Entriamo nel cuore del sistema di fattorizzazione GC57

Introduzione

Non posso certo permettermi di proporre un enunciato accademico in senso stretto. Tuttavia, negli ultimi tempi ho cercato di formalizzare ciò che nella mia mente rappresenta il cuore del metodo GC57: la logica aritmetica che lo rende funzionante e deterministico.

Questo documento non ha la pretesa di essere una dimostrazione matematica accettata dalla comunità accademica, ma vuole offrire una descrizione strutturata e trasparente del ragionamento che mi ha portato a sviluppare il sistema GC57.

Indice

Origine del metodo GC57

Nel contesto della fattorizzazione di numeri semiprimi, sono qui a proporre un approccio deterministico basato su una struttura asimmetrica tra due interi positivi A e B, con B≫A, cioè bit(B)>bit(A). Tale schema, denominato *GC57*, si fonda su proprietà modulari associate alla chiave C=B-1 e consente la fattorizzazione diretta di un semiprimo S=(A+x)(B+y) attraverso un semplice calcolo del massimo comune divisore

Obiettivo della formalizzazione

Lo scopo di questa formalizzazione è presentare, in modo ordinato e comprensibile, i passaggi aritmetici che spiegano come il metodo GC57 funziona.

Il focus è sul meccanismo numerico:

- la costruzione del semiprimo,
- il ruolo della chiave modulare,
- la logica del calcolo del massimo comune divisore.

Questa esposizione intende mettere in evidenza la struttura interna del metodo, senza entrare nel merito delle valutazioni accademiche o delle implicazioni crittografiche.

📘 Teorema - Intervallo di fattorizzazione GC57

Siano $A, B \in N \; con \; B \gg A$ dove la condizione $B \gg A$ indica che B è significativamente maggiore di A in termini di ordine di grandezza (tipicamente bit(B) > bit(A)).

Sia definito il prodotto massimo teorico:

$$N=(A+1)(B+1)$$

sia definita la **chiave modulare**:

$$C=B-1$$

Da questo, definiamo l’intervallo di lavoro I come:

$$I = 2 \cdot \left[\frac{C}{N \bmod C} \right]$$

Poiché C=B-1, la grandezza di I è influenzata direttamente dalla distanza in bit tra A e B. Maggiore è la distanza in bit tra A e B, maggiore è C, e più grande sarà l'intervallo I, ovvero:

$$\text{Ampiezza}(I) \sim 2^{\text{bit}(B-A)}$$

Questo implica che:

- Il GC57 può selezionare un numero elevatissimo di coppie (x,y) da testare all'interno dell'intervallo.
- Ogni coppia è compatibile con una struttura di prodotti validamente costruita da (A+x)(B+y).
- L'intervallo fornisce una zona sicura di pre-image in cui la proprietà modulare della chiave continua a funzionare.

📘 Teorema – Proprietà GC57 del modulo su B-1

Enunciato: Siano $A, B \in N$ con $B \gg A$. Siano $x,y \in I$, dove $I \subset N$ è l'intervallo deterministico definito dalla chiave $C=B-1$

Sia

$$S = (A+x)(B+y) \text{ e } C = B-1.$$

Allora:

$$MCD(S, S \bmod C) = A+x$$

Dimostrazione:

Partiamo dalla definizione:

$$S=(A+x)(B+y)$$

Poiché B=C+1, allora:

$$B+y=(C+1)+y=C+(y+1)$$

Sostituendo in S, otteniamo:

$$S=(A+x)(B+y) = (A+x)(C+(y+1)) = (A+x)C+(A+x)(y+1)$$

A questo punto possiamo ricondurci alla **forma della divisione euclidea**:

$$S=qC+r$$

dove:

- definiamo $q:=A+x$ (quoziente)
- definiamo $r:=(A+x)(y+1)$ (resto)

