de aplicações levou à elaboração de um padrão para processamento aritmético em ponto flutuante. Em 1985, o Standards Committee of the IEEE Computer Society e o American National Standards Institute aprovaram o padrão IEEE 754, que rapidamente foi adotado pela indústria de *hardware* (IEEE, 1985).

Neste padrão estão definidos quatro formatos de representação, divididos em dois grupos: básico e estendido (*extended*, em inglês). Cada grupo possui uma representação para precisão simples e uma representação para precisão dupla. Para garantia de portabilidade mínima, as máquinas disponíveis normalmente suportam a precisão simples e a precisão dupla do grupo básico.

O formato de precisão simples básico utiliza 32 bits para codificação em ponto flutuante. Sua capacidade de representação abrange o intervalo $[1{,}18\times10^{-38},\ 3{,}40\times10^{+38}]$. O formato de precisão dupla básico utiliza 64 bits e sua capacidade de representação abrange o intervalo $[2{,}23\times10^{-308},\ 1{,}79\times10^{+308}]$.

Para N não muito grande, a utilização do computador para o cálculo de $N!$ e de ρ^N requer certos cuidados por parte do programador diante das limitações de representação aritmética das máquinas atualmente disponíveis. Por exemplo, para

$\rho = 100$ e $N > 154$, a potência ρ^N leva a *overflow*, enquanto 171! também leva a *overflow* (QIAO, 1998). Isto ocorre pelas seguintes razões: o número avaliado ou é maior que o maior valor em ponto flutuante possível de ser representado $(1,79 \times 10^{+308})$, ou ocorrem indeterminações da forma ∞/∞ que são representadas como NaN (*Not a Number*), uma representação específica de resultado inválido ou indeterminado (IEEE, 1985). A ocorrência de um NaN pode ser propagada na computação ou, pior ainda, provocar a interrupção da computação (INTEL, 2006). Todavia, existem métodos para contornar essas dificuldades.

A primeira tentativa para sanar os problemas mencionados é naturalmente procurar expressar o fatorial de N por meio da fórmula de Stirling.

$$N! \approx \left(\frac{N}{e}\right)^N \sqrt{2\pi N}$$

O erro dessa fórmula não é superior a $1/12N$ no cálculo do fatorial. Quanto ao somatório do denominador, tendo em vista que a função geratriz da série $\sum_{k\geq 0} \frac{\rho^k}{k!}$ é e^ρ, para N suficientemente grande, temos:

$$\sum_{k=0}^{N} \frac{\rho^k}{k!} \approx e^\rho .$$

Ao substituirmos essas relações na expressão da probabilidade de bloqueio, P_b, obtemos a seguinte forma alternativa para (6).

$$P_b \approx \frac{\rho^N e^{N-\rho}}{N^N \sqrt{2\pi N}}$$

Como é possível notar em (7), infelizmente a potência N^N está presente, o que traz dificuldade de avaliação da potência para um expoente grande.

A solução vislumbrada por nós para este dilema consiste em aplicar o logaritmo a ambos os membros da expressão (7).

$$\ln(P_b) \approx (N-\rho) - \tfrac{1}{2}\ln(2\pi) - (N+\tfrac{1}{2})\ln(N) + N\ln(\rho)$$

Naturalmente, o cálculo de P_b será efetuado ao avaliar a expressão (8).

$$P_b \approx e^{\ln(P_b)}$$

Desta forma, imaginamos contornar as dificuldades inerentes ao tratamento de números de grande porte. Os testes serão mostrados na próxima seção.

A solução proposta por Qiao na referência (QIAO, 1998) para avaliar a expressão (5) é criativa e alia robustez e eficiência. Tal solução consiste em escrever a fórmula (5) de outro modo. Dividimos o numerador e o denominador de (5) por $\rho^N / N!$, donde obtemos:

$$P_b = \frac{\frac{\rho^N}{N!}}{\sum_{k0}^{N}\frac{\rho^k}{k!}} = \frac{1}{1+\frac{N!}{(N-1)!\,\rho}+\frac{N!}{(N-2)!\,\rho^2}\cdots+\frac{N!}{2!\rho^{n-2}}+\frac{N!}{1!\rho^{n-1}}+\frac{N!}{\rho^n}}$$

$$= \frac{1}{1+\frac{N}{\rho}+\frac{N}{\rho}\frac{(N-1)}{\rho}+\cdots+\frac{N}{\rho}\frac{(N-1)}{\rho}\cdots\frac{1}{\rho}}$$

