Damer Tufaile

Engenheiro Civil pela Escola de Engenharia da Universidade Mackenzie

Teste de Primalidade

Resumo

Neste artigo apresentamos duas conjecturas baseadas na matriz de Bettencourt, nas SNPEFPs (Séries Numéricas Parcialmente Eliminadas de Falsos Primos) e em três artigos anteriores de nossa autoria publicados no Blog DNI, bem como dois exemplos em que se mostra o grau de dificuldade para determinar um primo consecutivo a um primo previamente conhecido, um algoritmo derivado do algoritmo hipercircular para testar a primalidade de números inteiros e um roteiro que sirva de base para a elaboração de um programa fonte deste algoritmo.

Introdução

O problema de testar a primalidade de um número inteiro, isto é, determinar se ele é primo ou não está relacionado com o problema da decomposição de números inteiros. Por exemplo, quando utilizamos o algoritmo hipercircular, que é determinístico (ver classificação dos algoritmos mais detalhada em **(1)** capítulo 3 página 30) para decompor um número inteiro qualquer, se o algoritmo mostra que ele não é passível de ser decomposto conclui-se daí que o número testado é primo. Então o problema de testar a primalidade de um inteiro, em última análise, pode ser resolvido através da utilização de algoritmos determinísticos. Na verdade o problema não está bem em sua resolução, mas sim na obtenção de meios ou argumentos que diminuam a quantidade de tempo gasto na sua resolução.

Chagas (2) em seu trabalho de graduação: ***Teste de primalidade: “Uma visão computacional”*** disserta sobre o estado da arte do algoritmo AKS considerado **top** de linha entre os algoritmos atualmente existentes e as melhorias devidas a vários pesquisadores que contribuíram para melhorar seu desempenho. Apesar dos esforços que ainda continuam a ser feitos para melhorá-lo, em termos de rapidez, ainda deixam a desejar.

Para uma boa compreensão do texto, caso o leitor não conheça, recomendamos a leitura dos artigos *“Algoritmo hiperbólico parcialmente eliminado de falsos primos”* **(3)** *“Algoritmo Circular”* **(4)** e *“Algoritmo hipercircular”* **(5)**.

1) Conjecturas baseadas na matriz de Bettencourt e SNPEFPs

Na tabela 1 apresentamos a matriz de **Bettencourt (6)** que permite visualizar algumas propriedades úteis a partir das quais é possível estabelecer conjecturas no que se referem à **geração** e **ordenação** de números primos.

1	7	13	19	25	31	37	43	49	55	61	67	73	79	85	91	
2	8	14	20	26	32	38	44	50	56	62	68	74	80	86	92	
3	9	15	21	27	33	39	45	51	57	63	69	75	84	87	93	
4	10	16	22	28	34	40	46	52	58	64	70	76	82	88	94	
5	11	17	23	29	35	41	47	53	59	65	71	77	83	89	95	
6	12	18	25	30	36	42	48	54	60	66	72	78	84	90	96	

Tabela 1

Dias (1) no capítulo 1 página 10 elenca algumas das conjecturas que ainda esperam por demonstrações entre elas as mais famosas se referem às conjecturas de Goldbach e Riemman e à existência de infinitos pares de primos gêmeos (primos cuja diferença é igual a 2). Apresentamos a seguir duas conjecturas baseadas na matriz de Bettencourt e SNPEFPs.

Uma **1ª conjectura** diz respeito à resposta da seguinte indagação: **existiria uma ou mais fórmulas passíveis de gerar e ordenar de forma cíclica, a exemplo das SNPEFPs, os números primos?**

