Ciencias Matemáticas, Vol. 33, No. 1, Pag. 56-61, 2019

Algoritmo determinista de factorización
Deterministic factorization algoritms

Alexi Massó Muñoz[1], Magdiel Vicet Morejón[2], Redelio Hernández Rubio[3]

Resumen En el presente artículo se incluyen los resultados de una incipiente investigación encaminada a obtener un algoritmo matemático-computacional que permita la factorización de manera determinista de un número. En el mismo se incluyen los puntos de vista y elementos que pudieran permitir la optimización del algoritmo así como particularidades que adaptan el algoritmo para la factorización del número n del RSA..

Abstract On this article we present the results of the early investigation to get an matematical and computacional determinist factoring algorithm. We included to some view points and elements to allow the optimization of the algorithm and explain some particualarities that adjust the method to factorizate the RSA n number.

Palabras Clave
Factorización, Método, Algoritmo.

[1] *Instituto de Criptografía, universidad de la Habana, La Habana, Cuba, alexi.masso@matcom.uh.cu´*
[2] *Instituto de Criptografía, universidad de la Habana, La Habana, Cuba, magdiel.vicet@matcom.uh.cu*
[3] *Cátedra de Informática, Instituto Técnico Militar, La Habana, Cuba*

1. Introducción

El uso del algoritmo RSA es, todavía, bastante alto. Incluso cuando se saben las posibilidades de los algoritmos de factorización existentes, se sigue usando este método asimétrico en aplicaciones donde no se requiere una alta potencia criptográfica. Otra variante de uso de este algoritmo lo podemos encontrar también mediante la utilización de llaves de gran cantidad de bits (8192 bits o superior).
Es por ello que la factorización de números grandes sigue siendo una necesidad latente en el mundo de la Criptografía. No existen en la actualidad algoritmos de factorización deterministas que, aplicados a números grandes, sean factibles desde el punto de vista del tiempo de cálculo necesario para ejecutarlos en computadoras clásicas.
En el caso de la factorización del número n del RSA existen algoritmos con un buen desempeño como es el caso de la criba cuadrática y la criba del campo numérico. Estos algoritmos permiten realizar la factorización incluso mediante el uso del paralelismo para aumentar su eficiencia. Sin embargo cuando son usados para factorizar los números grandes que se usan hoy en dia en el RSA todavía son prohibitivos. Muestra de lo anterior se puede obtener si se analizan los resultados del *RSA Factoring Challenge*. Para la factorización del RSA-768 se han utilizado más de dos a?os de cómputo en varios cientos de CPUs lo que equivale a más de 1500 años de un solo procesador.
En este artículo se presentan los primeros resultados de una incipiente investigación encaminada a obtener un algoritmo que permita la factorización de manera determinista de un número y un conjunto de aspectos para su posible optimización (aunque la optimización no es objetivo del presente trabajo).

2. Multiplicación de números enteros

Sean dos números del mismo orden

$$a = (a_0 ... a_n) \;\; y \;\; b = (b_0 ... b_n) \tag{1}$$

El producto de ambos se calcula por:

$$c = \sum_{j}^{n} \sum_{i}^{n} a_i b_j 10^{i+j} \tag{2}$$

Donde se ha usado la base 10 para fijar ideas.
El orden de c es al menos de $2n$ o en todo caso $2n+1$.
Los productos parciales que dan lugar a c puede ponerse como:

$$\begin{aligned}
c'_0 &= a_0 b_0 \\
c'_0 &= a_0 b_0 \\
c'_1 &= a_0 b_1 + a_1 b_0 \\
c'_2 &= a_0 b_2 + a_1 b_1 + a_2 b_0 \\
&? \\
c'_k &= a_0 b_k + a_1 b_{(k-1)} + ... + a_k b_0 \\
&? \\
c'_n &= a_0 b_n + a_1 b_{(n-1)} + ... + a_n b_0 \\
c'_{(n+1)} &= a_1 b_n + a_2 b_{(n-1)} + ... + a_n b_1 \\
c'_{(2n-1)} &= a_(n-1) b_n + a_n b_{(n-1)} \\
c'_{2n} z &= a_n b_n
\end{aligned} \tag{3}$$