O denominador pode então ser expresso com o uso da regra de Horner, como a seguir:

$$P_b = \frac{1}{1+\frac{N}{\rho}+\frac{N}{\rho}\frac{(N-1)}{\rho}+\cdots+\frac{N}{\rho}\frac{(N-1)}{\rho}\cdots\frac{1}{\rho}} = \frac{1}{1+\frac{N}{\rho}(\cdots(1+\frac{2}{\rho}(1+\frac{1}{\rho})\ \cdots)}$$

A solução proposta por Qiao na referência (QIAO, 1998) é resumida no Algoritmo 3.

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

Algoritmo 3. Cálculo da probabilidade de bloqueio pelo método de Qiao.

Dado o número inteiro N e dada a intensidade de tráfego ρ

Se $\rho = 0$ então $P_b = 0$

Em caso contrário

Faça $s \leftarrow 0$

Para $k = 1, 2, 3, \cdots, N$

$s = (1 + s)\left(k/\rho \right)$

$P_b = 1/(1 + s)$

Fim. { P_b exato}

Supondo que o esforço exigido para o computador realizar uma multiplicação é idêntico ao requerido por uma divisão ou por uma adição, o Algoritmo 3 requer cerca de $3N$ operações de ponto flutuante.

APLICAÇÕES E ANÁLISE DE RESULTADOS

Esta seção é dedicada a aplicações das fórmulas de cálculo da soma $\sum_{k=0}^{n} k^m$ e é também endereçada à análise das fórmulas propostas para avaliação da probabilidade de bloqueio do modelo Erlang-B.

Análise Comparativa das Fórmulas

Obteremos as fórmulas (1) a (4) para o seguinte somatório $\sum_{k=0}^{n} k^4$.

Fórmula (1) – utiliza os números de Bernoulli obtidos do Algoritmo 2:

$$\sum_{k=0}^{n} k^4 = \frac{1}{5}\left[\binom{5}{1}B_4(n+1) + \binom{5}{2}B_3(n+1)^2 + \binom{5}{3}B_2(n+1)^3 + \binom{5}{4}B_1(n+1)^4 + \binom{5}{5}B_0(n+1)^5\right]$$

$$= \frac{1}{5}\left[-\frac{5}{30}(n+1) + \frac{10}{6}(n+1)^3 - \frac{5}{2}(n+1)^4 + (n+1)^5\right] = \frac{(n+1)^5}{5} - \frac{(n+1)^4}{2} + \frac{(n+1)^3}{3} - \frac{(n+1)}{30}$$

Fórmula (2) – utiliza os números de Bernoulli obtidos do Algoritmo 1:

$$\sum_{k=0}^{n} k^4 = \frac{1}{5}\left[\binom{5}{1}B_4 n + \binom{5}{2}B_3 n^2 + \binom{5}{3}B_2 n^3 + \binom{5}{4}B_1 n^4 + \binom{5}{5}B_0 n^5\right]$$

$$= \frac{1}{5}\left[-\frac{5}{30}n + \frac{10}{6}n^3 + \frac{5}{2}n^4 + n^5\right] = \frac{n^5}{5} + \frac{n^4}{2} + \frac{n^3}{3} - \frac{n}{30}$$

Fórmula (3) – fórmula proposta:

$$\sum_{k=0}^{n} k^4 = \frac{1}{2}(-1)^3\left\{{4 \atop 1}\right\}(n+1)^{\underline{2}} + \frac{1}{3}(-1)^2\left\{{4 \atop 2}\right\}(n+2)^{\underline{3}} + \frac{1}{4}(-1)^1\left\{{4 \atop 3}\right\}(n+3)^{\underline{4}} + \frac{1}{5}(-1)^0\left\{{4 \atop 4}\right\}(n+4)^{\underline{5}}$$

$$= -\frac{1}{2}(n+1)^{\underline{2}} + \frac{7}{3}(n+2)^{\underline{3}} - \frac{6}{4}(n+3)^{\underline{4}} + \frac{1}{5}(n+4)^{\underline{5}}$$

$$= \frac{(n+4)(n+3)(n+2)(n+1)n}{5} - \frac{6(n+3)(n+2)(n+1)n}{4} + \frac{7(n+2)(n+1)n}{3} - \frac{(n+1)n}{2}$$

