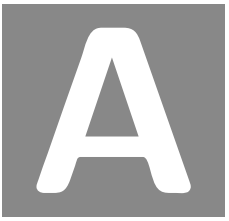
Appendici Matematica

APPENDICE



Teoria dell'informazione

Dovresti chiamarla entropia, per due ragioni. Innanzitutto la tua funzione di incertezza è già nota in meccanica statistica con quel nome. In secondo luogo, ancora più importante, nessuno sa cosa sia con certezza l'entropia, così in una discussione avrai sarai sempre in vantaggio.

J. von Neumann, in risposta a C. Shannon

La teoria dell'informazione (*Information Theory*) è una branca della matematica applicata nata alla fine degli anni '40, grazie al lavoro di Shannon [Sha48] *Una Teoria Matematica della Comunicazione* pubblicato sulla rivista tecnica dei laboratori Bell. La ricerca di Shannon riguardava principalmente il problema della compressione dell'informazione e, più in generale, della memorizzazione e della comunicazione digitale. I suoi risultati hanno gettato le basi per una vera e propria disciplina, con applicazioni in crittografia, statistica, biologia, e molti altri settori.

Guida per il lettore. Questa appendice riassume alcuni concetti di base usati in diverse parti del testo. Nel Paragrafo A.1 richiameremo il concetto di variabile aleatoria e di distanza statistica tra variabili aleatorie. Nel Paragrafo A.2 enunceremo (e dimostreremo) alcune disuguaglianze fondamentali in teoria della probabilità, in particolare il limite di Chernoff. Infine, nel Paragrafo A.3, presenteremo una breve panoramica sui concetti di entropia e di informazione mutua.

Per un'introduzione alla teoria delle probabilità si vedano ad esempio [Tij04; Gut05], per approfondire i concetti di teoria dell'informazione si suggerisce una lettura di [CK97; CT06].

Venturi D.: Crittografia nel Paese delle Meraviglie
DOI 10.1007/978-88-470-2481-6_15, © Springer-Verlag Italia 2012

A.1 Variabili aleatorie

Una variabile aleatoria (*random variable*) può essere pensata come l'output di un esperimento del quale non si può predire con certezza il risultato. A seconda di come è descritto l'output dell'esperimento parleremo di variabili aleatorie discrete o di variabili aleatorie continue. Nel seguito ci soffermiamo solo sulle prime. La nostra trattazione sarà principalmente informale; si raccomanda [Bil79] per un approccio rigoroso.

Variabili aleatorie discrete. Essenzialmente una variabile aleatoria discreta X mappa un evento $\mathcal{E}$ in un valore di un insieme numerabile $\mathcal{X}$ (ad esempio l'insieme dei numeri interi); ciascun valore ha associata una quantità — detta *probabilità* — maggiore o uguale di zero e minore o uguale di 1. In questo modo, si associa ad ogni variabile aleatoria discreta una distribuzione di probabilità, detta anche densità discreta o funzione di massa di probabilità.

Consideriamo, ad esempio, l'evento $\mathcal{E}$ che descrive il lancio di un dado (non truccato e con sei facce). Dato l'insieme $\mathcal{X} = \{1, 2, 3, 4, 5, 6\}$, la variabile aleatoria X associa ad ogni possibile risultato del lancio del dado un elemento dell'insieme $\mathcal{X}$:

$$X = \begin{cases} 1 & \text{se esce } \boxdot, \\ 2 & \text{se esce } \boxdot, \\ 3 & \text{se esce } \boxdot, \\ 4 & \text{se esce } \boxdot, \\ 5 & \text{se esce } \boxdot, \\ 6 & \text{se esce } \boxdot. \end{cases}$$

La densità discreta sarà quindi

$$P_X(x) = \begin{cases} 1/6 & \text{se } x = 1, \ldots, 6 \\ 0 & \text{altrimenti.} \end{cases}$$

Più in generale, data una variabile aleatoria aleatoria X definita su un'insieme numerabile $\mathcal{X}$, la sua densità discreta nel punto $x \in \mathcal{X}$ è la probabilità che X assuma il valore x. In simboli:

$$P_X(x) = \mathbb{P}\left[X = x\right] \quad \forall x \in \mathcal{X}.$$

Siccome la somma delle probabilità di tutti gli eventi è l'evento certo, si ha sempre $\sum_{x \in \mathcal{X}} P_X(x) = 1$.

Indipendenza statistica. Quando si ha a che fare con due variabili aleatorie contemporaneamente, in genere non è sufficiente conoscere i valori di probabilità delle singole variabili. Intuitivamente il motivo è che le due variabili aleatorie possono essere in qualche modo "correlate". In questo senso, la coppia $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ è essa stessa una variabile aleatoria caratterizzata dalla densità discreta (detta *congiunta*):

$$P_{X,Y}(X, Y) = \mathbb{P}[X = x \wedge Y = y].$$

Le densità discrete delle singole variabili aleatorie componenti, cosiddette *marginali*, si ottengono per saturazione della congiunta:

$$P_X(x) = \sum_{y \in \mathcal{Y}} P_{X,Y}(x, y) \tag{A.1}$$

$$P_Y(y) = \sum_{x \in \mathcal{X}} P_{X,Y}(x, y). \tag{A.2}$$

Possiamo esprimere il rapporto tra densità congiunta e marginale attraverso il concetto di *densità discreta condizionata*. La probabilità dell'evento $\mathcal{E}_1$ condizionata all'evento $\mathcal{E}_2$ è la probabilità che l'evento $\mathcal{E}_1$ si verifichi condizionatamente al fatto che si verifichi $\mathcal{E}_2$. Ad esempio, tornando all'esempio del dado, la probabilità che esca ⚄ sapendo che il risultato del lancio sarà ⚀, ⚄ oppure ⚃ è 1/3. Formalmente, abbiamo:

$$\mathbb{P}[\mathcal{E}_1 \mid \mathcal{E}_2] = \frac{\mathbb{P}[\mathcal{E}_1 \wedge \mathcal{E}_2]}{\mathbb{P}[\mathcal{E}_2]}.$$

Due eventi si dicono *statisticamente indipendenti* quando $\mathbb{P}[\mathcal{E}_1 \mid \mathcal{E}_2] = \mathbb{P}[\mathcal{E}_1]$ (ovvero quando la probabilità che $\mathcal{E}_1$ si verifichi dato che si verifica $\mathcal{E}_2$ è uguale alla probabilità che $\mathcal{E}_1$ si verifichi a priori). La relazione tra $\mathbb{P}[\mathcal{E}_1 \mid \mathcal{E}_2]$ e $\mathbb{P}[\mathcal{E}_2 \mid \mathcal{E}_1]$ è data dal teorema di Bayes, espresso dalla seguente equazione:

$$\mathbb{P}[\mathcal{E}_2 \mid \mathcal{E}_1] = \mathbb{P}[\mathcal{E}_1 \mid \mathcal{E}_2] \cdot \frac{\mathbb{P}[\mathcal{E}_2]}{\mathbb{P}[\mathcal{E}_1]}. \tag{A.3}$$

Passando alle densità, date due variabile aleatorie $X \in \mathcal{X}$ ed $Y \in \mathcal{Y}$, la probabilità condizionata di $Y = y$ dato $X = x$ è:

$$\mathbb{P}[Y = y \mid X = x] = \frac{\mathbb{P}[X = x \wedge Y = y]}{\mathbb{P}[X = x]} \overset{(A.3)}{=} \frac{\mathbb{P}[Y = y | X = x]\,\mathbb{P}[Y = y]}{\mathbb{P}[X = x]}.$$

Quindi,

$$P_{X,Y}(x,y) = P_{Y|X}(y|x)P_X(x) = P_{X|Y}(x|y)P_Y(y).$$

Analogamente a quanto visto per gli eventi, due variabile aleatorie si dicono statisticamente indipendenti se $P_{X,Y}(x,y) = P_X(x)P_Y(y)$.

Valore atteso e varianza. La densità discreta di una variabile aleatoria X è caratterizzata da diversi parametri. In alcuni casi può essere sufficiente conoscere quale valore assume la variabile aleatoria in media; si definisce quindi il *valore atteso* di X come:

$$\mu = \mathbb{E}[X] = \sum_{x \in \mathcal{X}} x P_X(x).$$

Non è difficile mostrare che il valore atteso è un operatore lineare: il valore atteso della somma di più variabili aleatorie altro non è che la somma dei singoli valori attesi. Riportiamo il caso di due variabili aleatorie; l'estensione al caso generale è un semplice esercizio (cf. Esercizio A.2).

Lemma A.1 (Linearità del valore atteso). *Siano X ed Y due variabile aleatorie discrete definite (rispettivamente) su due insiemi $\mathcal{X}$ ed $\mathcal{Y}$. Per ogni $\alpha, \beta \in \mathbb{R}$ abbiamo $\mathbb{E}[\alpha X + \beta Y] = \alpha \mathbb{E}[X] + \beta \mathbb{E}[Y]$.*

Dimostrazione. La dimostrazione segue direttamente dalla definizione di densità congiunta:

$$\begin{aligned}
\mathbb{E}[\alpha X + \beta Y] &= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} (\alpha x + \beta y) \mathbb{P}[X = x \wedge Y = y] \\
&= \alpha \sum_{x \in \mathcal{X}} x \sum_{y \in \mathcal{Y}} \mathbb{P}[X = x \wedge Y = y] + \beta \sum_{y \in \mathcal{Y}} y \sum_{x \in \mathcal{X}} \mathbb{P}[X = x \wedge Y = y] \\
&= [\text{usando l'Eq. (A.1) e l'Eq. (A.2)}] \\
&= \alpha \sum_{x \in \mathcal{X}} x \cdot \mathbb{P}[X = x] + \beta \sum_{y \in \mathcal{Y}} y \cdot \mathbb{P}[Y = y] \\
&= \alpha \mathbb{E}[X] + \beta \mathbb{E}[Y].
\end{aligned}$$

□

Anche quando è noto il valore medio, è lecito chiedersi quanto i valori di X (tipicamente) si allontanano da $\mathbb{E}[X]$; a tale scopo si definisce la *varianza* di una variabile aleatoria come:

$$\sigma^2 = \mathbb{V}[X] = \mathbb{E}\left[(X - \mu)^2\right].$$

La quantità $\sigma = \sqrt{\sigma^2}$ è detta *deviazione standard*. Un semplice calcolo mostra:

$$\begin{aligned}\mathbb{V}[X] &= \mathbb{E}\left[(X-\mu)^2\right] = \mathbb{E}\left[X^2 - 2\mu X + \mu^2\right] \\ &= [\text{per la linearità del valore atteso (cf. Lemma A.1)}] \\ &= \mathbb{E}\left[X^2\right] - 2\mu^2 + \mu^2 \\ &= \mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2.\end{aligned}$$

La varianza (in generale) non è lineare. In alcuni casi particolari però la proprietà di linearità è soddisfatta. Di particolare interesse è il caso di variabili aleatorie *indipendenti a coppie*. Diremo che n variabili aleatorie $X_1, \ldots, X_n$ definite sull'insieme $\mathcal{X}$ sono indipendenti a coppie, se per ogni $i, j \in [n]$ (con $i \neq j$) e per ogni $a, b \in \mathbb{R}$, risulta:

$$\mathbb{P}[X_i = a \wedge X_j = b] = \mathbb{P}[X_i = a] \cdot \mathbb{P}[X_j = b].$$

Nel caso in cui l'indipendenza a coppie è soddisfatta la varianza della variabile aleatoria somma coincide con la somma delle singole varianze:

Lemma A.2 (Linearità della varianza). *Siano $X_1, \ldots, X_n$ variabili aleatorie indipendenti a coppie; poniamo $X = \sum_{i=1}^{n} X_i$. Allora:*

$$\mathbb{V}[X] = \sum_{i=1}^{n} \mathbb{V}[X_i].$$

Dimostrazione. Basta applicare la definizione e sfruttare la proprietà di linearità del valore atteso:

$$\begin{aligned}\mathbb{V}[X] &= \mathbb{E}\left[(X_1 + \ldots + X_n)^2\right] - (\mathbb{E}[X_1 + \ldots + X_n])^2 \\ &= \mathbb{E}[(X_1 + \ldots + X_n) \cdot (X_1 + \ldots + X_n)] - (\mathbb{E}[X_1 + \ldots + X_n])^2 \\ &= \mathbb{E}\left[\sum_{i,j} X_i X_j\right] - \sum_{i,j} \mathbb{E}[X_i] \cdot \mathbb{E}[X_j] \\ &= [\text{dalla linearità del valore atteso}] \\ &= \sum_{i,j} \mathbb{E}[X_i \cdot X_j] - \sum_{i,j} \mathbb{E}[X_i] \cdot \mathbb{E}[X_j] \\ &= \sum_{i} \mathbb{E}\left[X_i^2\right] + \sum_{i \neq j} \mathbb{E}[X_i \cdot X_j] - \sum_{i} (\mathbb{E}[X_i])^2 - \sum_{i \neq j} \mathbb{E}[X_i] \cdot \mathbb{E}[X_j]\end{aligned}$$

$$= \sum_i \left(\mathbb{E}\left[X_i^2\right] - (\mathbb{E}\left[X_i\right])^2 \right) + \sum_{i \neq j} \left(\mathbb{E}\left[X_i \cdot X_j\right] - \mathbb{E}\left[X_i\right] \cdot \mathbb{E}\left[X_j\right] \right)$$
$$= \sum_i \mathbb{V}\left[X_i\right] + \sum_{i \neq j} \left(\mathbb{E}\left[X_i \cdot X_j\right] - \mathbb{E}\left[X_i\right] \cdot \mathbb{E}\left[X_j\right] \right).$$

Siccome le variabili aleatorie $X_1, \ldots, X_n$ sono indipendenti a coppie, l'ultima sommatoria è nulla. □

Spazi metrici e distanza statistica. Dato uno spazio, possiamo dotarlo di una struttura arricchendolo con una *metrica* (detta anche *distanza*). In questo modo gli elementi dello spazio sono luoghi geometrici che (quando non identici) hanno distanza positiva.

Definizione A.1 (Metrica su uno spazio $\mathcal{X}$). Si definisce metrica una funzione $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ che soddisfa le seguenti proprietà:

- *Definita positiva.* $d(x, y) \geq 0$, $\forall\ x, y \in \mathcal{X}$ e $d(x, y) = 0$ se e solo se $x = y$.
- *Simmetria.* $d(x, y) = d(y, x)$, $\forall\ x, y \in \mathcal{X}$.
- *Disuguaglianza triangolare.* $d(x, z) \leq d(x, y) + d(y, z)$, $\forall\ x, y, z \in \mathcal{X}$.

■

In genere la scelta non è univoca, ed infatti esistono tante metriche. Un esempio classico è costituito dallo spazio delle stringhe binarie a lunghezza n (detto anche *spazio di Hamming* ed indicato con $\{0,1\}^n$), dove si definisce la distanza di Hamming.