Observando a 1ª e 2ª colunas da tabela 1 deparamos com uma formação **inicial única** que não mais se repete ao longo do conjunto infinito de números primos. De fato, a sequência de primos 2, 3, 5 e 7, espécie de **primos quadrigêmeos**, constitui uma **singularidade** não recorrente. Somente a tríade 3, 5 e 7 é responsável pela geração de 45,71% dos falsos primos ímpares, conforme descrito em **(6) tabela 5**. Esta singularidade é possível de ser visualizada na matriz de Bettencourt examinando as linhas e colunas: os primos e falsos primos ímpares desta matriz se alinham nas 1ª e 5ª linhas (os primos estão assinalados em negrito) e a partir da 2ª coluna só é possível a existência de primos gêmeos (primos cuja diferença é 2), isto porque entre os números localizados nas 1ª e 5ª colunas sempre haverá um múltiplo de 3 (ímpares na 3ª e pares na 6ª colunas) o que permite somente a possibilidade da existência de primos gêmeos ou primo seguido de falso primo ou ainda falsos primos gêmeos pares ou ímpares. Exemplos: 41 e 43 são primos ímpares gêmeos; 37 e 39 são primo e falso primo gêmeos, 50 e 52 são falsos primos gêmeos pares e 49 e 51 são falsos primos gêmeos ímpares. Uma eventual possível comprovação da existência desta série **exigiria a eliminação prévia dos primos 2 e 3 da serie numérica a ser determinada.**

Uma **segunda conjectura** que não exigiria a eliminação dos primos 3 e 5 da série numérica eventualmente a ser estabelecida partiria da seguinte indagação: **existiriam séries numéricas cíclicas que de forma similar, às SNPEFPs com terminações numéricas decenárias, seriam passíveis de gerar e ordenar números primos?**

Pesquisamos para as terminações numéricas decenárias 11 e 13 duas amostras contendo os primos com estas terminações no intervalo situado entre zero e um milhão para verificar a ocorrência de eventuais repetições cíclicas, porém, seja porque a amostra fosse relativamente limitada ou porque estas repetições, na verdade não ocorreriam, não foi detectado qualquer sinal da existência delas. Apesar disto, foi possível perceber um tipo de repetição aparentemente caótica onde a diferença entre primos consecutivos alternam entre pequenos e grandes incrementos situados entre 100 e 3500. Por exemplo, na tabela 2 apresentamos os primos com terminações numéricas decenárias em 11 e 13 situados aproximadamente no intervalo entre 840.000 e 850.000.

Primos com terminação numérica decenária em 11	Diferença entre primos consecutivos terminados em 11	Primos com terminação numérica decenária em 13	Diferença entre primos consecutivos terminados em 13
839911		839413	
840611	700	840713	1300
841411	800	841013	300
842111	700	841213	200
842311	200	841913	700
843211	900	842113	200
843811	600	842813	700
843911	100	843113	300
844111	200	843613	500
844511	400	844013	400
845111	600	844513	500
848611	3500	844913	400
849311	700	846113	1200
850211	900	846913	800
		847213	300
		847813	600
		848213	400
		848713	500
		850613	1900

Tabela 2

Ali é possível observar que nem sempre as quantidades de primos com terminações em 11 e 13 são iguais. No caso das terminações em 11 as diferenças entre primos consecutivos são relativamente menores quando confrontadas com a diferença de 3500 entre os primos 848611 e 845111, Isto significa que nos intervalos anteriores ocorre uma abundância de primos com as mais variadas terminações decenárias onde as diferenças entre eles são efetivamente menores, enquanto que no intervalo onde a diferença entre primos consecutivos é 3500 a quantidade de primos diminui e a diferença entre primos com terminações decenárias as mais variadas aumentam. Já no caso das terminações decenárias em 13 a diferença é mais homogênea, porém, com três intervalos acima de 1000 onde a quantidade de primos é mais rarefeita. Caso confirmado sua veracidade, esta conjectura implicaria em determinar, com eventual repetência, séries para cada uma das 40 terminações numéricas decenárias 1, 3, 7, 9, 11, 13,... 93, 97 e 99.