Fórmula (4) (MILLER, 2000):

$$\sum_{k=0}^{n} k^4 = \frac{1}{2}\left\{{4 \atop 1}\right\}(n+1)^{\underline{2}} + \frac{1}{3}\left\{{4 \atop 2}\right\}(n+1)^{\underline{3}} + \frac{1}{4}\left\{{4 \atop 3}\right\}(n+1)^{\underline{4}} + \frac{1}{5}\left\{{4 \atop 4}\right\}(n+1)^{\underline{5}}$$

$$= \frac{1}{2}(n+1)^{\underline{2}} + \frac{7}{3}(n+1)^{\underline{3}} + \frac{6}{4}(n+1)^{\underline{4}} + \frac{1}{5}(n+1)^{\underline{5}}$$

$$= \frac{(n+1)n(n-1)(n-2)(n-3)}{5} + \frac{6(n+1)n(n-1)(n-2)}{4} + \frac{7(n+1)n(n-1)}{3} + \frac{(n+1)n}{2}$$

As expressões obtidas para a série escolhida são distintas, porém conduzem ao mesmo valor numérico para um dado inteiro positivo n.

Custo Computacional do Algoritmo de Decomposição LU

Algoritmo 4. Decomposição LU de uma matriz A não esparsa (isto é, densa).

Dada uma matriz quadrada A, $n \times n$, com suas submatrizes principais não-singulares.

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

Para $k = 1, \cdots, n-1$

Para $i = k+1, \cdots, n$

Se $a(k,k) \neq 0$, $m \leftarrow a(i,k) \big/ a(k,k)$

$a(i,k) \leftarrow 0$

$l(i,k) \leftarrow m$

Para $j = k+1, \cdots, n$

$a(i,j) \leftarrow a(i,j) - m\,a(k,j)$

Fim. {matrizes L e U tais que $A = LU$ }

Supondo um *flop* a unidade de operação de ponto flutuante que corresponde a um produto (ou divisão) acompanhado de uma adição (ou subtração), o custo aritmético do Algoritmo 4 é a soma do número de *flops* para anular os elementos das $n-1$ colunas: a primeira linha da matriz fica inalterada e na obtenção dos zeros da primeira coluna realizamos $n(n-1)$ *flops* e, na segunda coluna, $(n-1)(n-2)$ *flops* e, assim por diante, resultando no seguinte somatório,

$$\sum_{k=1}^{n-1}(n-k+1)(n-k) = \sum_{k=1}^{n-1}(n^2+n-2kn+k^2-k) = \sum_{k=1}^{n-1}(n^2+n) - (2n+1)\sum_{k=1}^{n-1}k + \sum_{k=1}^{n-1}k^2 .$$

Utilizamos qualquer uma das fórmulas (1)-(4) para avaliar as somas $\sum_{k=1}^{n-1} k$ e $\sum_{k=1}^{n-1} k^2$, donde obtemos o número de *flops* requerido pelo Algoritmo 4:

$$\sum_{k=1}^{n-1}(n-k+1)(n-k) = \frac{n^3-n}{3}$$

Portanto, concluímos que o Algoritmo 4, desconsiderando o termo em n, requer aproximadamente $n^3/3$ operações de ponto flutuante. Do ponto de vista da teoria de complexidade de algoritmos, dizemos que o Algoritmo 4 é $O(n^3)$ *flops* (GÁCS, 1999).

Probabilidade de Bloqueio

A fórmula (8) é implementada para uma ampla faixa de valores, ou seja, para $N \in \{1, 2, 3, \cdots, 100\}$, e para um valor de intensidade de tráfego fixo, $\rho = 30$. Os resultados obtidos são comparados com os valores calculados pela aplicação direta da fórmula (5). O erro relativo percentual obtido comporta-se conforme mostra a Figura 1.