Definizione A.2 (Distanza di Hamming). Date due stringhe binarie $\mathbf{x}, \mathbf{y} \in \{0,1\}^n$ la loro distanza di Hamming $d_H : \{0,1\}^n \times \{0,1\}^n \to \mathbb{R}^+$ è definita come:

$$d_H(\mathbf{x}, \mathbf{y}) = \#\left\{i :\ x_i \neq y_i, \forall\ i = 1, 2, \ldots, n\right\}.$$

■

Non è complesso verificare che la distanza di Hamming è una metrica. In effetti ciò deriva dal fatto che

$$d_H(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i|;$$

pertanto, siccome la funzione valore assoluto è una metrica, anche la distanza di Hamming lo è. Si definisce il *peso di Hamming* di una stringa binaria $\mathbf{x} \in \{0,1\}^n$ come il numero di valori "1" in $\mathbf{x}$. Più precisamente $w_H(\mathbf{x}) = d_H(\mathbf{x}, \mathbf{0})$, dove $\mathbf{0} = (0, \ldots, 0)$ è il vettore tutto nullo in $\{0,1\}^n$.

Un secondo esempio è quello dello spazio euclideo $\mathbb{R}^n$. Dato un vettore $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$, sia

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$$

la sua norma l-1. È immediato verificare che la mappa $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_1$ è una metrica. Analogamente, se consideriamo la norma l-2:

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2}, \tag{A.4}$$

è banale verificare che la mappa $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$ è una metrica.[87] (Questa non è altro che la familiare distanza euclidea definita dal teorema di Pitagora nel piano Cartesiano.) Osserviamo che, per ogni vettore $\mathbf{x} \in \mathbb{R}^n$, risulta:

$$\|\mathbf{x}\|_1 \leq \sqrt{n}\|\mathbf{x}\|_2. \tag{A.5}$$

Generalizzando questi esempi, è possibile definire una distanza tra distribuzioni di probabilità discrete. Tale distanza è detta *distanza statistica*.

Definizione A.3 (Distanza statistica). Date due variabili aleatorie X_0, X_1 con valori in $\mathcal{X}$, la loro distanza statistica è:

$$\Delta(X_0, X_1) = \frac{1}{2} \sum_{x \in \mathcal{X}} \left| \mathbb{P}[X_0 = x] - \mathbb{P}[X_1 = x] \right|.$$

(Notare che $\Delta(X_0, X_1) = 1/2\|X_0 - X_1\|_1$.) ■

Sia U_n la distribuzione uniforme su $\{0,1\}^n$. In genere, quando X assume valori in $\mathcal{X} = \{0,1\}^n$, si è soliti indicare con $d(X) = \Delta(X, U_n)$ la distanza

[87] Più in generale, per ogni $p > 0$, data la norma l-p:

$$\|\mathbf{x}\|_p = \sqrt[p]{\sum_{i=1}^{n} |x_i|^p},$$

la mappa $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_p$ è una metrica.

statistica tra X e la distribuzione uniforme su $\{0,1\}^n$. Quando $d(X) \leq \epsilon$ si dice che X è ϵ-vicina alla distribuzione uniforme. Analogamente:

$$\Delta(X_0, X_1|Y) = \Delta((X_0, Y), (X_1, Y))$$
$$d(X|Y) = \Delta((X, Y), (U_n, Y)).$$

Si può dimostrare che la distanza statistica è una metrica (cf. Definizione A.1).

A.2 Alcune disuguaglianze

In questo paragrafo studieremo alcune disuguaglianze molto importanti in teoria della probabilità. La seguente disuguaglianza di Markov collega probabilità e valori attesi.

Lemma A.3 (Disuguaglianza di Markov). *Ogni variabile aleatoria X non negativa soddisfa:*

$$\mathbb{P}[X \geq n] \leq \frac{\mathbb{E}[X]}{n}.$$

Dimostrazione. Possiamo scrivere:

$$\mathbb{E}[X] = \sum_{x \geq 0} x \cdot \mathbb{P}[X = x] \geq \sum_{0 \leq x < n} \mathbb{P}[X = x] \cdot 0 + \sum_{x \geq n} \mathbb{P}[X = x] \cdot n$$
$$= \mathbb{P}[X \geq n] \cdot n.$$

□

La disuguaglianza di Chebyshev garantisce che, per ogni variabile aleatoria, quasi tutti i valori sono "vicini alla media".

Lemma A.4 (Disuguaglianza di Chebyshev). *Sia X una variabile aleatoria con varianza σ^2. Per ogni $n > 0$, risulta:*

$$\mathbb{P}[|X - \mathbb{E}[X]| \geq n] \leq \frac{\sigma^2}{n^2}.$$

Dimostrazione. Definiamo la variabile aleatoria non negativa $Y = (X - \mathbb{E}[X])^2$ ed applichiamo la disuguaglianza di Markov ad Y:

$$\mathbb{P}[|X - \mathbb{E}[X]| \geq n] \leq \mathbb{P}\left[(X - \mathbb{E}[X])^2 \geq n^2\right] = \mathbb{P}\left[Y \geq n^2\right]$$
$$\leq \frac{\mathbb{E}[Y]}{n^2} = \frac{\mathbb{E}\left[(X - \mathbb{E}[X])^2\right]}{n^2} = \frac{\sigma^2}{n^2}.$$

□

La stima fornita dalla disequazione di Chebyshev è in un certo senso grossolana. Il seguente teorema di Chernoff, consente una caratterizzazione più fine (cf. Esercizio A.6).

Teorema A.5 (Limite di Chernoff). *Siano $X_1, X_2, \ldots, X_n$, variabili aleatorie mutuamente indipendenti su $\{0,1\}$ e sia $\mu = \sum_{i=1}^{n} \mathbb{E}[X_i]$. Allora:*

(I) $\forall\ \delta > 0$,

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \geq (1+\delta)\mu\right] \leq \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu};$$

(II) $\forall\ 0 < \delta \leq 1$,

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \geq (1+\delta)\mu\right] \leq e^{-\mu\delta^2/3};$$

(III) $\forall\ c \geq 6\mu$,

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \geq c\right] \leq 2^{-c}.$$

Dimostrazione. Definiamo $X = \sum_{i=1}^{n} X_i$. Posto $p_i = \mathbb{P}[X_i = 1]$, si ha $\mu = \sum_{i=1}^{n} p_i$. Introduciamo un parametro t il cui ruolo sarà noto a breve; per ogni $t > 0$, possiamo scrivere:

$$\mathbb{P}[X > (1+\delta)\mu] = \mathbb{P}\left[e^{tX} > e^{t(1+\delta)\mu}\right].$$

Applicando la disuguaglianza di Markov al secondo membro, otteniamo:

$$\mathbb{P}[X > (1+\delta)\mu] \leq \frac{\mathbb{E}\left[e^{tX}\right]}{e^{t(1+\delta)\mu}}. \tag{A.6}$$

Notare che, siccome le variabili aleatorie X_i sono mutuamente indipendenti, abbiamo:

$$\mathbb{E}\left[e^{tX}\right] = \mathbb{E}\left[e^{t\sum_{i=1}^{n} X_i}\right] = \mathbb{E}\left[\prod_{i=1}^{n} e^{tX_i}\right] = \prod_{i=1}^{n} \mathbb{E}\left[e^{tX_i}\right]. \tag{A.7}$$

Possiamo inoltre limitare superiormente il termine e^{tX_i} sfruttando il fatto che $e^x > 1 + x$ e che $X_i \in \{0,1\}$:

$$\mathbb{E}\left[e^{tX_i}\right] = p_i e^t + (1 - p_i) = 1 + (e^t - 1)p_i \leq e^{(e^t-1)p_i}.$$

Ma allora, sostituendo nell'Eq. (A.7), si ottiene:

$$\mathbb{E}\left[e^{tX}\right] < \prod_{i=1}^{n} e^{(e^t-1)p_i} = e^{\sum_{i=1}^{n}(e^t-1)p_i} = e^{(e^t-1)\mu}.$$

Sostituendo quest'ultima espressione nell'Eq. (A.6), infine, otteniamo:

$$\mathbb{P}\left[X > (1+\delta)\mu\right] \leq \frac{\mathbb{E}\left[e^{tX}\right]}{e^{t(1+\delta)\mu}} < \frac{e^{(e^t-1)\mu}}{e^{t(1+\delta)\mu}} = \left(e^{e^t-1-t(1+\delta)}\right)^{\mu}.$$

L'ultima espressione è valida per ogni $t > 0$. Per ottenere l'asserto non ci resta che trovare il t ottimo, che rende la disuguaglianza stretta: vogliamo cioè minimizzare $e^t - 1 - t(1+\delta)$. Posto

$$\frac{d}{dt}(e^t - 1 - t(1+\delta)) = e^t - 1 - \delta \stackrel{!}{=} 0,$$

si ottiene la soluzione unica $t = \ln(1+\delta)$. Quindi:

$$\mathbb{P}\left[X > (1+\delta)\mu\right] < \left(e^{\delta-(1+\delta)\ln(1+\delta)}\right)^{\mu} = \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu},$$

che è esattamente l'espressione in (I).

Per dimostrare la (II) scriviamo l'espansione in serie di Mclaurin[88] del termine $\ln(1+\delta)$, ovvero:

$$(1+\delta)\ln(1+\delta) = (1+\delta)\sum_{i\geq 1}(-1)^{i+1}\frac{\delta^i}{i} = \delta + \sum_{i\geq 2}(-1)^i\delta^i\left(\frac{1}{i-1} - \frac{1}{i}\right).$$

[88] In analisi matematica, un risultato di Taylor consente di approssimare una funzione (differenziabile) attorno ad un dato punto attraverso alcuni polinomi, detti polinomi di Taylor, i cui coefficienti dipendono solo dalle derivate della funzione nel punto. In particolare sia $f : [a,b] \to \mathbb{R}^n$ una funzione differenziabile n volte nell'intervallo $[a,b]$ e sia $x_0 \in [a,b]$. Allora:

$$f(x) = \sum_{i=0}^{\infty}\frac{f^{(i)}(x_0)}{i!}(x-x_0)^i,$$

dove con $f^{(i)}(x_0)$ si intende la derivata i-sima di f calcolata nel punto x_0. Quando $x_0 = 0$ si parla anche di *serie di Mclaurin*. Un semplice calcolo mostra:

$$\ln(1+x) = \sum_{i=1}^{\infty}(-1)^{i+1}\frac{x^i}{i} \qquad \text{quando } -1 < x \leq 1.$$

Se $0 \leq \delta < 1$ possiamo ignorare i termini di ordine superiore, ottenendo:

$$(1+\delta)\ln(1+\delta) > \delta + \frac{\delta^2}{2} - \frac{\delta^3}{6} \geq \delta - \frac{\delta^2}{3};$$

quindi

$$\mathbb{P}\left[X > (1+\delta)\mu\right] < \left(e^{\delta-(1+\delta)\ln(1+\delta)}\right)^{\mu} \leq e^{-\frac{\delta^2\mu}{3}} \qquad (0 \leq \delta < 1),$$

come desiderato.

Per quanto riguarda la (III), infine, sia $c = (1+\delta)\mu$. Quando $c \geq 6\mu$ abbiamo $\delta = c/\mu - 1 \geq 5$, quindi usando quanto dimostrato in (I):

$$\begin{aligned}\mathbb{P}\left[X \geq c\right] &\leq \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu} \\ &\leq \left(\frac{e}{1+\delta}\right)^{(1+\delta)\mu} \\ &\leq \left(\frac{e}{6}\right)^{c} \leq 2^{-c}.\end{aligned}$$

□

Si possono considerare alcune varianti, discusse brevemente di seguito. Un procedimento simile a quello usato nella dimostrazione del Teorema A.5 mostra la validità delle seguenti disuguaglianze:

(I') $\forall\ 0 < \delta < 1$,

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \leq (1-\delta)\mu\right] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right)^{\mu};$$

(II') $\forall\ 0 < \delta < 1$,

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \leq (1-\delta)\mu\right] \leq e^{-\mu\delta^2/2}.$$

In particolare, la seconda disuguaglianza afferma che la probabilità che l'output di n esperimenti indipendenti sia δ-lontano dal valore atteso (per $0 \leq \delta < 1$), è dell'ordine di $e^{-\Omega(n\delta^2)}$.

A.3 Entropia

Il termine *entropia* fa riferimento all'uso che ne ha fatto per primo Shannon in [Sha48]. Data una variabile aleatoria $X \in \mathcal{X}$, definiamo la sua entropia (detta appunto di Shannon) come:

$$H(X) = -\sum_{x \in \mathcal{X}} P_X(x) \log P_X(x).$$

Intuitivamente, l'entropia di una variabile aleatoria può essere vista come una *misura del contenuto informativo* associato alla variabile aleatoria stessa; in altri termini la quantità $H(X)$ rappresenta la quantità d'informazione contenuta in X.

Consideriamo ad esempio un lancio di moneta, non necessariamente bilanciata, dove sono note le probabilità che esca "testa" oppure "croce". L'entropia dell'esito del prossimo lancio è massima se la moneta non è truccata, cioè quando "testa" e "croce" hanno la stessa probabilità (pari a 1/2) di verificarsi. Questa è infatti la massima situazione di incertezza, in quanto è più difficile prevedere se uscirà "testa" oppure "croce". D'altra parte, quando la moneta è truccata uno degli esiti avrà una probabilità maggiore di verificarsi e quindi c'è meno incertezza, che si riflette in una minore entropia. Il caso estremo è quello in cui la moneta ha lo stesso simbolo su entrambe le facce: in questo caso non c'è incertezza e l'entropia è zero (cf. Fig. A.1).

L'estensione della definizione di entropia al caso di due variabili aleatorie è pressoché immediata:

$$H(X,Y) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_{X,Y}(x,y) \log P_{X,Y}(x,y).$$

Intuitivamente saremmo portati a dire che la quantità d'informazione associata ad X ed Y (congiuntamente) è sempre maggiore della quantità d'informazione associata alla sola X. In effetti, applicando le definizione, non è complesso verificare che $H(X,Y) \geq H(X)$ per ogni coppia di variabili aleatorie $X \in \mathcal{X}$ ed $Y \in \mathcal{Y}$ (cf. Esercizio A.9). Se alla quantità $H(X,Y)$ sottraiamo il contenuto informativo associato ad X otteniamo la quantità d'informazione residua in Y "dopo aver visto X". Tale quantità è detta *entropia condizionata* di Y dato X ed è definita come $H(Y|X) = H(X,Y) - H(X)$. Seguendo il significato intuitivo di entropia, date n variabili aleatorie $X_1, \ldots, X_n$ a valori in $\mathcal{X}$, saremmo portati a pensare che la loro entropia congiunta è esattamente il contenuto d'informazione in X_1 più il contributo di X_2 dopo aver visto X_1 (cioè $H(X_2|X_1)$), più il contributo di X_3 dopo aver visto X_1, X_2 (cioè $H(X_3|X_1, X_2)$), e così via. In effetti

si dimostra che, per ogni scelta delle variabili aleatorie $X_1, \ldots, X_n$, si ha esattamente $H(X_1, \ldots, X_n) = H(X_1) + \sum_{i=2}^{n} H(X_i|X_1, \ldots, X_{i-1})$. Quest'ultima uguaglianza è detta regola della catena (*chain rule*, cf. Esercizio A.10).