Pertanto:

$$S \bmod C = r = (A+x)(y+1)$$

Ora calcoliamo il massimo comune divisore tra S e $S \bmod C$:

$$MCD(S,S \bmod C)= MCD((A+x)(B+y),(A+x)(y+1))$$

Possiamo estrarre il fattore comune $A+x$:

$$=(A+x) \cdot MCD(B+y,y+1)$$

allora:

$$MCD(S, S \bmod C)=A+x$$

📘 Spiegazione

Abbiamo:

$$S = (A+x)(B+y)$$

e

$$C=B-1$$

Scriviamo:

$$B+y = C+(y+1)$$

Allora:

$$S=(A+x)(C+(y+1)) = (A+x)C+(A+x)(y+1)$$

Da cui:

$$S \bmod C = (A+x)(y+1)$$

Quindi:

$$MCD(S, S \bmod C) = MCD((A+x)(B+y),(A+x)(y+1)) = (A+x)MCD(B+y,y+1)$$

Ora osserviamo:

👉 **All'interno dell'intervallo definito dal metodo GC57**, la costruzione del semiprimo garantisce che:

$$MCD(B+y,y+1)=1$$

per ogni coppia (x,y) selezionata.

Nota

****

👉 *Questo accade perché il metodo GC57, attraverso la chiave C=B-1, identifica in modo deterministico un intervallo all'interno del quale le coppie (x,y) portano sempre alla condizione:*

MCD(B+y, y+1)=1

e di conseguenza

$$MCD(S, S \bmod C)=A+x$$

La coprimalità e la validità della proprietà non sono eventi casuali, ma sono garantite dalla mappa numerica generata dalla chiave. All'interno dell'intervallo individuato, la soluzione è sempre deterministica e certa.

👉 **L’intervallo I è deterministico e può essere calcolato a partire da S e dalla chiave C=B-1**

****

📘 Tempo costante di fattorizzazione

(Invarianza computazionale del GC57):

Sia S=(A+x)(B+y), con A+x e B+y generati come numeri primi tramite una funzione deterministica (es. `NextPrime`), secondo lo schema GC57 , con A,B ∈ S, B≫A, e sia C:=B-1 la chiave associata.

Allora la fattorizzazione del semiprimo S, attraverso la funzione:

$$MCD(S,S \bmod C)$$

avviene in **tempo computazionale costante**, indipendentemente dalla dimensione in bit di S, a condizione che:

- la chiave C=B-1 sia nota,
- A e B rispettino la condizione di asimmetria (ovvero bit(B)≫bit(A)),
- il semiprimo S sia costruito secondo lo schema GC57.

Dimostrazione (comportamentale e sperimentale):

1. La funzione usata è:

   $$f(S)=MCD(S,S\,mod\,(B-1))$$

   È una funzione composta da due operazioni fondamentali:

   - Modulo: $S \bmod C$
   - Euclide: MCD(S, ·)

2. Entrambe sono operazioni con **complessità logaritmica** rispetto al numero di bit di S, ma in pratica, grazie all'implementazione hardware/software degli algoritmi di Euclide esteso e modulo binario, il tempo di esecuzione non cresce in modo significativo neppure per $S>2^{8000}$

3. I test sperimentali eseguiti con semiprimi oltre i **50.000 bit** confermano che il tempo di esecuzione della funzione GC57 rimane costante (mediamente <1 secondo), a differenza dei metodi classici che crescono in **tempo subesponenziale**.

4. La ragione è che non avviene **nessuna ricerca o esplorazione dello spazio dei divisori**. La chiave C guida direttamente alla base A+x, rendendo il processo **non iterativo**.

****

Segue una dimostrazione sull'aumento dell'intervallo I in base alla caratteristica dei numeri coinvolti.

Esempio numerico 1:

- a = 13
- b = 19
- Ae = 14
- Be = 20
- $n = (a^{14} + 1)(b^{20} + 1)$
- $c = b^{20} - 1$
- $I = 2 \cdot [\frac{C}{N \bmod C}] = 9546959644$
- x = random(1, I), y = random(1, I)
- p = NextPrime(a^{14} + x)
- q = NextPrime(b^{20} + y)
- $S = p \cdot q$
- MCD(S, S mod c) = p

Esempio numerico 2:

- a = 875623876876125387508255267 53
- b = 23675423657652856523525649860165651
- Ae = 10
- Be = 12
- $n = (a^{10} + 1)(b^{12} + 1)$
- $c = b^{12} - 1$
- $I = 2 \cdot [\frac{C}{N \bmod C}] = 2^{408}$

• x = random(1, I), y = random(1, I)

• p = NextPrime(a^{10} + x)

• q = NextPrime(b^{12} + y)

