

LA FATTORIZZAZIONE VELOCE

E IL PROBLEMA $P = NP$ (con accenno alla RSA e al logaritmo discreto)

Gruppo Eratostene

Sommario

In questo lavoro sulla fattorizzazione si ripropone il polinomio $N + d^2 = s^2$ per una fattorizzazione veloce, o almeno più veloce di quella tradizionale, nel caso la semidifferenza $d = (q - p)/2$ sia piccola, con diversi esempi pratici. Infine si accenna al logaritmo discreto, e ad una sia pur lontana ma teoricamente possibile ricerca di un algoritmo efficiente per il suo calcolo, possibilmente ispirato dal nostro suddetto polinomio, così come alcuni attuali algoritmi sono ispirati agli attuali algoritmi per la fattorizzazione degli interi; mentre però questi ultimi

riguardano la crittografia RSA, gli algoritmi per il calcolo del logaritmo discreto riguardano la crittografia basata sulle curve ellittiche (EEC), usata nei piccoli dispositivi elettronici (telefonini, palmari, ecc) che richiedono, com'è noto, chiavi di cifratura minori rispetto alla crittografia RSA. E' una sfida difficile per connettere i due sistemi di cifratura tramite algoritmi di fattorizzazione e di calcolo del logaritmo discreto, ma non si sa mai. qualcuno potrebbe alla fine riuscirci.

Su una fattorizzazione più veloce di quella tradizionale ($N/p = q$) sono già noti diversi algoritmi e relativi software, per esempio il PARI/GP in grado di trattare numeri enormi; o anche l'algoritmo di Shor, che però richiede l'uso di computer quantistici ancora in fase iniziale di

sperimentazione. In questo lavoro riproponiamo il nostro polinomio:

$$N + d^2 = s^2 \quad (1)$$

con $N = p * q$, $d = (q-p)/2$ ed $s = (p + q)/2$, e quindi con

N possibile numero RSA = prodotto di p per q , d

semidifferenza d ed s semisomma tra p e q . La

semisomma richiama la nota ex congettura di Goldbach

secondo la quale un numero pari N è somma di due numeri

primi p e q , ed è stata trattata nei nostri precedenti lavori;

qui essa interviene come semisomma dei due primi p e q

connessa al loro prodotto N tramite il suddetto polinomio

(1). Tale polinomio è, come vedremo, molto veloce ed

efficiente nel fattorizzare N quando p e q sono due numeri

primi molto vicini, e quindi con valori di d (semidifferenza)

molto bassi (tendenti a zero), e quindi con rapporto q/p

molto vicini ad 1. Per i numeri RSA il rapporto tra q e p è

di solito compreso tra 1 e 2, essendo i due numeri primi comparabili per grandezza (numero delle cifre di cui sono composti, per esempio di 600 cifre o anche più attualmente usati per renderli più sicuri e difficili da individuare, a causa dei lunghissimi tempi di fattorizzazione classica. Per p e q molto distanti, invece, conviene la fattorizzazione classica, che inizia con i numeri primi più piccoli, che però non sono usati nel sistema RSA.

Per fattorizzare questi ultimi, è relativamente più vantaggioso il nostro polinomio (ed eventuali futuri miglioramenti dell'algoritmo di Fermat al quale si collega), tanto più quanto i numeri primi p e q sono vicini tra loro, e quindi con piccole semidifferenze.