Una metafora per l'entropia. Possiamo rendere più esplicita la metafora di "entropia come contenuto informativo associato ad una variabile aleatoria" attraverso il concetto di misura additiva su insiemi. Sia $\mu(\mathcal{X})$ una misura associata all'insieme $\mathcal{X}$ (ad esempio la sua cardinalità). Associamo $\mu(\mathcal{X})$ all'entropia $H(X)$ (cf. anche Fig. A.2). Dati due insiemi $\mathcal{X}$ ed $\mathcal{Y}$ la misura dell'unione non è altro che la misura di $\mathcal{X}$ più la misura di $\mathcal{Y} \setminus \mathcal{X}$, ovvero $\mu(\mathcal{X} \cup \mathcal{Y}) = \mu(\mathcal{X}) + \mu(\mathcal{Y} \setminus \mathcal{X})$. Allo stesso modo $H(X, Y) = H(X) + H(Y|X)$. Continuando a seguire questa metafora, ci aspettiamo che la misura associata ad $\mathcal{Y}$ sia sempre maggiore di quella associata ad $\mathcal{Y} \setminus \mathcal{X}$, ovvero $\mu(\mathcal{Y}) \geq \mu(\mathcal{Y} \setminus \mathcal{X})$. In effetti, si può dimostrare che $H(Y) - H(Y|X) \geq 0$ ed in particolare l'uguaglianza è valida se e solo se le variabili aleatorie X ed Y sono statisticamente indipendenti. La quantità:

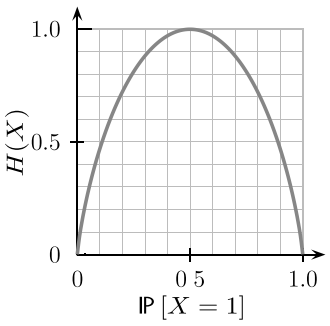


Fig. A.1. Entropia $H(X)$ di un lancio di moneta (misurata in bit) al variare del bilanciamento della moneta

$$I(X \wedge Y) = H(Y) - H(Y|X) = H(X) - H(X|Y) \geq 0,$$

è detta *informazione mutua* delle variabili aleatorie X ed Y e nella nostra metafora rappresenta la misura dell'intersezione $\mu(\mathcal{X} \cap \mathcal{Y})$.[89]

Mettendo insieme quest'ultima disuguaglianza con le definizioni precedenti,

[89] Sebbene questa metafora aiuti a visualizzare molti concetti, bisogna utilizzarla con accortezza. Ecco un controesempio. Si definisce *informazione mutua condizionata* la quantità $I(X \wedge Y|Z) = H(X|Z) + H(Y|X) - H(X, Y|Z)$. La nostra metafora ci porta a dire che $I(X \wedge Y) \geq I(X \wedge Y|Z)$, perché tale quantità corrisponde alla misura dell'intersezione $\mu(\mathcal{X} \cap \mathcal{Y} \cap \mathcal{Z})$. Purtroppo ciò non sempre è verificato, in particolare esistono casi in cui $I(X \wedge Y) \geq I(X \wedge Y|Z)$ ed altri casi in cui $I(X \wedge Y) < I(X \wedge Y|Z)$ (cf. Esercizio A.11).

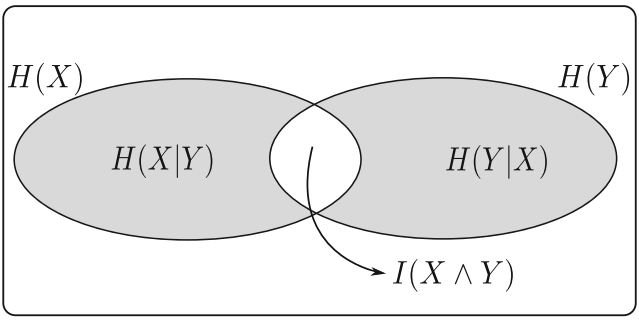


Fig. A.2. Entropia e misura di insiemi

troviamo:

$$\begin{aligned} I(X \wedge Y) &= H(Y) - H(Y|X) \\ &= H(Y) - (H(X,Y) - H(X)) \\ &= H(X) + H(Y) - H(X,Y) \\ &\geq 0, \end{aligned}$$

ed il segno uguale si verifica quando e solo quando X ed Y sono statisticamente indipendenti. Ciò significa che l'informazione associata ad X ed Y (se correlate), è maggiore della somma dei contributi informativi associati ad X ed Y prese singolarmente.

Entropia minima. L'entropia di Rényi [Rén61], è una generalizzazione dell'entropia di Shannon, utile a quantificare diversi gradi di incertezza associati ad una variabile aleatoria generica.

Definizione A.4 (Entropia di Rényi). Sia X una variabile aleatoria su $\mathcal{X} = \{x_1, \ldots, x_n\}$. L'entropia di Rényi di ordine α, con $\alpha \geq 0$, è definita come:

$$\mathbf{H}_\alpha(X) = \frac{1}{1-\alpha} \log \left(\sum_{i=1}^{n} p_i^\alpha \right),$$

essendo $p_i = \mathbb{P}[X = x_i]$. ■

Alcuni casi particolari sono riportati di seguito.

- $\alpha = 0$. Abbiamo $\mathbf{H}_0(X) = \log n = \log |\mathcal{X}|$, che è il logaritmo della cardinalità di $\mathcal{X}$, a volte detto *entropia di Hartley*.
- $\alpha \to 1$. Nel limite per α che tende ad 1, abbiamo:

$$\mathbf{H}_1(X) = \lim_{\alpha \to 1} \mathbf{H}_\alpha(X) = -\sum_{i=1}^{n} p_i \log p_i,$$

  che è l'entropia di Shannon $H(X)$.
- $\alpha \to \infty$. Nel limite per $\alpha \to \infty$ si parla di entropia minima (*min-entropy*):

$$\mathbf{H}_\infty(X) = \lim_{\alpha \to \infty} \mathbf{H}_\alpha(X) = -\log \sup_{i=1,\ldots,n} p_i = -\log \max_{x \in \mathcal{X}} \mathbb{P}\left[X = x\right],$$

  così detta perché è il valore più piccolo di $\mathbf{H}_\alpha$.

L'entropia minima misura la quantità di randomicità in X. Infatti la quantità $\max_{x \in \mathcal{X}} \mathbb{P}\left[X = x\right] = 2^{-\mathbf{H}_\infty(X)}$ rappresenta la massima probabilità di predire correttamente X; in questo senso, l'entropia minima misura quanto è difficile predire X.

Esercizi

Esercizio A.1. Sia $\mathscr{A}$ un algoritmo che prende come input una stringa binaria $x \in \{0,1\}^n$ e restituisce 1 se e solo se i primi 3 bit di x sono 1. Sia $(\mathsf{Enc}, \mathsf{Dec})$ il cifrario OTP del Crittosistema 2.1. Calcolare:

$$\mathbb{P}\left[\mathscr{A}(c) = 1 : \ k \xleftarrow{\$} \{0,1\}^n, c \leftarrow \mathsf{Enc}_k(1^n)\right].$$

Esercizio A.2. Estendere il Lemma A.1 al caso di $n > 2$ variabili aleatorie.

Esercizio A.3. La legge debole dei grandi numeri dice che se $X_1, X_2, X_3, \ldots$ sono variabili aleatorie indipendenti ed identicamente distribuite, con valore atteso μ e deviazione standard σ, allora per ogni $\epsilon > 0$ si ha:

$$\lim_{n\to\infty} \mathbb{P}\left[\left|\frac{X_1 + X_2 + \ldots + X_n}{n} - \mu\right| > \epsilon\right] = 0.$$

Dimostrare la legge debole dei grandi numeri, utilizzando il seguente schema:

1. Sia $X^* = \frac{X_1+X_2+\ldots+X_n}{n}$. Calcolare $\mu^* = \mathbb{E}[X]$.
2. Calcolare $\sigma^* = \mathbb{V}[X]$ e mostrare che per ogni costante $\epsilon > 0$ si ha $\epsilon \geq \sigma/n = \sqrt{n}\sigma^*$ per n sufficientemente grande.
3. Applicare la disuguaglianza di Chebyshev usando i valori determinati ai passi precedenti.

Esercizio A.4. Consideriamo n variabili aleatorie Bernoulliane $X_i \xleftarrow{\$} \mathsf{Ber}_\tau$, ovvero $\mathbb{P}[X_i = 1] = \tau$ per ogni $i \in [n]$. Determinare la probabilità che $\sum_{i=1}^n X_i > n \cdot \tau'$, per una costante $\tau < \tau' < n$.

Esercizio A.5. Consideriamo n variabili aleatorie $X_i \xleftarrow{\$} \{0,1\}$. Determinare la probabilità che $\sum_{i=1}^n X_i \leq n \cdot \tau'$, per una costante $\tau < \tau' < 1/2$.

Esercizio A.6. Sia X la variabile aleatoria che rappresenta il numero di valori "testa" in n lanci di moneta indipendenti. Applicare la disuguaglianza di Chebyshev per limitare la probabilità che X sia più piccola di $n/4$ e più grande di $3n/4$. Confrontare il risultato con lo stesso limite ottenuto applicando il teorema di Chernoff.

Esercizio A.7. Una moneta non truccata è lanciata fintanto che non si ottiene "testa". Sia X il numero di lanci; determinare $H(X)$.

Esercizio A.8. Consideriamo le variabili aleatorie binarie X, Y, con distribuzione di probabilità congiunta determinata dalle equazioni:

$$P_{XY}(0,0) = 1/6 \qquad P_{XY}(0,1) = 1/6$$
$$P_{XY}(1,0) = 1/2 \qquad P_{XY}(1,1) = 1/6.$$

Calcolare $H(X), H(Y), H(X|Y), H(Y|X)$, $H(X,Y)$, $H(Y) - H(Y|X)$ ed $I(X \wedge Y)$.

Esercizio A.9. Dimostrare che per ogni coppia di variabili aleatorie X, Y, definite su due insiemi $\mathcal{X}$, $\mathcal{Y}$, si ha $H(X,Y) \geq H(X)$.
(Suggerimento: applicare la definizione, quindi usare la regola di Bayes.)

Esercizio A.10. Dimostrare la regola della catena, ovvero:

$$H(X_1, \ldots, X_n) = H(X_1) + \sum_{i=2}^{n} H(X_i | X_1, \ldots, X_{i+1}).$$

(Suggerimento: usare ricorsivamente l'uguaglianza $H(A,B) = H(A) + H(B|A)$, cominciando dal caso $A = (X_1, \ldots, X_{n-1})$, $B = X_n$.)

Esercizio A.11. Questo esercizio mostra che non è possibile stabilire un ordinamento generale per le quantità $I(X \wedge Y)$ ed $I(X \wedge Y | Z)$.

1. Sia $X = Y = Z$, con $H(X) > 0$. Mostrare che $I(X \wedge Y) - I(X \wedge Y | Z) > 0$.
2. Siano X, Y variabili aleatorie binarie indipendenti ed uniformemente distribuite, e $Z = X \oplus Y$. Mostrare che $I(X \wedge Y) - I(X \wedge Y | Z) < 0$.

Esercizio A.12. Sia X una variabile aleatoria su un insieme $\mathcal{X}$. Mostrare che per ogni scelta della distribuzione di probabilità P_X, risulta $\mathbf{H}_\infty(X) \leq \log(\#\mathcal{X})$.

Letture consigliate

[Bil79] Patrick Billingsley. *Probability and Measure*. John Wiley e Sons, 1979.

[CK97] Imre Csiszár e János Körner. *Information Theory: Coding Theorems for Discrete Memoryless Systems*. Second. Akademiai Kiado, 1997.

[CT06] Thomas M. Cover e Joy A. Thomas. *Elements of Information Theory*. Second. New York: Wiley-Interscience, 2006.

[Gut05] Allan Gut. *Probability: A Graduate Course*. Springer-Verlang, 2005.

[Rén61] Alfréd Rényi. "On Measures of Information and Entropy". In: *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*. 1961, pp. 547–561.

[Sha48] Claude E. Shannon. "A mathematical theory of communication". In: *Bell Sys. Tech. J.* 27 (1948), pp. 623–656.

[Tij04] Henk Tijms. *Understanding Probability: Chance Rules in Everyday Life*. Cambridge University Press, 2004.

APPENDICE B

Teoria dei numeri

Cubum autem in duos cubos, aut quadratoquadratum in duos quadratoquadratos, et generaliter nullam in infinitum ultra quadratum potestatem in duos eiusdem nominis fas est dividere cuius rei demonstrationem mirabilem sane detexi. Hanc marginis exiguitas non caperet.

Pierre de Fermat (ai margini di una copia dell'Arithmetica di Diofanto)

La teoria dei numeri riguarda lo studio delle proprietà dei numeri interi. La formulazione di molti dei problemi tipici della materia può essere facilmente compresa anche da un non-matematico; tuttavia la soluzione degli stessi problemi è spesso complessa, e richiede tecniche non banali.

La teoria dei numeri nasce nell'antica Grecia, con lo studio dei criteri di divisibilità degli interi (da parte di Euclide ad esempio). Più avanti (nel XVII secolo) fu Pierre de Fermat ad iniziare uno studio più sistematico. Il matematico Francese, in particolare, enunciò numerose congetture, senza fornire per esse una dimostrazione.[90] I contributi successivi, ad opera di Eulero, Lagrange e Gauss, hanno quindi dato inizio alla teoria dei numeri moderna.

Guida per il lettore. Questa appendice richiama alcuni risultati di base in teoria dei numeri, usati in diverse parti del testo. Nel Paragrafo B.1 studieremo i criteri di divisibilità e l'algoritmo euclideo di divisione. Compiremo quindi un viaggio nel mondo dell'aritmetica modulare, studiando le congruenze (nel Paragrafo B.2) e le proprietà dei residui quadratici (nel Paragrafo B.3). nel Paragrafo B.4 ci soffermeremo, infine, sullo studio delle curve ellittiche su un campo finito, che hanno diverse applicazioni in crittografia.

Per un'introduzione più rigorosa alla teoria "classica" dei numeri, si rimanda a [JJ05].

[90] La più famosa di queste è senz'altro il famoso "ultimo teorema", che afferma che non esistono soluzioni intere all'equazione $X^n + Y^n = Z^n$ per $n > 2$. La soluzione dell'enigma di Fermat ha visto impegnate, senza fortuna, intere generazioni di matematici. La congettura è stata dimostrata da Andrew Wiles [Wil95], nel 1994. L'avvincente storia della congettura è raccontata in [Sin97].

Venturi D.: Crittografia nel Paese delle Meraviglie
DOI 10.1007/978-88-470-2481-6_16,

B.1 Algoritmo euclideo

Cominciamo introducendo i concetti fondamentali di quoziente e resto di una divisione.

Lemma B.1 (Quoziente e resto). *Se a e b sono interi, con $b > 0$, allora esiste un'unica coppia di interi (q, r) tali che $a = qb + r$ e $0 \leq r < b$.*

Dimostrazione. La dimostrazione è costituita da due parti: nella prima parte mostreremo che la coppia (q, r) esiste e nella seconda parte che essa è unica.