Es decir la cantidad de sumandos crece hasta c'_n y luego decrece hasta el último valor, el producto total se expresa conforme a:

$$c = \sum_{i=0}^{2n} c'_i 10^i \tag{4}$$

Está claro que los c'_i son los números que existieron en una multiplicación por columnas de no realizarse las operaciones de acarreo correspondientes.
Por ejemplo en la multiplicación:

$$\begin{array}{cccc} & & 47 & \\ & & *64 & \\ \hline & & 16 & 28 \\ 28 & 42 & & \\ \hline 28 & 58 & 28 & \end{array}$$

donde:

$$\begin{array}{l} c'_0 = 28 \\ c'_1 = 58 \\ c'_2 = 28 \end{array}$$

A partir de los c'_i se puede construir el número c representado en base 10 o sea como:

$$c = \sum_{i=0}^{n-1} c_i 10^i \tag{5}$$

Entonces:

$$\begin{array}{rcl} c_0 & = & \hat{R}(c'_0) \\ c_1 & = & \hat{R}(c'_1 + \hat{D}(c'_0)) \\ c_2 & = & \hat{R}(c'_2 + \hat{D}(c'_1 + \hat{D}(c'_0))) \end{array} \tag{6}$$

Las propiedades de los operadores *R* barra y *D* barra se exponen en [1]. Esto nos lleva a que:

$$c_k = \hat{R}(c'_k + \hat{D}(c'_{k-1} + \hat{D}(c'_{k-2} + \hat{D}(c'_{k-3} + \cdots)))) \tag{7}$$

La fórmula (5) nos dice que en cualquier dígito c_k influye en primer lugar el producto correspondiente a ese orden c'_k y luego los restantes productos pero atenuados por el operador $\hat{D}$ que divide el resultado entre 10, lo que implica que c'_0 influye con $\hat{D}(c'_0)$ en c'_1 y con $\hat{D}^2(c'_0)$ en c'_2. Como los c'_i son productos de números de 1 dígito su mayor valor es 81 las influencias del valor de c'_i sobre los c_k posteriores se va atenuando fuertemente, en el caso analizado realmente:

$$\hat{D}^2(c'_0) = 0 \tag{8}$$

pero puede demostrarse en el caso extremo c'_0 influye en c_2 a lo sumo en una unidad.
Al realizarse una multiplicación, las operaciones en las cifras inferiores influyen de manera decreciente en las superiores.
Veamos c?mo se cumple la relaci?n (5)
Sean la multiplicación:

$$\begin{array}{c} 934 \\ *956 \\ \hline 892904 \end{array}$$

Donde:

$$\begin{array}{ccccc} & & 934 & & \\ & & *956 & & \\ \hline & & 54 & 18 & 24 \\ & 45 & 15 & 20 & \\ 81 & 27 & 36 & & \\ \hline 81 & 72 & 105 & 38 & 24 \end{array}$$

Siendo:

$$c'_0 = 24,\ c'_1 = 38,\ c'_2 = 105,\ c'_3 = 72\ y\ c'_4 = 81$$

Por lo que:

$$\begin{array}{rcl} c_0 & = & \hat{R}(24) = 4 \\ c_1 & = & \hat{R}(38 + \hat{D}(24)) = 0 \\ c_2 & = & \hat{R}(105 + \hat{D}(38 + \hat{D}(24))) = 9 \\ c_3 & = & \hat{R}(72 + \hat{D}(105 + \hat{D}(38 + \hat{D}(24)))) = 2 \\ c_4 & = & \hat{R}(81 + \hat{D}(72 + \hat{D}(105 + \hat{D}(38 + \hat{D}(24))))) = 9 \\ c_5 & = & \hat{D}(89) = 8 \end{array}$$