• S = p · q

• MCD(S, S mod c) = p

****

📌 Perché funziona il metodo GC57

Il metodo GC57 si basa su una costruzione aritmetica che combina in modo deterministico la divisione euclidea e la proprietà della coprimalità dei fattori primi. La chiave C=B-1 è scelta per ottenere una riduzione del semiprimo

$$\mathbf{S=(A+x)(B+y)}$$

in una forma che conserva un'informazione diretta sul fattore minore:

$$\mathbf{S=(A+x)C+(A+x)(1+y)} \text{ e quindi } \mathbf{S \bmod C=(A+x)(1+y)}$$

Quando calcoliamo MCD(S,S mod C) otteniamo

$$\mathbf{MCD((A+x)(B+y),(A+x)(1+y))=(A+x)MCD(B+y,1+y)}$$

Se i fattori sono primi, la coprimalità è garantita:

$$\mathbf{MCD(B+y,1+y)=1}$$

e quindi

$$\mathbf{MCD(S,S \bmod C)=A+x}$$

👉 Il metodo funziona perché:

- il modulo C=B-1 isola una componente che contiene A+x in forma diretta;
- la coprimalità tra B+y e 1+y1 è garantita dalla natura prima dei fattori;
- il calcolo del massimo comune divisore agisce come un "filtro" che estrae A+x senza ambiguità o bisogno di ricerca.

Il processo è quindi deterministico: non si basa su probabilità, esplorazione dello spazio dei divisori o tentativi, ma su una struttura numerica che rende certa la fattorizzazione in un singolo passo.

Conclusioni

Il metodo GC57, basato sulla costruzione asimmetrica dei fattori e sull'uso della chiave modulare C=B-1, fornisce un approccio deterministico alla fattorizzazione di semiprimi.

L'intervallo I, calcolabile a partire da N e da C, consente di individuare coppie (x,y) che garantiscono la proprietà:

$$\mathbf{MCD(S,S \bmod C)=A+x}$$

La chiave guida direttamente al fattore, senza necessità di esplorare lo spazio dei divisori, e costituisce un'alternativa concettuale ai metodi classici di fattorizzazione.

📌 Osservazione sui fattori composti

Il metodo GC57 e la chiave **C=B-1** operano indipendentemente dalla natura di **B**, che può essere primo, composto, pari o dispari. Tuttavia, l'efficacia della funzione

$\mathbf{MCD(S,S \bmod C)=A+x}$ è garantita solo nel caso in cui **(A+x)** e **(B+y)** siano entrambi numeri primi.

Se uno o entrambi i fattori **(A+x)** o **(B+y)** non sono primi, la condizione **MCD(B+y, y+1)=1** può non essere soddisfatta, restituendo con **$\mathbf{MCD(S,S \bmod C)}$** un divisore diverso di **(A+x)**

Questo comportamento:

- non rappresenta un limite della chiave **C**;
- non compromette la validità del metodo GC57 nel contesto della sicurezza informatica;
- riflette una differenza sostanziale tra semiprimi (per i quali il metodo è concepito) e prodotti con fattori composti, che sono oggetto di semplice studio teorico e non di applicazioni crittografiche.

Appendice: Estensioni teoriche della chiave modulare.

Il metodo GC57 si fonda sull'uso della chiave

$$\mathbf{C=B-1}$$

che rappresenta la scelta ottimale per preservare la struttura aritmetica del semiprimo e garantire la validità del calcolo

$$\mathbf{MCD(S,S \bmod C)=A+x}$$

È possibile considerare, a scopo puramente teorico, varianti della chiave come

$$\mathbf{C=B-n \quad (A < C \leq B)}$$

o sottomultipli come

$$\mathbf{C=B/2 \text{ o } B^{e/2}}$$

In questi casi l'intervallo utile:

$$I = 2 \cdot [\frac{C}{N \bmod C}]$$

si restringe progressivamente man mano che C si allontana da B-1;

quando C si avvicina o scende a livello di A, l'intervallo si annulla e il metodo perde significato; tentativi di usare C > B, come C=B+1, non preservano la struttura necessaria e compromettono il funzionamento del processo.

Nota finale:

Questo saggio nasce come contributo personale alla descrizione logica e aritmetica del metodo GC57, aperto a valutazioni e approfondimenti da parte della comunità matematica e crittografica.