Definiamo l'insieme $\mathcal{S} = \{a - nb : n \in \mathbb{Z}\}$. Affermiamo che $\mathcal{S}$ contiene almeno un elemento non-negativo. Infatti se $a > 0$, allora possiamo scegliere $n = 0$ e $a \in \mathcal{S}$. Se invece $a < 0$, possiamo scegliere $n = a$ da cui $a - ab = a(1 - b) \in \mathcal{S}$. Siccome per ipotesi $b > 0$, abbiamo che $(1 - b)$ è negativo e quindi $a(1 - b)$ è un numero positivo oppure nullo (quando $b = 1$). In entrambi i casi comunque $a - ab \in \mathcal{S}$ è non-negativo.

Sia dunque r il minimo intero non negativo contenuto in $\mathcal{S}$. Osserviamo che $r = a - qb$ per un qualche $q \in \mathbb{Z}$, da cui $a = r + qb$. Resta da mostrare che $0 \leq r < b$. Siccome $r \in \mathcal{S}$ è non-negativo per definizione, dobbiamo solamente mostrare che $r < b$. Supponiamo che $r \geq b$, allora:

$$r' = a - (q + 1)b = a - qb - b = r - b \geq 0,$$

contraddicendo il fatto che r sia il minimo elemento non-negativo di $\mathcal{S}$.

Per mostrare che la coppia (q, r) è unica, supponiamo che esistano due coppie distinte (q, r) e (q', r') tali che $a = qb + r$ ed $a = q'b + r'$. Segue $r - r' = b(q' - q)$, ovvero b divide $r - r'$. Siccome però $0 \leq r, r' < b$, deve essere $-b < -r' \leq r - r' < r < b$; pertanto il divisore b è più grande (in valore assoluto) del numero $r - r'$ che divide, il che è possibile solo se $r - r' = 0$ ovvero $r = r'$.

Sostituendo in $qb + r = q'b + r'$, troviamo $qb = q'b$ da cui segue $q = q'$. □

Il resto della divisione è indicato con $a \bmod b = r$. Quando $r = 0$, si è soliti dire che b divide a (oppure che b è un divisore di a) e si scrive $b|a$. Un divisore comune di a e b è un intero che divide sia a che b. Si definisce massimo comun divisore (*greatest common divisor*) di a e b — indicato con $\gcd(a, b)$ — il massimo tra i divisori comuni di a e b. (La definizione è ben posta perché l'insieme dei divisori comuni ad a e b è non vuoto (1 ne fa sempre parte) ed è finito; quindi ha senso parlare di massimo.) Quando $\gcd(a, b) = 1$ si dice che a e b sono *coprimi*.

Descriviamo ora l'algoritmo euclideo per il calcolo del massimo comun divisore tra a e b. L'ingrediente fondamentale è il seguente:

Lemma B.2 (Algoritmo di Euclide). *Siano a e b due interi tali che $a \geq b > 0$. Allora* $\gcd(a, b) = \gcd(b, a \bmod b)$.

Dimostrazione. È sufficiente mostrare che i divisori comuni ad a e b sono gli stessi comuni a b ed $a \bmod b$. Per il Lemma B.1 possiamo sempre scrivere $a = qb + a \bmod b$, dove $q = \left\lfloor \frac{a}{b} \right\rfloor$ e $a \bmod b$ è il resto della divisione tra a e b. Pertanto, un divisore comune ad a e b divide anche $a - qb = a \bmod b$.

D'altra parte, un divisore comune di b ed $a \bmod b$ divide anche $a = qb + a \bmod b$; quindi l'asserto. □

L'algoritmo euclideo sfrutta il lemma precedente in modo iterativo. In primo luogo se $b = 0$ abbiamo immediatamente $\gcd(a, 0) = 0$. Se invece $b \neq 0$ possiamo usare il Lemma B.2 e concludere $\gcd(a, b) = \gcd(b, a \bmod b)$, con $b > a \bmod b \geq 0$, in quanto $a \bmod b$ è il resto della divisione tra a e b. A questo punto applichiamo ricorsivamente il Lemma B.2 agli elementi $(b, a \bmod b)$: ad ogni passo il secondo termine del gcd diminuisce fino a diventare 0. A quel punto:

$$\gcd(a, b) = \gcd(b, a \bmod b) = \cdots = \gcd(d, 0) = d,$$

ed abbiamo calcolato $d = \gcd(a, b)$.

Un esempio chiarisce immediatamente le idee. Sia $a = 185$ e $b = 40$. Applicando iterativamente il Lemma B.2, troviamo:

$$\begin{aligned} \gcd(185, 40) &= \gcd(40, 25) = \gcd(25, 15) = \gcd(15, 10) \\ &= \gcd(10, 5) = \gcd(5, 0) = 5. \end{aligned}$$

Complessità. Vogliamo ora mostrare che l'algoritmo di Euclide è efficiente. Per fare questo dobbiamo innanzitutto discutere la complessità computazionale associata alle operazioni elementari di somma, prodotto e divisione. Sia $O(1)$ il costo relativo alla somma di due bit ed indichiamo con s_a la lunghezza di a (in binario), ovvero $s_a = \lfloor \log a \rfloor + 1$. (Un'espressione identica vale per s_b relativamente a b.) È banale verificare che (cf. Esercizio B.1):

- somma e sottrazione hanno complessità $O(\max\{s_a, s_b\})$;
- la moltiplicazione costa $O(s_a \cdot s_b)$;
- la divisione costa $O(s_q \cdot s_b)$, essendo q è il quoziente della divisione tra a e b.

Osserviamo inoltre che lo spazio richiesto in memoria è $O(s_a + s_b)$.

Per poter valutare la complessità dell'algoritmo di Euclide, dobbiamo in qualche modo limitare il numero di iterazioni necessarie fino al termine dell'algoritmo stesso. Per fare ciò ci serviremo dei numeri di Fibonacci, introdotti per la prima volta dal matematico italiano Leonardo Fibonacci (nel XIII secolo) per studiare la velocità di riproduzione di una famiglia di conigli.

Definizione B.1 (Numeri di Fibonacci). La sequenza di interi $\{F_n\}_{n=0}^{+\infty}$, definita dalle relazioni $F_0 = 0$, $F_1 = 1$, $F_{n+2} = F_n + F_{n+1}$, è detta sequenza di Fibonacci. ■

I primi elementi della sequenza di Fibonacci sono $0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$. La sequenza soddisfa alcune proprietà interessanti:

Lemma B.3 (Proprietà della sequenza di Fibonacci). *Sia* $\Theta = (1+\sqrt{5})/2$ *il rapporto aureo. Per ogni numero naturale* $n \in \mathbb{N}$ *è valido quanto segue:*

$$(i)\Theta^{n+2} = \Theta^n + \Theta^{n+1} \qquad (ii)(1-\Theta)^{n+2} = (1-\Theta)^n + (1-\Theta)^{n+1}$$

$$(iii)F_n = \frac{1}{\sqrt{5}}\left(\Theta^n - (1-\Theta)^n\right) \qquad (iv)F_{n+1} < \Theta^n < F_{n+2}.$$

Dimostrazione. Per dimostrare (i) e (ii) basta osservare che Θ e $(1-\Theta)$ sono soluzione dell'equazione di secondo grado $X^2 = X + 1$. Pertanto $\Theta^2 = \Theta + 1$ e $(1-\Theta)^2 = (1-\Theta) + 1$. L'asserto segue moltiplicando ambo i membri per Θ^n ed $(1-\Theta)^2$ (rispettivamente).

La (iii) è anche detta formula di Binet (perché mostrata indipendentemente da Binet). Dimostreremo l'espressione per induzione su n. Il caso base si ha in corrispondenza dei valori F_0 ed F_1, ovvero quando $n = 0$ ed $n = 1$:

$$F_0 = \frac{1}{\sqrt{5}}(1-1) = 0 \qquad F_1 = \frac{1}{\sqrt{5}}(\Theta - 1 + \Theta).$$

Assumiamo ora che la formula sia valida per ogni intero $< n$ e mostriamo che ciò implica che essa sia valida anche per n:

$$\begin{aligned} F_n = F_{n-1} + F_{n-2} &= \frac{1}{\sqrt{5}}(\Theta^{n-1} - (1-\Theta)^{n-1} + \Theta^{n-2} - (1-\Theta)^{n-2}) \\ &= \frac{1}{\sqrt{5}}(\underbrace{\Theta^{n-1} + \Theta^{n-2}}_{\Theta^n} - \underbrace{(1-\Theta)^{n-1} + (1-\Theta)^{n-2}}_{(1-\Theta)^n}) \\ &= \frac{1}{\sqrt{5}}(\Theta^n - (1-\Theta)^n). \end{aligned}$$

Dimostreremo la (iv) sempre per induzione su n. Quando $n = 1$ abbiamo in effetti $F_1 = 1 < \Theta < F_2 = 2$. Assumendo che l'espressione sia verificata per ogni intero $< n$, possiamo scrivere:

$$F_{n-1} < \Theta^{n-2} < F_n \qquad F_n < \Theta^{n-1} < F_{n+1}.$$

Ma allora:

$$F_{n-1} + F_n = F_{n+1} < \Theta^{n-2} + \Theta^{n-1} = \Theta^n < F_n + F_{n+1} = F_{n+2},$$

come volevasi dimostrare. □

Il seguente teorema, collega il numero di iterazioni dell'algoritmo di Euclide con la sequenza di Fibonacci.

Teorema B.4 (Teorema di Lamè). *Sia $a \geq b > 0$ ed indichiamo con $E(a, b)$ il numero di passi nell'algoritmo euclideo. Allora per ogni numero naturale $n \in \mathbb{N}$ abbiamo $E(a, b) < n$ ogni volta che $b < F_{n+1}$ oppure $a < F_{n+2}$.*

Dimostrazione. È sufficiente dimostrare che ogni volta in cui $E(a, b) \geq n$, deve aversi $b \geq F_{n+1}$ ed anche $a \geq F_{n+2}$. Procediamo per induzione su n. Nel caso banale in cui $n = 0$, l'asserto diventa:

$$E(a, b) \geq 0 \quad \Rightarrow \quad b \geq F_1 = 1 \text{ e } a \geq F_2 = 2,$$

il che è sicuramente vero in quanto per ipotesi $a \geq b > 0$.

Supponiamo ora che $E(a, b) \geq k$ implichi $b \geq F_{k+1}$ e $a \geq F_{k+2}$, per ogni intero $k \leq n$; mostreremo che ciò implica l'asserto per ogni $k \geq n + 1$. In altri termini dobbiamo dimostrare che se $E(a, b) \geq n + 1$, allora $a \geq F_{n+3}$ e $b \geq F_{n+2}$. Sia $E(a, b) \geq n + 1$; usando il Lemma B.2 possiamo scrivere $\gcd(a, b) = \gcd(b, a \bmod b)$ ed il calcolo di $\gcd(b, a \bmod b)$ richiede esattamente $E(a, b) - 1$ iterazioni. Siccome $E(b, a \bmod b) = E(a, b) - 1 \geq n$, possiamo usare l'ipotesi induttiva per concludere che $b \geq F_{n+2}$ e $a \bmod b \geq F_{n+1}$. Pertanto resta da mostrare che $a \geq F_{n+3}$; ma $a = qb + a \bmod b \geq b + a \bmod b$, da cui $a \geq b + a \bmod b \geq F_{n+2} + F_{n+1} = F_{n+3}$. Questo conclude la prova. □

Il Teorema di Lamè ci consente di calcolare la complessità dell'algoritmo euclideo.

Corollario B.5 (Complessità dell'algoritmo di Euclide). *Poniamo $n = \max\{a, b\}$. La complessità dell'algoritmo di Euclide è $O(\log^3 n)$.*

Dimostrazione. Sia $b = b_0 > 0$ fissato, e concentriamoci su $t = E(a, b_0)$. Il Teorema B.4 implica $a \geq F_{t+2}$. D'altra parte, per il Lemma B.3, sappiamo che $F_{t+2} > \Theta^t$ e quindi $a > \Theta^t$; pertanto:

$$\log_\Theta a > t \quad \Rightarrow \quad t < \log(a) < \log(n).$$

Siccome ogni iterazione dell'algoritmo prevede una divisione (costo $O(\log^2 n)$), la complessità è $O(\log^3 n)$. □

Versione estesa. Possiamo definire una versione estesa dell'algoritmo di Euclide, utile per trovare soluzioni intere ad equazioni della forma $aX + bY = \gcd(a, b)$. La seguente espressione, dovuta a Bézout, ci assicura che una soluzione esiste sempre:

Teorema B.6 (Identità di Bézout). *Se a e b sono interi (non entrambi nulli), allora esistono sempre due interi u, v tali che:*

$$\gcd(a, b) = a \cdot u + b \cdot v.$$

Dimostrazione. Si tratta semplicemente di scrivere le relazioni dell'algoritmo di Euclide alla rovescia. Sia $d = \gcd(a, b) = r_t$ l'ultimo resto non-nullo dell'algoritmo di Euclide (ovvero $r_{t+1} = 0$). Possiamo scrivere:

$$r_t = r_{t-2} - q_{t-1} r_{t-1},$$

esprimendo così d in funzione di r_{t-2} ed r_{t-1}. D'altra parte, al passo precedente dell'algoritmo, si deve avere:

$$r_{t-1} = r_{t-3} - q_{t-2} r_{t-2},$$

e se sostituiamo r_{t-1} nell'espressione per r_t, abbiamo espresso d in funzione di r_{t-2} e di r_{t-3}. Se proseguiamo a ritroso fino al primo passo dell'algoritmo, possiamo esprimere d come somma di un multiplo di $r_0 = a$ ed un multiplo di $r_1 = b$, cioè:

$$\gcd(a, b) = a \cdot u + b \cdot v.$$

□

Più in generale è possibile mostrare che ogni equazione del tipo $aX + bY = c$ ammette soluzione se e solo se c è un multiplo di $\gcd(a, b)$. È facile vedere che i valori di u e v non sono unici; comunque essi possono essere visti come una soluzione dell'equazione $aX + bY = \gcd(a, b)$. Per calcolare tale soluzione, si può usare l'algoritmo esteso di Euclide:

Lemma B.7 (Algoritmo di Euclide esteso). *Definiamo le quantità:*

$$\begin{aligned} x_{k+1} &= q_k x_k + x_{k-1} \qquad x_0 = 1,\ x_1 = 0 \\ y_{k+1} &= q_k x_k + y_{k-1} \qquad y_0 = 0,\ y_1 = 1, \end{aligned}$$

avendo indicato con q_k il k-simo quoziente dell'algoritmo di Euclide applicato agli interi $a \geq b > 0$ (per ogni $k = 1, \cdots, t$). Possiamo scrivere:

$$r_k = (-1)^k x_k a + (-1)^{k+1} y_k b \qquad k = 0, 1, \cdots, t,$$

dove r_k è il k-simo resto nell'algoritmo euclideo, dato da $r_{k-2} = r_{k-1} q_{k-1} + r_k$ per ogni $k = 2, 3, \cdots, t$. (Al solito $r_0 = a$, $r_1 = b$ ed $r_t = d = \gcd(a, b)$.)