Circa il problema del millennio $P = NP$ (del quale la fattorizzazione è uno tra il migliaio di casi simili), esso è vero se la fattorizzazione fosse un problema di tipo NP –

completo (questo però non si sa ancora bene), e se anche la RH fosse vera, poiché in questo caso esisterebbe un polinomio per la fattorizzazione veramente veloce. Tale polinomio potrebbe anche essere il nostro oppure un suo ancora più efficiente derivato), ancora però da scoprire (ma il nostro potrebbe già spianare la strada) oppure ancora un altro polinomio su basi diverse, ma anch'esso ancora tutto da scoprire. Il vero problema della fattorizzazione dei numeri RSA, rimane ancora la grandezza dei medesimi , di centinaia di cifre decimali e almeno 2048 bit, considerati sicuri). Con il nostro polinomio la difficoltà passa però dal numero di cifre di N (chiave pubblica) e dei due numeri primi p e q (chiave privata), al numero di cifre della semidifferenza, molto più piccolo, e quindi con minore tempo di calcolo (Rif. e citazioni nel Mathbuilding dell'Ing.

Rosario Turco, si veda la sezione Link del nostro sito).

Il caso limite (fattorizzazione istantanea) sono i numeri primi gemelli (sconsigliabilissimi quindi per formare numeri RSA $N = p * q$ con p e q primi gemelli), poiché la loro differenza è, com'è noto, $D = q - p = 2$ e quindi la loro semidifferenza è 1. Quindi il nostro polinomio si riduce ora

$$a \ N + d^2 = N + 1 = s^2, \text{ da cui ricavare subito}$$

$$s = \sqrt{N + 1} \text{ e quindi } p = s - 1, \text{ e } q = s + 1 :$$

la fattorizzazione è fatta con un solo tentativo,

$$d^2 = 1^2 = 1. \text{ Per semidifferenze più grandi,}$$

occorrono d tentativi, aggiungendo ad N tutti i quadrati successivi di d :

1, 4, 9, 16, 25, ... , finché non si trova il valore esatto di d , tale che $N + d^2$ sia il quadrato perfetto di s .

Alcuni esempi pratici:

a) numeri gemelli, per es. $p = 29$ e $q = 31$.

$$N = 29 * 31 = 899$$

$$899 + 1 = 900; \quad s \sqrt{900} = 30$$

$$p = s - 1 = 30 - 1 = 29; \quad q = s + 1 = 30 + 1 = 31.$$

b) Numeri primi con differenza 4, e quindi con $d = 4/2 = 2$

$$p = 97, \quad q = 97 + 4 = 101$$

$$N = 97 * 101 = 9\,797$$

primo tentativo:

$$N + 1 = 9\,798, \quad s = \sqrt{9\,798} = 98,98 \text{ non intero}$$

secondo tentativo, con $d = 2$:

$$N + 4 = 9\,797 + 4 = 9\,801, \quad s = \sqrt{9\,801} = 99 \text{ intero}$$

$$p = s - d = 99 - 2 = 97, \quad q = s + d = 99 + 2 = 101.$$

c) numeri primi con differenza $D = 30$ e $d = 15$.

$$p = 97, \quad q = 127$$

$$N = 97 * 127 = 12\,319$$

dopo 14 tentativi con s non interi, otteniamo:

$$N = 12\,319 + 225 = 12\,544 = s^2, \quad s = \sqrt{12\,544} = 112$$

intero, e quindi questo è l'ultimo tentativo:

$p = 112 - 15 = 97$; $q = 112 + 15 = 127$. Se per esempio,

12 319 fosse un numero RSA, con la fattorizzazione

classica si devono fare 25 tentativi (poiché 97 è il 25°

numero primo), mentre con il nostro polinomio ne bastano

solo 15, con risparmio di tempo pari a $25/15 = 1,66$ volte

minore, ovvero $15/25 = 0,57$ cioè quasi la metà, il che non è

poco, Tempo che naturalmente diminuisce ancora al

diminuire di d e quindi dei tentativi fatti per arrivare ad s

numero intero e radice quadrata di $N + d^2$.