2) Grau de dificuldade na determinação de primos consecutivos a partir de um primo qualquer

Para **ilustrar** o grau de dificuldade desta determinação apresentamos na tabela 3 uma matriz simétrica criada a partir de uma linha e uma coluna de números primos das quais foram excluídos os primos 2, 3 e 5 (por já terem sido seus múltiplos previamente eliminados nos cálculos) e incluído o inteiro 1(hum) para permitir a existência do produto de um número primo qualquer pela unidade. Repare que a matriz é **simétrica** e representa em cada cruzamento de linhas e colunas o produto de **pares** de primos, portanto, a menos da 1ª coluna e 1ª linha que representam os produtos de primos pela unidade, as demais constitui o universo dos falsos primos compostos, cada um deles, somente por dois números primos (aqui denominados de falsos primos **primeiro**). Nela cada falso primo primeiro situado na diagonal principal corresponde ao quadrado de cada um dos primos que constituem a tabela 3. Por exemplo, no cruzamento da linha 17 com a coluna 41 encontramos o falso primo primeiro 697 correspondente ao produto $17 \cdot 41$ e no cruzamento da linha 67 com a coluna 67 encontramos o falso primo primeiro 4489 que corresponde ao quadrado perfeito de 67. Os falsos primos primeiro posicionados imediatamente à esquerda da diagonal principal (falsos primos de cada linha imediatamente anterior a um quadrado perfeito qualquer) apontam para a necessidade do surgimento de um novo primo. De fato, para perceber isto vamos imaginar um corte na tabela 3 na altura do primo 89 (ver linha e coluna mais grossas correspondentes a este corte). Observe que o falso primo primeiro 7387($83 \cdot 89$) que antecede ao falso primo primeiro 7821 ($89 \cdot 89$) corresponde ao máximo produto de primos primeiro que é possível obter inferior e anterior ao maior falso primo primeiro 7821. Se quiséssemos prosseguir na geração de produtos de pares de falsos primos primeiro seria necessário pesquisar qual é o primo

que sucede ao primo 89. Como o próximo primo é um número ímpar e sabendo que o distanciamento (diferença entre primos consecutivos) varia segundo uma progressão aritmética de razão 2: 2, 4, 6,... então o próximo primo poderá ser um dos seguintes números : 91, 93, 97, 99, 101e teríamos que testar sucessivamente a primalidade de cada um desses números até descobrir um novo número primo. Procedendo aos testes tem-se que 91 é divisível por 7, 93 é múltiplo de 3 e 97 que só é divisível por ele mesmo e por 1. Logo 97 seria o próximo primo. Se fizéssemos um corte na altura do primo 113 para obter o próximo primo (127) teríamos que testar a primalidade dos ímpares 117, 119, 121, 123 e 127 onde 117 é múltiplo de 3, 119 é múltiplo de 7, 121 é múltiplo de 11 e 127 que é primo, pois só é divisível por ele mesmo e por 1. A diferença entre primos consecutivos pode atingir valores bem maiores dos que foram apresentados nesses exemplos. Numa pesquisa parcial por nós realizada entre primos situados no intervalo zero e 300.000 detectou-se a existência de distanciamentos ou incrementos que atingiram valores de até 82 o que mostra, dependendo do primo inicialmente adotado, que a busca de um ou mais primos consecutivos é um problema que pode ter um alto grau de indeterminação, mas que é passível de ser resolvido, como vimos acima, pelo processo de tentativa e erro testando sucessivamente a primalidade de ímpares consecutivos que sucedem um dado número primo. Percebe-se ainda que a dificuldade de resolução deste problema se relaciona com a quantidade de falsos primos que se interpõem entre primos consecutivos. Visto por este ângulo, o mistério dos primos perde um pouco dos seus encantos, pois, fica mais fácil compreender a razão da aparente dificuldade em achar meios ou fórmulas que propiciem a geração e ordenação de números primos. O algoritmo hiperbólico parcialmente eliminado de falsos primos poderá ser de grande valia nos testes de primalidade de primos consecutivos, pois, dependendo do primo limite adotado, boa parte dos falsos primos que teriam de ser testados seriam previamente eliminados. Aliás, o algoritmo que apresentamos no próximo item faz uso parcial dessas eliminações.