Dimostrazione. Per induzione su k. Il caso base ($k = 0$ e $k = 1$) è verificato, in quanto:

$$\begin{aligned} k = 0 \quad &\Rightarrow \quad r_0 = (-1)^0 x_0 a + (-1)^1 y_0 b = a \\ k = 1 \quad &\Rightarrow \quad r_1 = (-1)^1 x_1 a + (-1)^2 y_1 b = b. \end{aligned}$$

Supponiamo ora che l'asserto sia valido per ogni intero $< k$ e mostriamo che questo ne implica la validità per gli interi $\geq k$. Usando l'ipotesi induttiva possiamo concludere:

$$\begin{aligned} r_k &= r_{k-2} - r_{k-1} q_{k-1} \\ &= (-1)^{k-2} x_{k-2} a + (-1)^{k-1} y_{k-2} b - q_{k-1} \left[(-1)^{k-1} x_{k-1} a + (-1)^k y_{k-1} b \right] \\ &= (-1)^k a (x_{k-2} + q_{k-1} x_{k-1}) + (-1)^{k+1} b (y_{k-2} + q_{k-1} y_{k-1}) \\ &= (-1)^k x_k a + (-1)^{k+1} y_k b. \end{aligned}$$

□

Siccome in particolare

$$r_t = \gcd(a, b) = (-1)^t x_t a + (-1)^{t+1} y_t b,$$

segue che $x = (-1)^t x_t$ ed $y = (-1)^{t+1} y_t$ sono soluzioni di $aX + bY = \gcd(a, b)$.

Supponiamo ad esempio di voler risolvere $185X + 40Y = 5$. Applichiamo prima l'algoritmo di Euclide ad $a = 185$ e $b = 40$, ottenendo le relazioni

$$\begin{aligned} 185 = 40 \cdot 4 + 25 \quad &\Rightarrow \quad q_1 = 4, r_2 = 25 \\ 40 = 25 \cdot 1 + 15 \quad &\Rightarrow \quad q_2 = 1, r_3 = 15 \\ 25 = 15 \cdot 1 + 10 \quad &\Rightarrow \quad q_3 = 1, r_4 = 10 \\ 15 = 10 \cdot 1 + 5 \quad &\Rightarrow \quad q_4 = 1, r_5 = 5 \\ 5 = 10 \cdot 2 + 0 \quad &\Rightarrow \quad q_5 = 2, r_6 = 0. \end{aligned}$$

Quindi $t = 5$ e $\gcd(a, b) = r_t = 5$. È banale verificare, inoltre, che $x_5 = 3$ ed $y_5 = 14$. Pertanto:

$$\gcd(185, 40) = r_t = 5 = (-1)^5 \cdot 3 \cdot 185 + (-1)^6 \cdot 14 \cdot 40,$$

il che implica $x = -3$ ed $y = 14$. Non è complesso mostrare che anche la versione estesa ha complessità polinomiale.

B.2 L'aritmetica dell'orologio

In questo paragrafo studieremo i concetti di base dell'aritmetica modulare (detta anche aritmetica dell'orologio poiché su tale principi si basa il calcolo delle ore in cicli di 12 o 24). Per fare ciò ci occorrono alcune nozioni basilari di algebra, in particolare gruppi, anelli e campi.

Definizione B.2 (Gruppo). Un gruppo $(\mathbb{G}, +)$ è un insieme di elementi con un'operazione, per i quali sono verificati i seguenti assiomi:

- *Chiusura.* $\forall g_1, g_2 \in \mathbb{G}$ si ha $g_1 + g_2 \in \mathbb{G}$.
- *Associatività.* $\forall g_1, g_2, g_3 \in \mathbb{G}$ si ha $g_1 + (g_2 + g_3) = (g_1 + g_2) + g_3$.
- *Commutatività.* $\forall g_1, g_2 \in \mathbb{G}$ si ha $g_1 + g_2 = g_2 + g_1$.
- *Elemento neutro.* Esiste un elemento neutro $e \in \mathbb{G}$, tale che $\forall g \in \mathbb{G}$ si ha $g + e = g$.
- *Elemento inverso.* $\forall g \in \mathbb{G}$ esiste ed è unico un elemento inverso $g' \in \mathbb{G}$ tale che $g + g' = e$. ■

Quando vale la proprietà commutativa, il gruppo[91] si dice *abeliano* (in onore del matematico norvegese Niels Henrik Abel). Si definisce *ordine* del gruppo la sua cardinalità $\#\mathbb{G}$. Inoltre diremo che $(\mathbb{H}, +)$ è un *sottogruppo* di $(\mathbb{G}, +)$ se $\mathbb{H}$ è contenuto in $\mathbb{G}$ e se $(\mathbb{H}, +)$ è esso stesso un gruppo.

Definizione B.3 (Anello). Un anello $(\mathbb{A}, +, \cdot)$ è un insieme di elementi con due operazioni, per i quali sono verificati i seguenti assiomi:

- $(\mathbb{A}, +)$ è un gruppo abeliano con elemento neutro e.

[91] Solitamente quando l'operazione è $+$ l'elemento neutro è indicato con 0, l'inverso di g con $-g$ e si scrive ng per $g + \ldots + g$ (n volte).

- *Associatività del* $\cdot$. $\forall a_1, a_2, a_3 \in \mathbb{A}$ si ha $a_1 \cdot (a_2 \cdot a_3) = (a_1 \cdot a_2) \cdot a_3$.
- *Distributività del* $\cdot$ *rispetto al* $+$. $\forall a_1, a_2, a_3 \in \mathbb{A}$ si ha $a_1 \cdot (a_2 + a_3) = a_1 \cdot a_2 + a_1 \cdot a_3$. ■

Quando l'operazione $\cdot$ soddisfa anche la proprietà commutativa, si parla di *anello commutativo*, mentre diremo che $\mathbb{A}$ è *unitario* se esiste l'elemento 1 tale che $\forall a \in \mathbb{A}$ si abbia $a \cdot 1 = a$. Se infine l'anello è commutativo e non esistono divisori dello zero — ovvero se $a \cdot b = 0$ implica che $a = 0$ oppure $b = 0$ — si parla di *dominio d'integrità*.

Definizione B.4 (Campo). Un campo $(\mathbb{K}, +, \cdot)$ è un insieme di elementi con due operazioni, per i quali sono verificati i seguenti assiomi:

- $(\mathbb{K}, +, \cdot)$ è un anello unitario commutativo, con elemento neutro 1.
- *Inverso del* $\cdot$. $\forall x \in \mathbb{K}$ esiste ed è unico $x^{-1} \in \mathbb{K}$ tale che $x \cdot x^{-1} = 1$. ■

Ad esempio l'insieme $\mathbb{Z}$ degli interi è un gruppo rispetto alla somma, ma non lo è rispetto al prodotto (in quanto non sempre c'è un inverso). L'insieme $\mathbb{R} \setminus \{0\}$ di numeri reali (escluso lo 0) è un gruppo rispetto al prodotto.

Il seguente risultato, fondamentale in algebra, è dovuto a Lagrange.

Teorema B.8 (Teorema di Lagrange). *Sia* $\mathbb{H}$ *un sottogruppo di un gruppo finito* $\mathbb{G}$. *Allora* $\#\mathbb{H}$ *divide* $\#\mathbb{G}$.

Si veda ad esempio [Lan02, Proposizione 2.2] per una dimostrazione.

Congruenze e classi di residui. Possiamo ora dare la nozione di *congruenza*. Diremo che gli interi a e b sono congruenti modulo n — indicato con $a \equiv b \pmod{n}$ — se n divide $a - b$. Ad esempio $-2 \equiv 19 \pmod{21}$, $6 \equiv 0 \pmod{2}$ e $2^{340} \equiv 1 \pmod{341}$.

Le congruenze modulo n definiscono una *relazione d'equivalenza*[92] su $\mathbb{Z}$. In questo modo $\mathbb{Z}$ è ripartito in *classi d'equivalenza* contenenti gli interi tra loro

[92] Una relazione ϱ tra elementi di un insieme $\mathcal{S}$ è detta di equivalenza se è:

- *Riflessiva.* Per ogni elemento s di $\mathcal{S}$ si ha $s \varrho s$.
- *Simmetrica.* Per ogni coppia (s_1, s_2) di $\mathcal{S}$ si ha $s_1 \varrho s_2 \Rightarrow s_2 \varrho s_1$.
- *Transitiva.* Per ogni terna (s_1, s_2, s_3) di $\mathcal{S}$ si ha $s_1 \varrho s_2$ e $s_2 \varrho s_3 \Rightarrow s_1 \varrho s_3$.

Un sottoinsieme di $\mathcal{S}$ che contiene tutti e soli gli elementi equivalenti a un qualche elemento s di $\mathcal{S}$ prende il nome di classe di equivalenza di s. L'insieme delle classi di equivalenza su $\mathcal{S}$ si chiama insieme quoziente di $\mathcal{S}$ per la relazione ϱ e viene talvolta indicato con l'espressione $\mathcal{S}/\varrho$.

congruenti modulo n, ovvero gli interi che hanno lo stesso resto quando sono divisi per n. Ciò definisce l'insieme di residui modulo n, indicati con $\mathbb{Z}_n = \mathbb{Z}/n\mathbb{Z}$. Ad esempio per $n = 2$, abbiamo $\mathbb{Z}_2 = \{0, 1\}$, in quanto tutti gli interi sono congrui a 0 (se sono pari) oppure ad 1 (se sono dispari) modulo 2. (In altri termini le due classi d'equivalenza sono qui i numeri pari e i numeri dispari.) Quando invece $n = 3$, abbiamo $\mathbb{Z}_3 = \{0, 1, 2\}$ in quanto tutti gli interi n sono congrui a 0 (se $n/3$ ha resto 0), oppure ad 1 (se $n/3$ ha resto 1) oppure a 2 (se $n/3$ ha resto 2), modulo 3.

È facile verificare che $(\mathbb{Z}_n, +)$ è un gruppo rispetto all'operazione di somma modulo n; inoltre $(\mathbb{Z}_n, +, \cdot)$ è un anello commutativo. In generale invece $\mathbb{Z}_n$ non è un campo, in quanto non tutti gli elementi $a \in \mathbb{Z}_n$ sono invertibili:

Lemma B.9 (Elementi invertibili in $\mathbb{Z}_n$). *Se* $\gcd(a, n) > 1$ *allora* $a \in \mathbb{Z}_n$ *non è invertibile modulo* n.

Dimostrazione. Supponiamo per assurdo che a sia invertibile, ovvero esiste $b \in \mathbb{Z}_n$ con $ab \equiv 1 \pmod{n}$. Allora:

$$ab = 1 + nk \quad \Rightarrow \quad ab - nk = 1,$$

per qualche intero k. Evidentemente $\gcd(a, n)$ divide $ab - nk$ e quindi deve anche dividere 1, da cui $\gcd(a, n) = 1$ contro l'ipotesi. □

Si definisce con $\mathbb{Z}_n^*$ l'insieme dei residui invertibili modulo n. Applicando il Lemma B.9, sappiamo che:

$$\mathbb{Z}_n^* = \{a \in \mathbb{Z}_n : \ \gcd(a, n) = 1\}.$$

Notare che la cardinalità di $\mathbb{Z}_n^*$ è data dal numero di elementi minori di n e coprimi con esso.

Definizione B.5 (Funzione toziente di Eulero). Si definisce la funzione toziente di Eulero $\varphi(n)$ come il numero di elementi minori di n e coprimi con esso. ■

Pertanto $\#\mathbb{Z}_n^* = \varphi(n)$. Osserviamo che se $n = p$ è un primo, allora $\varphi(p) = p - 1$; si può dimostrare invece che se $n = p_1 \cdot p_2$, allora $\varphi(n) = \varphi(p_1) \cdot \varphi(p_2)$.

È immediato verificare che $(\mathbb{Z}_n, +, \cdot)$ è un campo se e solo se n è primo. Infatti se n è primo, ogni elemento tranne lo zero è invertibile. D'altra parte se $\mathbb{Z}_n$ è un campo, ogni elemento $< n$ è coprimo con n e quindi n deve essere primo. L'inverso di $a \in \mathbb{Z}_n^*$ può essere calcolato come soluzione dell'equazione $aX + nY = 1$ usando l'algoritmo di Euclide esteso (cf. Lemma B.7).

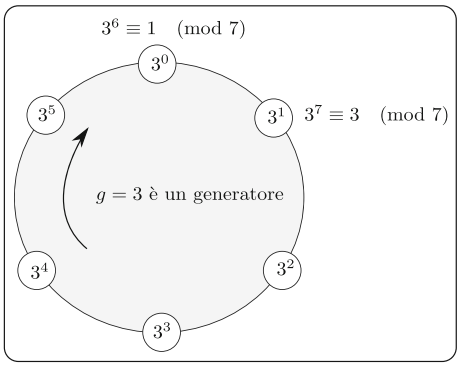


Fig. B.1. Il gruppo ciclico $\mathbb{Z}_7^*$ con il generatore $g = 3$

Sia ora p un primo e consideriamo il *gruppo unitario*:

$$\mathbb{Z}_p^* = \{1, 2, \ldots, p-1\} \quad \text{con} \quad \#\mathbb{Z}_p^* = \varphi(p) = p - 1,$$

ovvero il sottogruppo moltiplicativo di $\mathbb{Z}_p$ contenente tutti gli elementi invertibili modulo p (cioè tutti tranne lo zero). A partire da $g \in \mathbb{Z}_p^*$ calcoliamo le potenze successive di g: $g^0, g^1, g^2, \ldots$; siccome $\mathbb{Z}_p^*$ è finito esistono i, j (con $i \neq j$, diciamo $j < i$) tali che $g^i \equiv g^j \pmod p$. Pertanto $g^{i-j} \equiv 1 \pmod p$, ovvero esiste q tale che $g^q \equiv 1 \pmod p$. Si definisce *ordine* di g *il più piccolo* intero $\mathrm{ord}(g)$ tale che $g^{\mathrm{ord}(g)} \equiv 1 \pmod p$. Un elemento $g \in \mathbb{Z}_p^*$ di ordine $\mathrm{ord}(g) = k$ genera un sottogruppo $\mathbb{G} = \{g^0, g^1, g^2, \ldots, g^{k-1}\}$ di $\mathbb{Z}_p^*$, quindi possiamo concludere $\mathrm{ord}(g) \leq p - 1$. L'elemento g è detto *generatore* del sottogruppo $\mathbb{G}$.

Un elemento di ordine $p-1$ genera tutto $\mathbb{Z}_p^*$ ed è detto *primitivo*. Osserviamo che se g è un elemento primitivo allora $\mathbb{Z}_p^* = \{g^0, g^1, \ldots, g^{p-2}\}$; un gruppo generato dalle potenze successive di un suo elemento è detto *ciclico*. Si può dimostrare che $\mathbb{Z}_p^*$ è sempre ciclico, con $\varphi(p-1)$ elementi primitivi. Ad esempio l'insieme $\mathbb{Z}_7^* = \{1, 2, 3, 4, 5, 6\}$ è ciclico con $\varphi(6) = \varphi(3) \cdot \varphi(2) = 6$ elementi primitivi (cf. Fig. B.1).