Obsérvese que $\hat{D}(24) \equiv \hat{D}(c'_0)$ influyó en c_1 con 2 unidades y en c_2 con una unidad no teniendo peso ni en c_3, c_4 y c_5, por su parte c_1 influyó en c_2 pero no en c_3, c_4 y c_5.
De lo expuesto se desprende que las combinaciones multiplicatorias aumentan de c'_0 a c'_n y disminuyen de c'_0 a $c'_2 n$ lo que está en consecuencia con la estructura escalonada en la multiplicación.

3. Proceso de factorización

Ahora pongamos los coeficientes c'_i dados por (3) de forma escalonada como:

$$\begin{array}{rccccc} & \cdots & [a_0b_2] & [a_0b_1] & [a_0b_0] \\ & \cdots & [a_1b_1] & [a_1b_0] & \\ & \cdots & [a_2b_0] & & \\ \hline c= & \cdots & c'_2*10^2 & c'_1*10^1 & c'_0*10^0 \end{array} \tag{9}$$

Que se corresponde con el número:

$$c = \cdots c_2c_1c_0 \tag{10}$$

Se sabe que $c_0 = \hat{R}(c'_0)$. Si se conocen a_0 y b_0 entonces al restar c'_0 a c y dividir entre 10 se obtiene:

$$d = \frac{(c-c'_0)}{10} = \cdots c'_3*10^2 + c'_2*10^1 + c'_1*10^0 \tag{11}$$

Que se corresponde con el número escalonado:

$$\begin{array}{rccccc} & \cdots & [a_0b_3] & [a_0b_2] & [a_0b_1] \\ & \cdots & [a_1b_2] & [a_1b_1] & [a_1b_0] \\ & \cdots & [a_2b_1] & [a_2b_0] & \\ & \cdots & [a_3b_0] & & \\ \hline c= & \cdots & c'_3*10^2 & c'_2*10^1 & c'_1*10^0 \end{array} \tag{12}$$

Cuya representación se obtiene como:

$$d = \frac{(\cdots c_3c_2c_1c_0 - c'_0)}{10} = \cdots d_3d_2d_1d_0 \tag{13}$$

Donde ahora:

$$d_0 = \hat{R}(a_0b_1 + a_1b_0) = \hat{R}(\hat{R}(a_0b_1) + \hat{R}(a_1b_0)) \tag{14}$$

Si se conoce a_1, b_1 puede repetirse el proceso restando las combinaciones

$$r = a_0b_1 + a_1b_0 + a_1b_1*10 \qquad \text{o sea } e = \frac{d-r}{10} \tag{15}$$

Obteniéndose:

$$\begin{array}{rccccc} & \cdots & [a_0b_3] & [a_0b_2] & \\ & \cdots & [a_1b_2] & & \\ & \cdots & [a_2b_1] & [a_2b_0] & \\ & \cdots & [a_3b_0] & & \\ \hline e= & \cdots & e'_1*10^1 & e'_0*10^0 & = \cdots e_1e_0 \end{array} \tag{16}$$

Donde de nuevo:

$$e_0 = \hat{R}(a_0b_2 + a_2b_0) = \hat{R}(\hat{R}(a_0b_2) + \hat{R}(a_2b_0)) \tag{17}$$