Così ora la difficoltà (tempo di calcolo) ora è minore

poiché ora si basa sul numero delle cifre di d anziché del

numero di cifre di N , e quindi con relativo risparmio di

tempo. Non è questo il nostro scopo principale (violare il

codice RSA), ma solo quello di conoscere sempre meglio i

maggiori misteri ancora rimasti sui numeri primi; come

appunto la fattorizzazione e la RH. Per quanto riguarda i problemi minori, la congettura di Goldbach (connessa con il nostro polinomio tramite la semisomma s) e la congettura dei numeri primi gemelli (anch'essa connessa col nostro polinomio tramite la semisomma $s = \sqrt{(N+1)}$) abbiamo raggiunto ottimi risultati, per esempio collegando le due congetture con la scoperta che una coppia di numeri primi gemelli è sempre l'ultima coppia d Goldbach per N pari di forma $N = 12n$ (tranne la coppia iniziale di gemelli 3 e 5, poiché $3 + 5 = 8$ che non è di forma $12n$, e questo perché tre non è, insieme a 2, un numero primo di forma $6n \pm 1$); per tutti gli altri infiniti numeri primi, la loro somma si può scrivere come

$$p + q = 6m \pm 1 + 6n \pm 1 = 6(m + n), 6(m + n) \pm 2$$

(forme che contengono tutti i numeri pari).

I numeri primi gemelli, essendo di forma $p = 6n - 1$ e

$q = 6n + 1$, si possono sommare come

$$6n - 1 + 6n + 1 = 6n + 6n = 12n.$$

Vediamo ora l'accento al logaritmo discreto, alle curve ellittiche e al relativo sistema crittografico (alternativo al sistema RSA) e usato di solito per telefonini e palmari.

Tale sistema crittografico è basato, invece che sulla difficile fattorizzazione classica, sull'altrettanto difficile calcolo del logaritmo discreto, vedasi omonima voce su Wikipedia, che qui citeremo brevemente:

“... Definiamo perciò il “logaritmo discreto” di un numero x in base a ” quel numero b tale che

$$a^b \pmod{p} = x \quad \dots$$

Il calcolo dei logaritmi discreti sembra un problema difficile (non sono noti algoritmi efficienti) mentre il problema inversa dell'esponenziazione discreta non lo è.

Questa asimmetria è analoga a quella tra la fattorizzazione

di interi e la moltiplicazione degli interi. Entrambe queste asimmetrie sono state utilizzate per la costruzione di sistemi crittografici. Poiché

“ Esistono algoritmi più sofisticati, generalmente ispirati dai simili algoritmi per la fattorizzazione degli interi”

Si potrebbe cercare ed eventualmente anche trovare un algoritmico più efficiente anche per il calcolo del logaritmo discreto, ispirandosi al nostro polinomio di fattorizzazione più veloce, oggetto di questo lavoro, molto efficiente per piccoli valori di d . E tale maggiore efficienza potrebbe presentarsi poi nell'eventuale algoritmico di calcolo più veloce del logaritmo discreto. E quindi poi più efficientemente applicabile nella crittografia a curve ellittiche, così come il nostro lo è per la crittografia RSA.

Si propone quindi ai matematici , anche ai membri e collaboratori del nostro Gruppo ERATOSTENE, la ricerca

su tale algoritmo, anche se è un compito molto difficile.

Fourier ci mise una vita inventare la teoria alla base dell'algoritmo FFT, e agli informatici ad implementarlo.

Ma non si sa mai...

GRUPPO ERATOSTENE

Caltanissetta 1.10.2010 (Data di revisione)

Riferimenti

1. “Sulle spalle dei giganti” sul nostro sito

www.gruppoeratostene.com .

2. BlockNotes matematico dell'Ing. Rosario Turco,

<http://mathbuildingblock.blogspot.com/> vedi sezione

LINK.

3. “Numeri primi in cerca d'autore” in sezione “Articoli sulla Teoria dei numeri” sul nostro sito.

5. “Fattorizzazione con algoritmo generalizzato con

quadrati perfetti in ambito delle forme $6k \pm 1$ ” sul nostro sito, in sezione “Articoli sulla Fattorizzazione”