A volte è utile calcolare l'ordine di un elemento g appartenente ad un gruppo $\mathbb{G}$ con $\#\mathbb{G} = q$ elementi. Sia $q = p_1^{e_1} \cdot \cdots \cdot p_k^{e_k}$ la scomposizione in fattori primi di q. Per ogni divisore primo p_i di q, sia f_i il massimo intero tale che $g^{q/p_i^{f_i}} = 1$. Allora, l'ordine di g è:

$$\mathrm{ord}(g) = p_1^{e_1 - f_1} \cdot \cdots \cdot p_k^{e_k - f_k}.$$

Se ad esempio prendiamo $\mathbb{Z}_{101}^*$ e $g = 2$, si ha $q = 100 = 2^2 \cdot 5^2$. Poiché $2^{50} \equiv -1 \pmod{101}$ e $2^{20} \equiv -6 \pmod{101}$ si ha $f_1 = 0$ ed $f_2 = 0$; quindi $\mathrm{ord}(2) = 100$ in $\mathbb{Z}_{101}^*$ (ovvero $\mathbb{Z}_p^*$ è ciclico e 2 è un elemento primitivo).

È anche facile verificare quando un certo intero k è l'ordine di un elemento $g \in \mathbb{G}$. Ciò si verifica quando $g^k \equiv 1$ e $g^{k/p} \neq 1$ in $\mathbb{G}$, per ogni divisore primo p di k. Ad esempio 25 è l'ordine di 5 in $\mathbb{Z}_{101}^*$, poiché $5^{25} \equiv 1 \pmod{101}$, ma $5^5 \not\equiv -6 \pmod{101}$.

Alcuni teoremi fondamentali. Concludiamo il paragrafo con la dimostrazione di alcuni teoremi fondamentali nelle applicazioni crittografiche della teoria dei numeri.

Teorema B.10 (Teorema di Eulero). *Sia n un intero positivo. Allora per ogni $a \in \mathbb{Z}_n^*$ si ha $a^{\varphi(n)} \equiv 1 \pmod{n}$.*

Dimostrazione. Consideriamo l'insieme $\mathbb{Z}_n^* = \{x_1, \ldots, x_{\varphi(n)}\}$. Definiamo gli elementi $y_i = a \cdot x_i \bmod n$; è facile mostrare che questi non sono altro che gli elementi di $\mathbb{Z}_n^*$ in ordine diverso (ovvero permutati). Sia inoltre $M = y_1 \cdot \cdots \cdot y_{\varphi(n)}$; notare che M è coprimo con n, quindi è invertibile modulo n. Pertanto

$$\begin{aligned} M \equiv y_1 \cdot \cdots \cdot y_{\varphi(n)} = a x_1 \cdot \cdots \cdot a x_{\varphi(n)} = a^{\varphi(n)} M \pmod{n} \\ \Rightarrow \quad a^{\varphi(n)} \equiv 1 \pmod{n}. \end{aligned}$$

□

Un corollario del teorema di Eulero è il piccolo teorema di Fermat, dimostrato indipendentemente da Fermat.

Teorema B.11 (Piccolo teorema di Fermat). *Sia p un primo. Allora per ogni $a \in \mathbb{Z}_p^*$ si ha $a^{p-1} \equiv 1 \pmod{p}$.*

Dimostrazione. Banale se sostituiamo $n = p$ (per un primo p) nel Teorema di Eulero. □

Il teorema del resto cinese (*Chinese Remainder Theorem*, CRT) studia alcuni sistemi di congruenze.

Teorema B.12 (Teorema del resto cinese). *Siano $m_1, \ldots, m_k$ interi a due a due coprimi (ovvero $\gcd(m_i, m_j) = 1$ per ogni $i \neq j$). Allora, per ogni insieme di valori $(a_1, \ldots, a_k)$, esiste un intero x soluzione del sistema di congruenze:*

$$x \equiv a_i \pmod{m_i} \quad per \quad i = 1, \ldots, k.$$

Tale soluzione è unica modulo $\prod_{i=1}^{k} m_i$.

Dimostrazione. Sia $M_i = \prod_{j \neq i} m_j$. Siccome gli m_i sono a due a due coprimi, $\gcd(M_i, m_i) = 1$. Usando l'algoritmo di Euclide esteso possiamo quindi calcolare gli interi (x_i, y_i) tali che:

$$y_i M_i + x_i m_i \equiv \gcd(M_i, m_i) = 1 \pmod{m_i}.$$

Per tali valori risulta ovviamente $y_i M_i \equiv 1 \pmod{m_i}$. Sia allora:

$$x \equiv \sum_{i=1}^{k} a_i y_i M_i \pmod{m} \quad \text{con} \quad m = \prod_{i=1}^{k} m_i.$$

È facile verificare che tale x è il valore cercato. Infatti $x \equiv a_i \pmod{m_i}$ in quanto $a_i y_i M_i \equiv a_i \pmod{m_i}$ ed $a_j y_j M_j \equiv 0 \pmod{m_i}$ per ogni $j \neq i$. Tale soluzione è inoltre unica; siano infatti x ed x' due soluzioni. Allora $x \equiv x' \pmod{m_i}$, per ogni $i = 1, 2, \ldots k$. Ma poiché gli m_i sono coprimi, $x - x'$ deve essere multiplo di $m = \prod_{i=1}^{k} m_i$, ovvero $x \equiv x' \pmod{m}$. □

Consideriamo ad esempio le seguenti congruenze:

$$x \equiv 2 \pmod 4 \quad x \equiv 1 \pmod 3 \quad x \equiv 0 \pmod 5.$$

Abbiamo $m_1 = 4$, $m_2 = 3$, $m_3 = 5$, $a_1 = 2$, $a_2 = 1$ ed $a_3 = 0$. Quindi $m = 60$, $M_1 = 15$, $M_2 = 20$, $M_3 = 12$. È facile verificare che le soluzioni delle congruenze $y_1 M_1 \equiv 1 \pmod{m_1}$, $y_2 M_2 \equiv 1 \pmod{m_2}$ ed $y_3 M_3 \equiv 1 \pmod{m_3}$, sono (rispettivamente) $y_1 = 3$, $y_2 = 2$ ed $y_3 = 3$. Pertanto

$$x \equiv (2 \cdot 3 \cdot 15 + 1 \cdot 2 \cdot 20 + 0 \cdot 3 \cdot 12) \equiv 130 \equiv 10 \pmod{60}.$$

Notare che il costo computazionale è $O\left(\log^2 m\right)$, l'occupazione di memoria $O(\log m)$.

Nella sua forma più generale il CRT caratterizza la struttura di alcuni anelli commutativi attraverso isomorfismi. Ricordiamo che un isomorfismo è una

mappa biettiva ψ, tale che sia ψ che ψ^{-1} sono omomorfismi. Consideriamo gli anelli $(\mathbb{A}_1, +, \cdot)$ ed $(\mathbb{A}_2, \boxplus, \boxdot)$. La mappa

$$\begin{aligned} \psi :\ & \mathbb{A}_1 \longrightarrow \mathbb{A}_2 \\ & x \in \mathbb{A}_1 \mapsto \psi(x) \in \mathbb{A}_2, \end{aligned}$$

è un omomorfismo di anelli se preserva la struttura, ovvero se soddisfa

$$\begin{aligned} \psi(x_1 + x_2) &= \psi(x_1) \boxplus \psi(x_2) \\ \psi(x_1 \cdot x_2) &= \psi(x_1) \boxdot \psi(x_2). \end{aligned}$$

(Una definizione identica vale per i gruppi e si parla di omomorfismo di gruppi.) Quando $\mathbb{A}_1$ ed $\mathbb{A}_2$ sono isomorfi, si scrive $\mathbb{A}_1 \simeq \mathbb{A}_2$.

Teorema B.13 (CRT generalizzato). *Siano $p_1, \ldots, p_n$ numeri primi tali che $\gcd(p_i, p_j) = 1$ per ogni $i \neq j$. Sia inoltre $n = \prod_{i=1}^{n} p_i$. Allora la mappa*

$$\begin{aligned} f :\ & \mathbb{Z}_n \longrightarrow \mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_n} \\ & x \mapsto (x \bmod p_1, \ldots, x \bmod p_n)\,, \end{aligned}$$

e la mappa

$$\begin{aligned} g :\ & \mathbb{Z}_n^* \longrightarrow \mathbb{Z}_{p_1}^* \times \cdots \times \mathbb{Z}_{p_n}^* \\ & x \mapsto (x \bmod p_1, \ldots, x \bmod p_n)\,, \end{aligned}$$

sono isomorfismi.

B.3 Residui quadratici

In questo paragrafo ci concentreremo sulle proprietà dei residui quadratici, definiti di seguito.

Residui quadratici modulo p. Diremo che $a \in \mathbb{Z}_p^*$ è un residuo quadratico (*quadratic residue*) modulo p, se esiste $b \in \mathbb{Z}_p^*$ tale che $a \equiv b^2 \pmod{p}$. L'insieme dei residui quadratici modulo p si indica con

$$\mathbb{QR}_p = \left\{ a \in \mathbb{Z}_p^* :\ a \equiv b^2 \bmod p, \text{ per qualche } b \in \mathbb{Z}_p^* \right\}.$$

Quante radici quadrate ha un residuo quadratico modulo p?

Lemma B.14 (Radici quadrate in $\mathbb{QR}_p$). *Ogni elemento $a \in \mathbb{QR}_p$ ha esattamente due radici quadrate modulo p.*

Dimostrazione. Sia $a \in \mathbb{QR}_p$. Allora $a \equiv b^2 \pmod{p}$ per qualche $b \in \mathbb{Z}_p^*$. Ovviamente $(-b)^2 = b^2 \equiv a \pmod{p}$. Inoltre $b \not\equiv -b \pmod{p}$ e quindi gli elementi $\pm b$ sono due radici quadrate di a e possiamo concludere che a ha *almeno* due radici quadrate modulo p.

Per vedere che queste sono uniche, sia b' un'altra radice di a. Allora $b^2 = a \equiv (b')^2 \pmod{p}$, da cui $b^2 - (b')^2 \equiv 0 \pmod{p}$. Fattorizzando il primo membro si ottiene $(b-b')(b+b') \equiv 0 \pmod{p}$, così che una delle due seguenti condizioni è verificata: (i) $p \mid (b-b')$, oppure (ii) $p \mid (b+b')$. Nel primo caso $b \equiv b' \pmod{p}$ e nel secondo caso $-b \equiv b' \pmod{p}$; pertanto $\pm b$ sono le uniche radici di a modulo p. □

Siccome l'elemento neutro del prodotto è $1 \in \mathbb{QR}_p$, e siccome quando $a, a' \in \mathbb{QR}_p$ anche $a \cdot a' \in \mathbb{QR}_p$, l'insieme dei residui quadratici modulo p è un sottogruppo di $(\mathbb{Z}_p^*, \cdot)$. L'insieme $\mathbb{QNR}_p = \mathbb{Z}_p^* \setminus \mathbb{QR}_p$ è detto insieme dei non-residui quadratici di $\mathbb{Z}_p^*$.

Consideriamo, ad esempio, $\mathbb{Z}_{11}^* = \{0, 1, 2, \ldots, 10\}$. È facile verificare che $\mathbb{QR}_{11} = \{1 = 1^2 = 10^2, 4 = 2^2 = 9^2, 9 = 3^2 = 8^2, 5 = 4^2 = 7^2, 3 = 5^2 = 6^2\}$ (cf. Fig. B.2).

Il seguente lemma determina la cardinalità di $\mathbb{QR}_p$.

Lemma B.15 (Cardinalità di $\mathbb{QR}_p$). *Per ogni primo p, si ha $\#\mathbb{QR}_p = (p-1)/2$.*

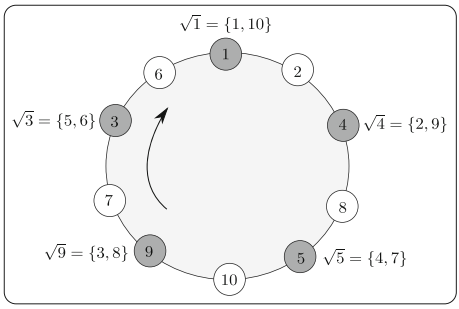


Fig. B.2. Il gruppo $\mathbb{QR}_{11}$

Dimostrazione. Sia g un generatore di $\mathbb{Z}_p^*$; abbiamo $\mathbb{QR}_p = \{g^2, g^4, \ldots, g^{p-1}\}$. Infatti, siccome ogni $x \in \mathbb{Z}_p^*$ è uguale a g^i per qualche i, possiamo concludere:

$$x^2 \equiv g^{2i} \pmod{p-1} \equiv g^j \pmod{p},$$

per un qualche j pari. □

Una conseguenza immediata è che $\#\mathbb{QNR}_p = p - (p-1)/2 = (p-1)/2$. Quindi esattamente metà degli elementi di $\mathbb{Z}_p^*$ è un residuo quadratico e l'altra metà è un non-residuo quadratico.

Possiamo anche verificare velocemente se un dato elemento è un residuo quadratico modulo p o meno.

Lemma B.16 (Criterio di Eulero). *Per ogni primo p, abbiamo che $a \in \mathbb{QR}_p$ se e solo se $a^{(p-1)/2} \equiv 1 \pmod{p}$.*

Dimostrazione. ($\Rightarrow$) Se $a \in \mathbb{QR}_p$ allora $a \equiv g^{2i} \pmod{p}$ per qualche i. Pertanto:

$$a^{\frac{p-1}{2}} \equiv \left(g^{2i}\right)^{\frac{p-1}{2}} \equiv g^{i(p-1)} \equiv 1 \pmod{p}.$$

($\Leftarrow$) Per assurdo. Supponiamo che $a \notin \mathbb{QR}_p$ con $a^{(p-1)/2} \equiv 1 \pmod{p}$. Dunque a è della forma $a \equiv g^{2i+1} \pmod{p}$ per qualche i. Pertanto:

$$a^{\frac{p-1}{2}} \equiv \left(g^{2i+1}\right)^{\frac{p-1}{2}} \equiv g^{i(p-1)} g^{\frac{p-1}{2}} \equiv g^{\frac{p-1}{2}} \not\equiv 1 \pmod{p},$$

(essendo g un generatore) contro l'ipotesi. □

Per distinguere i residui quadratici dai non-residui quadratici, si introduce il simbolo di Legendre.

Definizione B.6 (Simbolo di Legendre). Sia $a \in \mathbb{Z}_p^*$. Il simbolo di Legendre $J_p(a)$ è definito come:

$$J_p(a) = \begin{cases} +1 & \text{se } a \in \mathbb{QR}_p \\ -1 & \text{se } a \in \mathbb{QNR}_p. \end{cases}$$

■

Il criterio di Eulero può essere quindi riformulato come $J_p(a) \equiv a^{(p-1)/2} \pmod{p}$ per ogni $a \in \mathbb{Z}_p^*$.