El proceso puede reiterarse y siempre lleva a la ecuación:

$$d_{k0} = \hat{R}(\hat{R}(a_0b_k) + \hat{R}(a_kb_0)) \tag{18}$$

Donde a_0, b_0 y d_{k0} son conocidos:
Este proceso puede servir para factorizar un número si se pueden resolver las ecuaciones:

$$\begin{aligned} \hat{R}(a_0b_0) &= d_{00} = c_0 \\ \hat{R}(\hat{R}(a_0b_1) + \hat{R}(a_1b_0)) &= d_{10} = d_0 \\ \hat{R}(\hat{R}(a_0b_2) + \hat{R}(a_2b_0)) &= d_{10} = e_0 \end{aligned} \tag{19}$$

y así sucesivamente.
Las ecuaciones anteriores pueden desdoblarse para hacerlas más simple, en efecto dado que el valor máximo de $\hat{R}(a_0b_k)$ y $\hat{R}(a_kb_0)$ es el valor 9 su suma vale a lo sumo 18 por tanto las ecuaciones pueden plantearse como:

$$\begin{aligned} \hat{R}(a_0b_0) &= d_{00} \\ \hat{R}(a_0b_1) + \hat{R}(a_1b_0) &= d_{10}\ o \\ \hat{R}(a_0b_1) + \hat{R}(a_1b_0) &= 10 + d_{10}\ \textit{siempre que}\ d_{10} < 8 \\ \hat{R}(a_0b_2) + \hat{R}(a_2b_0) &= d_{20}\ o \\ \hat{R}(a_0b_2) + \hat{R}(a_2b_0) &= 10 + d_{20}\ \textit{siempre que}\ d_{20} < 8 \end{aligned} \tag{20}$$

Y de igual forma para los restantes coeficientes:
Obsérvese que las combinaciones de coeficiente a restar pueden realizarse simultáneamente dirigida a afectar la columna que se quiere dejar con dos términos, sin embargo, al usar todas las combinaciones posibles puede tenerse como criterio de parada cuando el número resultante de la resta sea menor o igual a cero.
Dado que solo son de interés los números impares donde no existen factores entre 2 y 9 las terminaciones de tales números son 1, 3, 7 y 9 y la tabla de multiplicar nos dice que en la ecuación:

$$\hat{R}(a_0b_0) = d_{00}\ d_{00} = 1, 3, 7\ o\ 9 \tag{21}$$

Existen 4 soluciones posibles en cada caso:
Tomando en cuenta lo expuesto puede enunciarse un método de factorización, en efecto por cada uno de los 4 pares a_0, b_0 que da la primera ecuación se dan valores a a_1 desde 0 a 9 y se ve si:

$$x = d_{10} - \hat{R}(a_1 b_0) \; o \; x' = 10 + d_{10} - \hat{R}(a_1 b_0) \tag{22}$$

son solución de $\hat{R}(a_0 b_1)$.
Obteniéndose los pares a_1, b_1 que satisfacen esta condición. Creándose un árbol de solución de la siguiente forma:

A partir del primer nivel hacia abajo el número de hijos posibles no se conoce, los caminos del grafo dan los posibles factores de los números $(a_n \cdots a_2 a_1 a_0)$, $(b_n \cdots b_2 b_1 b_0)$.
Este algoritmo así presentado, si bien es cierto que calcula de manera determinista todos los factores del número, incluyendo los primos, lo hace a un costo elevado, pues puede demostrarse que el par de ecuaciones (22):

$$\begin{aligned} \hat{R}(a_i b_l) + \hat{R}(a_m b_j) &= d_k \; o \\ \hat{R}(a_i b_l) + \hat{R}(a_m b_j) &= 10 + d_k \end{aligned} \tag{23}$$

Donde $i + l = j + m = k$
Posee 10 soluciones posibles para cada par a_i, b_j conocidos, por tanto, la cantidad de caminos del grafo aumentan de forma exponencial, de manera que en el peor caso, y después de analizada la cifra que posee el par (a_k, b_k) habrían 10^{k-1} caminos posibles. No obstante, vale la pena analizar algunas de las particularidades del método con vista a su posible optimización.
La factorización en sí posee algunas características numéricas que deben analizarse.
Los criterios de divisibilidad para factores de una cifra son simples y viables en todos los casos o bien por inspección de las cifras o uso de la aritmética modular. Por tanto, un número cualquiera puede reducirse hasta no tener factores de un dígito, en todos los casos las terminaciones posibles son:

$$a_0, b_0 = 1, 3, 7, 9 \tag{24}$$

Siempre que la suma de signos alternos de los dígitos sea diferente de cero para evitar la divisibilidad por once. Cada una de estas terminaciones tiene a su vez un conjunto de factores que la producen que son:

$$\begin{aligned} 1 \rightarrow a_0 = 1 b_0 = 1; a_0 = 9 b_0 = 9; a_0 = 7 b_0 = 3 \\ 3 \rightarrow a_0 = 1 b_0 = 3; a_0 = 9 b_0 = 7 \\ 7 \rightarrow a_0 = 7 b_0 = 1; a_0 = 9 b_0 = 3 \\ 9 \rightarrow a_0 = 9 b_0 = 1; a_0 = 7 b_0 = 3 \end{aligned} \tag{25}$$

Si se observan las estructuras (9), (12) y (16) se ve que la formula (23) queda reducida en todos los casos a:

$$\begin{aligned} \hat{R}(a_0 b_l) + \hat{R}(a_l b_0) &= d_l \; o \\ \hat{R}(a_0 b_l) + \hat{R}(a_l b_0) &= 10 + d_l \end{aligned} \tag{26}$$

Si $a_0 = b_0$ se obtienen las ecuaciones:

$$\begin{aligned} \hat{R}(a_0 b_l) + \hat{R}(a_l a_0) &= d_l \; o \\ \hat{R}(a_0 b_l) + \hat{R}(a_l a_0) &= 10 + d_l \end{aligned} \tag{27}$$

En ambos casos presentan las 10 soluciones posibles para los pares $(a_l b_l)$ no obstante hay posibilidad de reducción para casos particulares ya que cada columna de dígitos multiplicados tiene como resultado:

$$\sum_{k=0}^{l} a_k b_{l-k} = c'_l \tag{28}$$

Este resultado se corresponde con el dígito:

$$\hat{R}(\sum_{k=0}^{l} \hat{R}(a_k b_{l-k})) = d'_l \tag{29}$$

Que puede ponerse como:

$$\hat{R}(\hat{R}(a_0 b_l) + \hat{R}(\sum_{k=0}^{l} (a_k b_{l-k})) + \hat{R}(a_l b_0)) = d'_l \tag{30}$$

Supongamos que la columna $l-1$ ya fue eliminada en el proceso de resta y solo falta por restar los factores de a_{l-1} y b_{l-1} o sea:

$$a_{l-1} \sum_{k=1}^{l-1} b_k^{k-1} + b_{l-1} \sum_{k=1}^{l-1} a_k^{k-1} - a_{l-1} b_{l-1} 10^{l-1} \tag{31}$$

Entonces el término dado por (30) tiene cuatro sumandos:

$$\hat{R}(\hat{R}(a_0 b_l) + \hat{R}(a_1 b_{l-1}) + \hat{R}(a_{l-1} b_1) + \hat{R}(a_l b_0)) = d'_l \tag{32}$$

Que puede ponerse como:

$$\hat{R}(\hat{R}(a_0 b_l) + \hat{R}(a_1 b_{l-1} + a_{l-1} b_1) + \hat{R}(a_l b_0)) = d'_l \tag{33}$$

Que se reduce en:

$$\hat{R}(\hat{R}(a_0 b_l + a_l b_0) + \hat{R}(a_1 b_{l-1} + a_{l-1} b_1)) = d'_l \tag{34}$$

Las dos ecuaciones que se derivan de (34) son:

$$\begin{aligned} \hat{R}(a_0 b_l + a_l b_0) &= d'_l - \hat{R}(a_1 b_{l-1} + a_{l-1} b_1) \\ \hat{R}(a_0 b_l + a_l b_0) &= d'_l + 10 - \hat{R}(a_1 b_{l-1} + a_{l-1} b_1) \end{aligned} \tag{35}$$