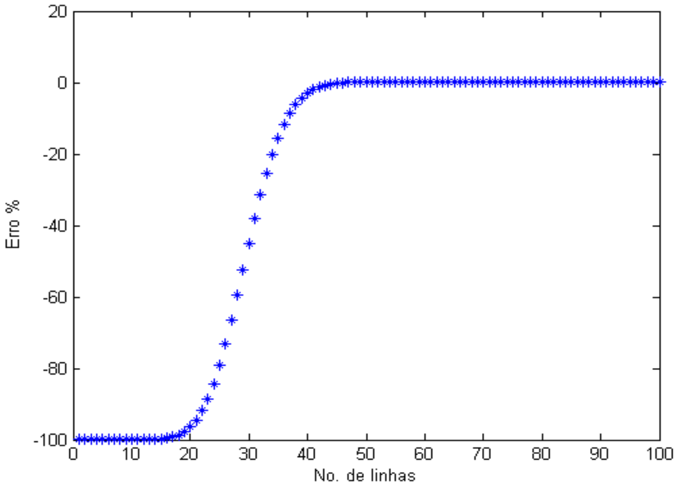

Figura 1: Erro Percentual Versus Número de Linhas (fórmula (8) e Erlang-B).

Notamos na figura 1 que, para $N > 40$, a fórmula (8) resulta em valores com excelente precisão. Infelizmente, para N grande e intensidade de tráfego elevada, a fórmula (8) apresenta erros inadmissíveis. A fórmula (8), que utiliza logaritmos, é aparentemente robusta.

A Tabela 1 mostra que, para valores de N em que a expressão original falha por ocorrência de *overflow*, a solução proposta avalia a probabilidade sem dificuldades.

O método proposto por Qiao, quando comparado à solução proposta, mostra de fato sua robustez, conforme atestam os resultados da Tabela 2.

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

Tabela 1: Valores de P_b Obtidos Através das Fórmulas (5) e (8) para Diferentes Quantidades de Linhas

Dados de entrada (r, N)	Probabilidades de bloqueio		Erro %
	Fórmula (5)	Fórmula (8)	
(5,12)	0,00344118753262	0,00345816662904	0,4934
(100,150)	6,511168497671872 $\times 10^{-7}$	6,514778779791440 $\times 10^{-7}$	0,0554
(100,156)	NaN	4,981973463434955 $\times 10^{-8}$	NaN
(200,171)	NaN	0,00333938442225	NaN
(200,1.000)	NaN	0,0000000000000	NaN

Tabela 2: Comparação dos resultados obtidos pelo algoritmo de Qiao e pela fórmula (8).

Dados de entrada (r, N)	Probabilidades de bloqueio		Erro %
	Qiao	Fórmula (8)	
(5,12)	0,00344118753262	0,00345816662904	0,4934
(100,150)	6,511168497671872 $\times 10^{-7}$	6,514778779791440 $\times 10^{-7}$	0,0554
(100,156)	4,979313290003586 $\times 10^{-8}$	4,981973463434955 $\times 10^{-8}$	0.0534
(200,171)	0,16703487737043	0,00333938442225	$-98,0000$
(200,1.000)	0,0000000000000	0,0000000000000	NaN

CONCLUSÃO

Neste trabalho foram analisados dois problemas de grande interesse em áreas aplicadas: a obtenção de fórmulas para cálculo de séries finitas de potências de inteiros; e o modelo de Erlang para dimensionamento de centrais telefônicas com enfoque no processamento numérico.

Com a intenção de despertar o interesse dos jovens estudantes para a Matemática Discreta e suas áreas correlatas, como Pesquisa Operacional e Complexidade de Algoritmos, escrevemos este artigo. Na atualidade, como estão disponíveis diversos sistemas computacionais que executam com extraordinária facilidade cálculos científicos, corremos o risco de ter a falsa impressão de que todos os problemas estão solucionados. Todavia, o uso desses sistemas requer uma adequada orientação sobre suas limitações e possibilidades, como aspectos de concepção de suas rotinas, quais teorias foram utilizadas para tal e a precisão dos resultados fornecidos. O objetivo principal é motivar os estudantes para não abrirem mão de se aprimorar quanto ao domínio da Matemática. Em particular, a Matemática Computacional.