Il simbolo di Legendre soddisfa la seguente proprietà moltiplicativa:

Lemma B.17 (Proprietà moltiplicativa del simbolo di Legendre). *Sia $p > 2$ un primo ed $a, b \in \mathbb{Z}_p^*$. Allora:*

$$J_p(a \cdot b) = J_p(a) \cdot J_p(b).$$

Dimostrazione. Usando il criterio di Eulero, possiamo scrivere:

$$J_p(a \cdot b) = (a \cdot b)^{\frac{p-1}{2}} \equiv a^{\frac{p-1}{2}} \cdot b^{\frac{p-1}{2}} = J_p(a) \cdot J_p(b) \pmod{p}.$$

Siccome $J_p(a \cdot b), J_p(a), J_p(b) \in \{\pm 1\}$, l'uguaglianza vale anche senza modulo. □

Un semplice corollario del Lemma B.17 è che se $a, a' \in \mathbb{QR}_p$ e $b, b' \in \mathbb{QNR}_p$, allora:

$$a \cdot a' \bmod p \in \mathbb{QR}_p \quad b \cdot b' \bmod p \in \mathbb{QR}_p \quad a \cdot b \bmod p \in \mathbb{QNR}_p.$$

A volte non vogliamo solamente poter verificare se un dato elemento è un residuo quadratico, ma (in caso affermativo) calcolarne esplicitamente una radice quadrata. In $\mathbb{QR}_p$ ciò è sempre possibile in modo efficiente. Ci limiteremo comunque a discutere il caso semplice in cui $p \equiv 3 \pmod 4$. Sia dunque p della forma $p = 4k + 3$, ne segue che l'ordine di $\mathbb{QR}_p$ è $(p-1)/2 = 2k+1$. È facile verificare allora che $\sqrt{a} \equiv a^{k+1} \pmod p$; infatti:

$$\left(a^{k+1}\right)^2 \equiv a^{2(k+1)} \equiv a^{2k+1} a \equiv a \pmod{p}.$$

Residui quadratici modulo $N = p \cdot q$. Passiamo ora a discutere il caso di $\mathbb{QR}_N$, quando $N = p \cdot q$ è il prodotto di due primi distinti; in questo caso, i residui quadratici in $\mathbb{Z}_N^*$ sono tutti gli elementi $a \in \mathbb{Z}_N^*$ della forma $a \equiv b^2 \pmod N$. Per caratterizzare la struttura di $\mathbb{QR}_N$ useremo il teorema del resto cinese (cf. Teorema B.13) ovvero il fatto che $\mathbb{Z}_N^* \simeq \mathbb{Z}_p^* \times \mathbb{Z}_q^*$; in virtù di ciò possiamo rappresentare un elemento $a \in \mathbb{Z}_N^*$ come $a = (a_p, a_q)$, dove $a_p = a \bmod p$ ed $a_q = a \bmod q$. L'osservazione chiave è data dal seguente:

Lemma B.18 (Elementi di $\mathbb{QR}_N$). *Sia $N = p \cdot q$ con p e q primi distinti ed $a = (a_p, a_q)$. Allora $a \in \mathbb{QR}_N$ se e solo se $a_p \in \mathbb{QR}_p$ ed $a_q \in \mathbb{QR}_q$.*

Dimostrazione. ($\Rightarrow$) Se a è in $\mathbb{QR}_N$, allora esiste $b \in \mathbb{Z}_N^*$ tale che $a \equiv b^2 \pmod N$. Sia $b = (b_p, b_q)$; segue:

$$(a_p, a_q) = a = b^2 = (b_p^2 \bmod p, b_q^2 \bmod q).$$

Ma allora:

$$a_p \equiv b_p^2 \pmod{p} \qquad a_q \equiv b_q^2 \pmod{q} \tag{B.1}$$

ed a_p, a_q sono residui quadratici in $\mathbb{QR}_p$ e $\mathbb{QR}_q$ (rispettivamente).

($\Leftarrow$) D'altra parte, se $a = (a_p, a_q)$ ed a_p, a_q sono residui quadratici (rispetto agli opportuni moduli), allora esistono $b_p \in \mathbb{Z}_p^*$ e $b_q \in \mathbb{Z}_q^*$ tali che l'Eq. (B.1) è verificata. Sia $b = (b_p, b_q)$. Percorrendo l'altro verso della dimostrazione al contrario, è immediato verificare che b è la radice quadrata di a modulo N. □

Se esaminiamo con attenzione la dimostrazione precedente, possiamo concludere che ogni residuo quadratico $a \in \mathbb{QR}_N$ ha esattamente 4 radici quadrate. Scriviamo $a = (a_p, a_q)$ e siano $\pm b_p$ e $\pm b_q$ le radici quadrate di a_p ed a_q modulo p e modulo q rispettivamente. (Queste sono le uniche radici quadrate di b modulo p e modulo q, cf. Lemma B.14.) Ne segue che le radici di a sono date dagli elementi di $\mathbb{Z}_N^*$ corrispondenti alle coppie

$$(b_p, b_q) \qquad (-b_p, b_q) \qquad (b_p, -b_q) \qquad (-b_p, -b_q).$$

Tali elementi sono distinti, perché b_p e $-b_p$ (risp. b_q e $-b_q$) sono unici modulo p (risp. modulo q). Pertanto:

$$\frac{\#\mathbb{QR}_N}{\mathbb{Z}_N^*} = \frac{\#\mathbb{QR}_p \cdot \#\mathbb{QR}_q}{\#\mathbb{Z}_N^*} = \frac{\frac{p-1}{2}\frac{q-1}{2}}{(p-1)(q-1)} = \frac{1}{4}.$$

Possiamo anche estendere il simbolo di Legendre al caso $N = p \cdot q$; in questo caso si parla di simbolo di Jacobi. Per ogni a coprimo con N, definiamo

$$J_N(a) = J_p(a) \cdot J_q(a).$$

Indicheremo con $\mathbb{J}_N^{+1}$ l'insieme degli elementi in $\mathbb{Z}_N^*$ con simbolo di Jacobi $+1$ (una definizione simile vale per $\mathbb{J}_n^{-1}$). Già sappiamo (dal Lemma B.18) che se $a \in \mathbb{QR}_N$, allora $J_N(a) = +1$; ciò implica immediatamente che $\mathbb{QR}_N \subseteq \mathbb{J}_N^{+1}$. Osserviamo però che $J_N(a) = +1$ anche quando $J_p(a) = J_q(a) = -1$. In altre parole, $J_N(a) = +1$ anche quando *entrambi* $a \bmod p$ ed $a \bmod q$ *non* sono residui quadratici (e quindi anche se a è un non-residuo quadratico modulo N). Per tenere conto di ciò, si introduce l'insieme di elementi di $\mathbb{Z}_N^*$ che hanno simbolo di Jacobi $+1$ pur essendo non-residui quadratici modulo N:

$$\mathbb{QINR}_N^{+1} = \{a \in \mathbb{Z}_N^* : \ a \in \mathbb{QINR}_N, \text{ ma } J_N(a) = +1\}.$$

Lemma B.19 (Struttura di $\mathbb{Z}_N^*$). *Sia $N = p \cdot q$, con p e q primi distinti. Allora: (i) esattamente metà degli elementi di $\mathbb{Z}_N^*$ sono in $\mathbb{J}_N^{+1}$, (ii) esattamente metà degli elementi di $\mathbb{J}_N^{+1}$ sono in $\mathbb{QR}_N$ (e l'altra metà sono in $\mathbb{QINR}_N$).*

Dimostrazione. (i) Sappiamo che $J_N(a) = +1$ se $J_p(a) = J_q(a) = +1$ oppure se $J_p(a) = J_q(a) = -1$. Inoltre metà degli elementi in $\mathbb{Z}_p^*$ (risp. $\mathbb{Z}_q^*$) ha simbolo di Jacobi $+1$ e l'altra metà -1. Allora:

$$\begin{aligned}\#\mathbb{J}_N^{+1} &= \#(\mathbb{J}_p^{+1} \times \mathbb{J}_q^{+1}) + \#(\mathbb{J}_p^{-1} \times \mathbb{J}_q^{-1}) = \#\mathbb{J}_p^{+1} \cdot \#\mathbb{J}_q^{+1} + \#\mathbb{J}_p^{-1} \cdot \#\mathbb{J}_q^{-1} \\ &= 2\frac{(p-1)(q-1)}{4} = \frac{\varphi(N)}{2} = \frac{\#\mathbb{Z}_N^*}{2}.\end{aligned}$$

(ii) Siccome $a \in \mathbb{QR}_N$ se e solo se $J_p(a) = J_q(a) = +1$, abbiamo:

$$\#\mathbb{QR}_N = \#(\mathbb{J}_p^{+1} \times \mathbb{J}_q^{+1}) = \frac{(p-1)}{2}\frac{(q-1)}{2} = \frac{\varphi(N)}{4},$$

da cui segue $\#\mathbb{QR}_N = \#\mathbb{J}_N^{+1}/2$. Siccome $\mathbb{QR}_N \subseteq \mathbb{J}_N^{+1}$, metà degli elementi di $\mathbb{J}_N^{+1}$ sono in $\mathbb{QR}_N$ (e quindi l'altra metà in $\mathbb{QNR}_N$). □

Il simbolo di Jacobi eredita la proprietà moltiplicativa dal simbolo di Legendre:

Lemma B.20 (Proprietà moltiplicativa del simbolo di Jacobi). *Siano $a, b \in \mathbb{Z}_N^*$. Allora $J_N(a \cdot b) = J_N(a) \cdot J_N(b)$.*

Dimostrazione. Possiamo scrivere:

$$\begin{aligned}J_N(a \cdot b) = J_p(a \cdot b) \cdot J_q(a \cdot b) &\overset{\text{Lemma B.17}}{=} J_p(a) \cdot J_p(b) \cdot J_q(a) \cdot J_q(b) \\ &= J_p(a) \cdot J_q(a) \cdot J_p(b) \cdot J_q(b) = J_N(a) \cdot J_N(b).\end{aligned}$$

□

Una conseguenza delle proprietà elencate è il seguente:

Lemma B.21 ($\mathbb{QR}_N$ e $\mathbb{QNR}^{+1}$). *Siano $a, a' \in \mathbb{QR}_N$ e $b, b' \in \mathbb{QNR}_N^{+1}$. Allora: (i) $a \cdot a' \bmod N \in \mathbb{QR}_N$, (ii) $b \cdot b' \bmod N \in \mathbb{QR}_N$ e (iii) $a \cdot b \bmod N \in \mathbb{QNR}_N^{+1}$.*

Dimostrazione. Dimostriamo solo la (iii), la prova delle altre affermazioni è lasciata come esercizio. Siccome $a \in \mathbb{QR}_N$, abbiamo $J_p(a) = J_q(a) = +1$. D'altra parte, siccome $b \in \mathbb{QNR}_N^{+1}$, abbiamo $J_p(b) = J_q(b) = -1$. Usando il Lemma B.17, possiamo scrivere:

$$J_p(a \cdot b) = J_p(a) \cdot J_p(b) = -1 \qquad \text{e} \qquad J_q(a \cdot b) = J_q(a) \cdot J_q(b) = -1,$$

quindi $J_N(a \cdot b) = +1$ per il Lemma B.20. Ora chiaramente $a \cdot b$ non è un residuo quadratico modulo N, perché $J_p(a \cdot b) = -1$ ovvero $a \cdot b \bmod p$ non è un residuo quadratico modulo p. Di conseguenza, $a \cdot b \in \mathbb{QNR}_N^{+1}$. □

Abbiamo visto che, usando il criterio di Eulero (cf. Lemma B.16), è possibile verificare in modo efficiente se un dato elemento di $\mathbb{Z}_p^*$ è un residuo quadratico oppure no. Analogamente, è semplice verificare se $a \in \mathbb{Z}_N^*$ è un elemento di $\mathbb{QR}_N$: si calcolano $J_p(a), J_q(a)$ e quindi si verifica che $J_p(a) = J_q(a) = +1$; se questo è il caso $a \in \mathbb{QR}_N$, altrimenti $a \in \mathbb{QNR}_N$. La situazione è più complicata quando la fattorizzazione di N *non* è nota: in questo caso non si conosce nessun algoritmo efficiente per decidere se a è un residuo quadratico modulo n oppure no. (Sorprendentemente, però, è noto un algoritmo per calcolare $J_N(a)$ senza conoscere la fattorizzazione di N [ES98; BZ10].)

Interi di Blum. Un primo di Blum è un primo p della forma $p \equiv 3 \pmod 4$. Un intero N è detto di Blum se $N = p \cdot q$ e se sia p che q sono primi di Blum. La principale proprietà degli interi di Blum è che ogni residuo quadratico a ha una radice b che è anch'essa un residuo quadratico. Tale radice quadrata, è anche detta radice quadrata *principale* di a.

Lemma B.22 (Radici in $\mathbb{QR}_p$ quando p è un primo di Blum). *Sia p un primo di Blum ed $a \in \mathbb{QR}_p$. Allora $b = a^{(p+1)/4} \bmod p$ è una radice quadrata principale di b modulo p.*

Dimostrazione. Dobbiamo mostrare che la radice quadrata di b (modulo p), è anch'essa un residuo quadratico. Siccome a è un residuo quadratico per ipotesi, il criterio di Eulero (cf. Lemma B.16) implica $a^{(p-1)/2} \equiv 1 \pmod p$. In effetti:

$$b^2 \equiv \left(a^{(p+1)/4}\right)^2 \equiv a \cdot a^{(p-1)/2} \equiv a \pmod p,$$

e quindi b è una radice quadrata di a modulo p. Per verificare che $b \in \mathbb{QR}_p$ basta applicare nuovamente il criterio di Eulero

$$b^{(p-1)/2} \equiv \left(a^{(p+1)/4}\right)^{(p-1)/2} \equiv \left(a^{(p-1)/2}\right)^{(p+1)/4} \equiv 1^{(p+1)/4} \equiv 1 \pmod p.$$

Segue $b \in \mathbb{QR}_p$ e quindi b è principale. □

Il caso di $\mathbb{QR}_N$, con $N = p \cdot q$, è simile:

Lemma B.23 (Radici in $\mathbb{QR}_N$ quando N è un intero di Blum). *Sia $N = p \cdot q$ un intero di Blum e sia $a \in \mathbb{QR}_N$. Allora a ha quattro radici quadrate modulo N, una sola delle quali è principali.*

Dimostrazione. Abbiamo già mostrato (cf. discussione dopo il Lemma B.18) che ogni $a \in \mathbb{QR}_N$ per $N = p \cdot q$ ha quattro radici modulo N. È un semplice esercizio verificare che di queste una sola è a sua volta un residuo quadratico. □

B.4 Curve ellittiche

Le curve ellittiche sono uno strumento potente in teoria dei numeri con svariate applicazioni in diversi campi della scienza. In crittografia hanno dato vita ad una vera e propria branca, detta appunto crittografia sulle curve ellittiche (*Elliptic Curve Cryptography*, ECC). In questo paragrafo introdurremo (per lo più informalmente) i concetti relativi alle curve ellittiche utili ad alcune applicazioni crittografiche. Una trattazione completa e rigorosa esula dagli scopi del testo. Si vedano ad esempio [Sil86; Was08] per approfondire. Una panoramica si trova anche in [Ven09].