3) Teste de primalidade baseado no algoritmo hipercircular

Independentemente das conjecturas e ideias apresentadas neste artigo pode-se estabelecer um algoritmo para testar a primalidade de um número inteiro baseado no algoritmo hipercircular. De fato, trata-se de uma simplificação que é possível quando o número a ser decomposto, por hipótese, for primo. Vamos exemplificar com o primo 71777. Como a decomposição deste número resulta no produto da unidade por ele mesmo ($1 \cdot 71777$) o Circulante que representa este produto tem o diâmetro $D=1+71777=71778$. Ao consultar a tabela 4 do artigo **algoritmo circular (4)** com terminação numérica decenária em 77, se se tratasse do caso geral de decomposição de um número composto, teríamos que utilizar duas séries uma com terminação diametral unitária 8 ímpar cada 20 e outra com terminação diametral 2 par cada 20. No exemplo em questão se o

número é primo a terminação diametral unitária tem que ser necessariamente 8 havendo, portanto, a necessidade de somente utilizar uma única série com terminação diametral unitária 8. Na tabela 4 apresentamos as terminações diametrais unitárias e decenárias bem como os incrementos em correspondência com as terminações numéricas decenárias a serem utilizados nos testes de primalidade onde é possível verificar que, independentemente das terminações numéricas decenárias, uma única série é utilizada num teste qualquer de primalidade. Observar que este teste somente responde se o número **é ou não é primo** sem se preocupar em decompô-lo em seus fatores caso ele não seja primo.

4) Elaboração de um roteiro para o software do teste de primalidade

Como o teste de primalidade em questão se baseia, em parte, no algoritmo hipercircular trata-se, portanto, de uma adaptação do roteiro elaborado para aquele algoritmo. A seguir elencamos os principais passos para sua consecução.

- 1) Criar uma tela na qual se possa digitar o dado de entrada que no caso é o número a ser testado em termos de primalidade. Só deverá ser aceito números inteiros. Os textos deverão ser apresentados em português e inglês
- 2) Após o teste deverá ser apresentado na tela os dados de saída com informações em português e inglês.
- 3) De posse do número $C=y^2$ a ser testado deverão ser elaborados os seguintes cálculos preliminares: verificar se o número C é igual ou divisível por cada um dos primos 2, 3, 5, 7 e 11 (primos inferiores ou igual ao primo limite que no caso do aplicativo DNI é 11. Se o primo limite adotado for maior que 11 esta verificação deve ser estendida até o primo limite adotado. Exemplo: se o primo limite escolhido for 31 estas verificações devem também incluir os primos 13, 17, 19, 23, 29 e 31. Caso qualquer destes primos for igual a C o software retorna na tela a mensagem: “**este número é primo**”. Caso contrário, se C for divisível por qualquer destes fatores o software deverá retornar na tela a mensagem: “**Este número não é primo**” e informa as quantidades de cálculo executadas pelos algoritmos, zera os arquivos e se posiciona para um novo cálculo.

Importante: Caso C digitado for igual a 1 (hum) retornar na tela a mensagem: “**Este número não é primo**”

Caso contrário prossegue-se nos cálculos passando para o item 4.

- 4) O algoritmo circular começa calculando o máximo valor que x (inteiro menor) pode alcançar (seu valor é dado por $x_{\max}=C^{1/2}$). Determina o diâmetro mínimo (seu valor é calculado em função de $x_{\max}$ e vale $D_{\min}=2*x_{\max}$). Determina o valor de x' que define os intervalos de cálculo onde os algoritmos circular e hipérbólico irão atuar que por sua vez é definido como $x'=x_{\max}/k$ onde k é uma constante maior que 1 (hum) é previamente definida. O valor de $D_{\min}$ é arredondado para cima. Este arredondamento é feito em função da terminação numérica decenária (dezena do número C a ser decomposto) e suas relações com as terminações diamétricas unitárias e decenárias devendo ser consultada a tabela 4 do presente artigo para se estabelecer o arredondamento necessário. O diâmetro mínimo é arredondado para o diâmetro de partida.