A principal contribuição desse artigo é a proposta de uma nova fórmula geral para o cálculo de séries de potências de inteiros, baseada no Teorema das Colunas do Triângulo de Pascal. A apresentação e a discussão de diferentes expressões que conduzem a resultados idênticos para uma série finita de potência para um dado expoente inteiro é também uma relevante contribuição do ponto de vista didático. O emprego de uma ou de outra expressão entre as apresentadas, incluindo as que existem na literatura, é uma escolha que deve ser ditada pela eficiência e a necessidade inerente a cada situação. O tratamento da fórmula de Erlang-B e a apresentação do algoritmo de Qiao é também uma contribuição desse trabalho. Da comparação do método de Qiao com a aplicação direta da probabilidade de bloqueio e a fórmula sugerida em (8), concluímos que Qiao de fato propôs um algoritmo robusto e eficiente.

Como provável aplicação dos desenvolvimentos feitos neste trabalho, indicamos a elaboração de rotinas (toolbox) de cálculo simbólico (por exemplo, somas simbólicas) de séries e também a avaliação eficiente e robusta de expressões de modelos probabi-

estudos, Goiânia, v. 35, n. 2, p. 201-224, mar./abr. 2008.

lísticos como Erlang-B e outros no ambiente de domínio público Octave, que atualmente não dispõem desses recursos.

Referências

ERLANG, A.K. The theory of probabilities and telephone conversations. *Nyt Tidsskrift for Matematik* B, 20, 1909.

ERLANG A.K. Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges. *Elektrotkeknikeren*, 13, 1917.

GÁCS, P.; LOVÁSZ, L. *Lectures notes on complexity of algorithms*. [Boston]: Boston University and Yale University, 1999.

HANSELMAN, D.; LITTLEFIELD, B. Matlab 6: curso completo. [S.l.]: Makron Books, 2003.

IEEE. Institute of Electric and Electronic Engineering. IEEE Standard for Binary Floating-Point Arithmetic – IEEE Std. 754-1985. 1985.

INTEL CORPORATION. IA-32 Intel Architecture: Software Developer's Manual: 2006. V. 1: Basic Architecture.

KANEKO, M. The Akyiama-Tanigawa algorithm for Bernoulli numbers. *Journal of Integer Sequences*, Vol. 3, 7p. 2000.

KWANG-WU, C. Algorithms for Bernoulli numbers and Euler numbers. *Journal of Integer Sequences*, V. 4, 2001

MILLER, V. S. Finite sums and summation. In: ROSEN, K. H. *et alli*. (Eds.). *Handbook of Discrete and Combinatorial Mathematics*. New York: CRC Press, 2000. p. 161-170.

MORGADO, A. C. de O. et al. *Análise combinatória e probabilidade*. 6. Ed. [S.l]: Sociedade Brasileira de Matemática. 2004. (Coleção do professor de matemática).

QIAO, S.; QIAO, L. A robust and efficient algorithm for evaluating Erlang-B formula. Canada: McMaster University, 1998. Disponível em: <http://cas.mcmaster.ca/~qiao>. Acesso em: 2006).

ROSEN, K.H. Handbook of discrete and combinatorial mathematics. New York: CRC Press, 2000.

WILF, H. S. *Generatingfunctionlogy*. Philadelphia: Academic Press, 1994.

Abstract: this article concerns about the generalized summation formulae for evaluation $m-$*power series and the pitfalls presented by numerical processing during calculation of Erlang-B busy probability. For*

the first problem is proposed a new closed form expression for the sum of power of the first n *positive numbers and, for the second problem, is presented the Qiao's algorithm as a nice way to solver this problem.*

Key words: *Power series, Bernoulli numbers and Stirling numbers of second kind, Erlang model*

ANTÔNIO CÉSAR BALEEIRO ALVES
Doutor em Engenharia Elétrica na Universidade Estadual de Campinas. Mestre pela Universidade Federal de Uberlândia. Graduado em Engenharia Elétrica pela Universidade Federal de Goiás (UFG). Professor no Departamento de Computação da Universidade Católica de Goiás (UCG) e na Escola de Engenharia Elétrica e de Computação da UFG.

AUGUSTO SILVA
Especialista em Ciência da Computação pela UCG. Mestrando em Engenharia da Computação na Universidade Federal de Goiás. Graduado em Ciências Econômicas pela UCG. Professor no Departamento de Computação da UCG.

KÁTIA KELVIS CASSIANO
Acadêmica de Ciência da Computação na UCG.

MAIANA SANTOS LOPES
Acadêmica de Engenharia da Computação pela UCG.