La primera relación de (35) no es posible cuando:

$$\hat{R}(a_1 b_{l-1} + a_{l-1} b_1) > d'_l \tag{36}$$

Por tanto, sólo es válida la segunda.
Vale aclarar que en el caso de $l = 2$ sólo se considera el término dado por:

$$\hat{R}(a_1 b_1)$$

Conforme a la regla de resta dada por (30).
Un ejemplo de este caso es el siguiente:
Factoricemos el número: 114356
Como el número tiene 6 cifras a lo sumo tiene 2 factores de 3 cifras. Para abreviar el proceso no se analizarán todas las posibilidades.
Suponemos $a_0 = 3$ y $b_0 = 2$
Al restar $a_0 b_0$ dividir en 10, tenemos:11435

$$\text{como } \hat{R}(3b_1) + \hat{R}(2a_1) = 5 \text{ ó } \hat{R}(3b_1) + \hat{R}(2a_1) = 15$$

Hacemos $a_1 = 5$ y $b_1 = 5$ quedando después de restar $a_1 b_0 + a_0 b_1$ y dividir en 10: 1141.
Aquí tenemos en cuenta que:

$$\hat{R}(a_1 b_1) = 5 > 1 \tag{37}$$

Por lo tanto, se escoge la ecuaciń:

$$\hat{R}(a_0 b_2) + \hat{R}(a_2 b_0) = 1 + 10 - \hat{R}(a_1 b_1) \tag{38}$$

Quedando

$$\hat{R}(3b_2) + \hat{R}(2a_2) = 6 \tag{39}$$

En efecto al restar $a_1 b_1$ queda:

$$\begin{array}{r} 1141 \\ -25 \\ \hline 1116 \end{array}$$

La ecuación (38) brinda 8 pares $(a_2 b_2)$ posibles que son:

$$(0,7)(1,8)(2,4)(0,3)(5,2)(6,8)(4,7)(0,8)$$

De los cuales se prueba el valor del par $(a_2 b_2)$ que completa el proceso es $a_2 = 2$ $b_2 = 4$.
Por tanto son factores 253 y 452.
Al margen de lo que se quería mostrar, el número del ejemplo es par por tanto en la práctica en vez de trabajar con el número de 6 cifras 114356 este puede simplificarse, como se observa cumple el criterio de divisibilidad por 4 ya que:

$$2c_1 + c_0 = 2 * 5 + 6 = 16$$

Es múltiplo de 16 además el número también es múltiplo de 11 ya que:

$$\sum_{i=0}^{s} (-1)^i c_i = 6 - 5 + 3 - 4 + 1 - 1 = 0$$

Por tanto, se requería hallar los factores de:

$$\frac{114356}{44} = 2599$$

4. Conclusiones

En efecto el algoritmo descrito permite la factorización de un número cualquiera. Este método puede ser optimizado para tratar de disminuir al máximo la cantidad de pares candidatos que se generan en la búsqueda de los factores.
En aras de aumentar la velocidad del método se pueden sustituir algunos de los pasos, por otros equivalentes, pero que sean computacionalmente más eficientes. Actualmente se trabaja en la optimización del método expuesto.

5. Referencias

[1] M. Vicet, R. Hernández y A. Massó. 2018. Sistemas numéricos embebidos

6. Bibliografía

- **Gvozdanovic, Jadranka. Numeral Types and Changes Worldwide (1999).**
- **Walter Mora F.2010. Introducción a la Teoría de Números.**
- **Victor Shoup. 2008. A Computational Introduction to Number Theory and Algebra, Version 2.**
- **R. Tijdeman December. 2007. Combinatorial and Analytic Number Theory.**

- **Martyn R. Dixon. 2010. Algebra and Number Theory.**
- **Schneier, B. (1996). Applied Cryptography (2da Edición). John Wiley and Sons.**
- **A. Menezes, P. van Oorschot, yS. Vanstone, 1996. Handbook of Applied Cryptography**