5) Os testes começam sempre pelo algoritmo circular utilizando a fórmula x ou $z = [(D+ou-(D^2-4y^2)^{1/2})] / 2$ e têm como condições de interrupção quando se obtiver x e z inteiros ou, caso contrário, quando x calculado for menor ou igual a x' . Caso ocorra a 1ª condição a tela do software retorna: “**Este número não é primo**”, bem como a quantidade de cálculo executada pelo algoritmo circular e o tempo gasto, após o que se zera os arquivos e se posiciona para um novo cálculo. Caso ocorra a 2ª condição de interrupção, isto é x menor que x' , então os cálculos passam a ser executados pelo algoritmo hiperbólico conforme descrito no item 6.

6) Caso não encontre x e z inteiros, diferentes de C e 1 , haverá um momento em que x calculado ficará menor que x' . Neste momento o algoritmo hiperbólico assume o comando dos cálculos com x , em princípio, variando agora de forma crescente de 13 a no máximo x' (Caso o primo limite for diferente de 11 , por exemplo, se o primo limite for igual a 31 o x de partida será 37). Para tanto, é utilizada a $SNPEFP_{11}$. Aqui, também duas condições de interrupção podem ocorrer: x e z inteiros são encontrados ou não. Se forem encontrados a tela do software deve retornar a mensagem: “**Este número não é primo**”, informa as quantidades de cálculo executadas pelos algoritmos, zera os arquivos e se posiciona para um novo cálculo. Quando x e z inteiros não são encontrados o limite de interrupção dos cálculos será quando x crescente for maior ou igual a x' . Repare que a variável x percorre todo o intervalo onde x poderia existir: variou de x_{max} a x' pelo algoritmo circular e de 13 a x' pelo algoritmo hiperbólico. Como x e z inteiros não foram encontrados a conclusão é, neste caso, a de que o número testado é primo. A tela do software deve retornar a mensagem: “**Este número é primo**”, e ainda informar as quantidades de cálculo executadas pelos algoritmos e os tempos gastos. Finalmente os arquivos são zerados e se posiciona para um novo cálculo.

TABELA DE TERMINAÇÕES DIAMETRAIS UNITÁRIAS E DECENÁRIA

Dezenas terminadas em	têm Terminações Unitárias dos Diâmetros em	e Terminações Decenárias dos Diâmetros em	Com Taxa de Incremento de
1	2	2	100
21	2	22	100
41	2	42	100
61	2	62	100
81	2	82	100
3, 23, 43, 63, 83	4	Par	20
7, 27, 47, 67, 87	8	Par	20
9	0	Im par	20
29	0	Im par	20
49	0	Im par	20
69	0	Im par	20
89	0	im par	20
11	2	12	100
31	2	32	100
51	2	52	100
71	2	72	100
91	2	92	100
13, 33, 53, 73, 93	4	Im par	20
17, 37, 57, 77, 97	8	Im par	20
19	0	Par	20
39	0	Par	20
59	0	Par	20
79	0	Par	20
99	0	Par	20

Tabela 4

5) Bibliografia

- (1) **Dias, Cristina Helena Bovo Batista** Números Primos e Divisibilidade: Estudo de Propriedades.
- (2) **Chagas, Amirton Bezerra** Testes de Primalidade: Uma Visão Computacional.
- (3) **Tufaile, Damer** Algoritmo hiperbólico parcialmente eliminado de falsos primos – blog DNI: www.damer.tufaile.nom.br
- (4) **Tufaile, Damer** Algoritmo Circular – blog DNI: www.damer.tufaile.nom.br
- (5) **Tufaile, Damer** Algoritmo hipercircular – blog DNI: www.damer.tufaile.nom.br
- (6) **Tufaile, Damer** A ciência de eliminar falsos primos – blog DNI: www.damer.tufaile.nom.br
- (6) **Bettencourt, Emanuel** : Conjectura de Goldbach – Como encontrar números primos. <http://aconjecturadegoldbach.blogspot.com.br/2009/02/licao-n-2.html>