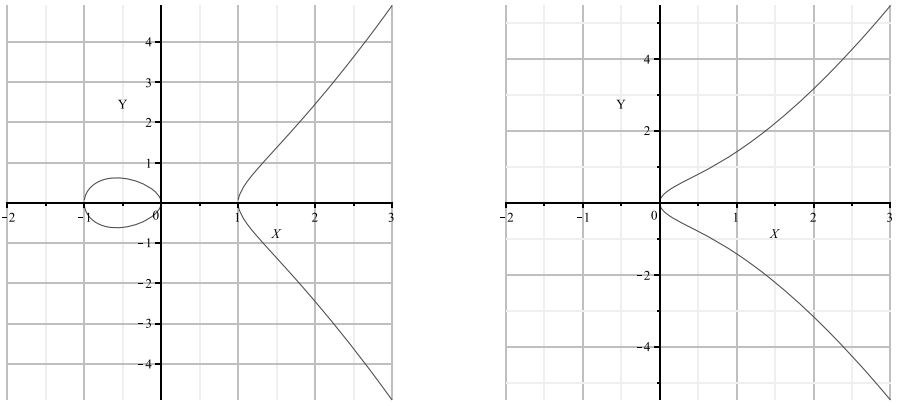
Curve ellittiche su campo $\mathbb{K}$. Sia $(\mathbb{K}, +, \cdot)$ un campo (cf. Definizione B.4) con elemento neutro della somma 0 ed elemento neutro del prodotto 1. Si definisce *caratteristica* di $\mathbb{K}$ (indicata con $\text{char}(\mathbb{K})$), come il minimo numero di volte che bisogna sommare l'elemento 1 per ottenere 0. Se tale valore non esiste, si dice che il campo ha caratteristica nulla, $\text{char}(\mathbb{K}) = 0$. (Ad esempio $\mathbb{Z}_p^*$ ha caratteristica p, mentre l'insieme dei reali $\mathbb{R}$ ha caratteristica 0.) Daremo la definizione di curva ellittica per campi con caratteristica diversa da 2 e 3; anche in questi casi è possibile dare una definizione, ma vanno in un certo senso trattati a parte [Was08, Capitolo 2].

Definizione B.7 (Curva ellittica su un campo $\mathbb{K}$). Sia $\mathbb{K}$ un campo con caratteristica $\text{char}(\mathbb{K}) \neq 2, 3$. Una curva ellittica su $\mathbb{K}$, indicata con $E(\mathbb{K})$, è definita dall'equazione

$$E: \; Y^2 = X^3 + AX + B \quad A, B \in \mathbb{K}. \tag{B.2}$$

■

La curva E è detta non-singolare se non ha zeri doppi, il che si verifica quando il discriminante $\Delta_E = 4A^3 + 27B^2$ è diverso da 0 in $\mathbb{K}$. Se $\mathbb{K} = \mathbb{R}$ possiamo tracciare un grafico della curva, come mostra la Fig. B.3. La curva E è simmetrica (rispetto all'asse delle ascisse) e può avere uno oppure tre zeri reali.



(a) $Y^2 = X^3 - X$ (b) $Y^2 = X^3 + X$

Fig. B.3. Due curve ellittiche su $\mathbb{K} = \mathbb{R}$

Per le applicazioni che ci interessano, è necessario *immergere la curva nel piano proiettivo*. Non presenteremo una definizione formale. Il risultato è che questa operazione consente di aggiungere alla curva un cosiddetto *punto all'infinito* (indicato con ∞) il cui ruolo è quello di dotare l'insieme dei punti di E della struttura di gruppo algebrico. Geometricamente, il punto ∞ può essere pensato come il punto in cui si incontrano due rette parallele.

Più formalmente, sia $Y = \lambda X + \mu$ l'equazione della retta per P e Q. Se calcoliamo l'intersezione con la curva otteniamo:

$$\begin{aligned}
&(\lambda X + \mu)^2 = X^3 + A\ X + B \\
\Rightarrow &X^3 - \lambda^2 X^2 + (A - 2\mu\lambda)X + B - \mu^2 = 0 \\
\Rightarrow &X^3 - \lambda^2 X^2 + (A - 2\mu\lambda)X + B - \mu^2 = (X - x_P)(X - x_Q)(X - x_S) = 0.
\end{aligned}$$

Sviluppando il secondo membro è facile vedere che il coefficiente di X^2 è $-(x_P + x_Q + x_S)$, così che deve risultare $\lambda^2 = x_P + x_Q + x_S$, ovvero

$$x_S = -x_P - x_Q + \lambda^2 = x_R.$$

Inoltre, possiamo sempre scrivere:

$$\begin{aligned}
\lambda = \frac{y_S - y_P}{x_S - x_P} &= \frac{-y_R - y_P}{x_R - x_P} \\
\Rightarrow \quad y_R &= -y_P - (x_R - x_P)\lambda.
\end{aligned}$$

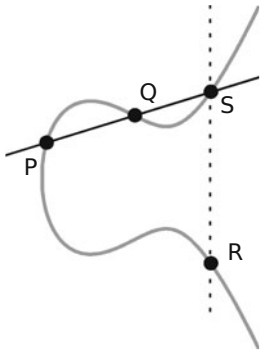


Fig. B.4. Il punto R, somma dei punti P e Q su una curva ellittica

La legge di gruppo. Definiamo ora un'operazione su $E(\mathbb{K})$ tale che dati due punti di $E(\mathbb{K})$, restituisce un altro punto di $E(\mathbb{K})$. Iniziamo con i due punti $P = (x_P, y_P)$ e $Q = (x_Q, y_Q)$, e consideriamo la retta passante attraverso essi; come mostra la Fig. B.4, tale retta interseca la curva in un terzo punto $S = (x_S, y_S)$. Sia $R = (x_R, y_R) = (x_S, -y_S)$ il simmetrico del punto S rispetto all'asse delle ascisse. La somma dei punti P e Q è definita come $P + Q = R$. Restano da distinguere tre casi:

1. $P \neq Q$. Abbiamo semplicemente:
$$\lambda = \frac{y_Q - y_P}{x_Q - x_P}.$$
2. $P = Q$. In questo caso λ è il coefficiente angolare della retta tangente alla curva E in $P = Q = (x_P, y_P)$. Dunque:
$$\begin{aligned} & Y = \sqrt{X^3 + Ax + B} \\ \Rightarrow \quad & \frac{dY}{dX} = \frac{3X^2 + A}{2\sqrt{X^3 + AX + B}} = \frac{3X^2 + A}{2Y} \\ \Rightarrow \quad & \lambda = \frac{3x_P^2 + A}{2y_P}. \end{aligned}$$
3. $P = -Q$. In questo caso poniamo $P + Q = P - P = \infty$.

Abbiamo così trovato la legge di gruppo (*group law*):

Definizione B.8 (Legge di gruppo). Sia $E(\mathbb{K})$ una curva ellittica $E : Y^2 = X^3 + AX + B$ su un campo $\mathbb{K}$, e siano $P = (x_P, y_P)$ e $Q = (x_Q, y_Q)$ due punti di E diversi dal punto all'infinito. Definiamo $P + Q = R = (x_R, y_R)$ come segue:

$$\lambda = \begin{cases} \dfrac{y_Q - y_P}{x_Q - x_P} & \text{se } P \neq Q \\ \dfrac{3x_P^2 + A}{2y_P} & \text{se } P = Q \end{cases}$$
$$x_R = -x_P - x_Q + \lambda^2$$
$$y_R = -y_P - (x_R - x_P)\lambda.$$

Quando $P = -Q$ poniamo $P + Q = \infty$. ■

Consideriamo ad esempio la curva $E : Y^2 = X^3 + X + 1$, con $\mathbb{K} = \mathbb{R}$ e $P = (0, 1)$. Notare che $\Delta_E = 4A^3 + 27B^2 = 31 \neq 0$ e quindi E è non-singolare. In effetti è semplice verificare che P è un punto di E. Calcoliamo $R = P + P$:

$$\lambda = \frac{3x_P^2 + A}{2y_P} = \frac{0+1}{2} = \frac{1}{2}$$
$$x_R = -x_P - x_P + \lambda^2 = \frac{1}{4}$$
$$y_R = -y_P - (x_R - x_P)\lambda = -\frac{9}{8}$$
$$\Rightarrow \quad P + P = R = \left(\frac{1}{4}, -\frac{9}{8}\right).$$

È immediato verificare che R è un punto di $E(\mathbb{R})$.

Il punto chiave, a prima vista sorprendente, è che l'insieme

$$E(\mathbb{K}) = \left\{(x, y) \in \mathbb{K} \times \mathbb{K} : \ y^2 = x^3 + A\, x + B\right\} \cup \{\infty\}$$

dei punti di $E(\mathbb{K})$ più il punto all'infinito, è un gruppo algebrico $(E(\mathbb{K}), +)$ (cf. Definizione B.2) rispetto all'operazione di somma della Definizione B.8.

Curve ellittiche su $\mathbb{Z}_p$. Le principali applicazioni delle curve ellittiche in crittografia hanno a che fare con curve ellittiche definite su un campo finito, ad esempio $\mathbb{K} = \mathbb{Z}_p$. Possiamo adattare facilmente la Definizione B.8 a questo caso particolare.

Definizione B.9 (Curva ellittica su $\mathbb{Z}_p$). Sia $p \neq 2, 3$. Una curva ellittica su $\mathbb{Z}_p$, indicata con $E(\mathbb{Z}_p)$, è definita da un'equazione della forma:

$$E: \; Y^2 \equiv X^3 + AX + B \pmod{p} \qquad A, B \in \mathbb{Z}_p.$$

■

La curva E è detta non-singolare se non ha zeri doppi, ovvero se il discriminante $\Delta_E = 4A^3 + 27B^2 \not\equiv 0 \pmod{p}$. Possiamo anche adattare la legge di gruppo di Definizione B.8:

Definizione B.10 (Legge di gruppo per $E(\mathbb{Z}_p)$). Sia $E(\mathbb{Z}_p)$ una curva ellittica $E : Y^2 \equiv X^3 + AX + B \pmod{p}$ sul campo $\mathbb{Z}_p$, e siano $P = (x_P, y_P)$ e $Q = (x_Q, y_Q)$ due punti di E diversi dal punto all'infinito. Definiamo $P + Q = R = (x_R, y_R)$ come segue:

$$\begin{aligned} \lambda &\equiv \begin{cases} \dfrac{y_Q - y_P}{x_Q - x_P} \pmod{p} & \text{se } P \neq Q \\ \dfrac{3x_P^2 + A}{2y_P} \pmod{p} & \text{se } P = Q \end{cases} \\ x_R &\equiv -x_P - x_Q + \lambda^2 \pmod{p} \\ y_R &\equiv -y_P - (x_R - x_P)\lambda \pmod{p}. \end{aligned} \tag{B.3}$$

Nel caso in cui $P = -Q$ poniamo $P + Q = \infty$. ■

Anche nel caso $\mathbb{K} = \mathbb{Z}_p$, infine, l'insieme

$$E(\mathbb{Z}_p) = \left\{(x, y) \in \mathbb{Z}_p \times \mathbb{Z}_p : y^2 \equiv x^3 + A\ x + B \pmod{p}\right\} \cup \{\infty\},$$

è un gruppo algebrico rispetto all'operazione di somma della Definizione B.10. Inoltre tale gruppo è finito, come mostrato dal seguente:

Teorema B.24 (Teorema di Hasse). *Sia $E : Y^2 = X^3 + AX + B$ una curva ellittica su $\mathbb{Z}_p$, con p un primo. Allora:*

$$\#E(\mathbb{Z}_p) = p + 1 - t,$$

per $|t| < 2\sqrt{p}$.

Il primo algoritmo efficiente per calcolare $\#E(\mathbb{Z}_p)$ è dovuto a Schoof [Sch85].

Concludiamo discutendo brevemente della complessità computazionale associata alla manipolazione e alla rappresentazione di una curva ellittica su un

campo finito. Definire una curva ellittica su $\mathbb{Z}_p$ equivale a scegliere A e B in $\mathbb{Z}_p$, quindi l'occupazione di memoria è $2 \log p = O(\log p)$. Un discorso identico vale per lo spazio necessario a memorizzare un punto P (due coordinate). Il costo dell'addizione è dominato dalla divisione nel calcolo di λ, quindi $O(\log^3 p)$ se ci riferiamo all'algoritmo euclideo (cf. Corollario B.5). Per calcolare un punto P di $E(\mathbb{Z}_p)$ si deve scegliere un valore $x \in \mathbb{Z}_p$, calcolare $\alpha \equiv x^3 + Ax + B \pmod{p}$ e sperare che α sia un residuo quadratico modulo p (il che è facilmente verificabile attraverso il criterio di Eulero, cf. Lemma B.16). Infine, è necessario estrarre la radice, il che può sempre essere fatto efficientemente in $\mathbb{Z}_p$ (cf. Appendice B.3).

Esercizi

Esercizio B.1. Determinare la complessità degli algoritmi elementari di somma, moltiplicazione e divisione in funzione della dimensione degli input.

Esercizio B.2. Usare l'algoritmo di Euclide per calcolare $\gcd(841, 294)$. Quindi, esprimere il risultato come combinazione lineare dei due numeri usando l'algoritmo di Euclide esteso.

Esercizio B.3. Calcolare $\gcd(X^4 - 4X^3 + 6X^2 - 4X + 1, X^3 - X^2 + X - 1)$.

Esercizio B.4. Risolvere la congruenza $3X \equiv 4 \pmod 7$.

Esercizio B.5. Usando il teorema del resto cinese, risolvere il seguente sistema di congruenze:

$$\begin{cases} X \equiv 2 \pmod 3 \\ X \equiv 3 \pmod 5 \\ X \equiv 4 \pmod{11} \\ X \equiv 5 \pmod{16}. \end{cases}$$

Esercizio B.6. Dimostrare che $\mathbb{Z}_{56}$ non è un gruppo rispetto all'operazione di moltiplicazione.

Esercizio B.7. Quanti e quali sono gli elementi di $\mathbb{Z}_{19}^*$? Il gruppo è ciclico? In caso affermativo dire quanti generatori ci sono, altrimenti trovare il sottogruppo di $\mathbb{Z}_{19}^*$ con ordine più grande.

Esercizio B.8. Calcolare 7^{120007} mod 143 a mano.

Esercizio B.9. Calcolare $J_{167}(91)$.

Esercizio B.10. Risolvere $X^2 \equiv 100 \pmod{231}$.

Esercizio B.11. Considerare il gruppo $\mathbb{Z}_{35}^*$.

1. Quanti elementi ci sono? Elencarli.
2. Per ogni elemento del gruppo, decidere se ha simbolo di Jacobi $+1$ oppure -1. Quanti sono gli elementi con simbolo di Jacobi $+1$.
3. Per ogni elemento con simbolo di Jacobi $+1$, determinare se l'elemento è un residuo quadratico oppure no. Quanti elementi con simbolo di Jacobi $+1$ sono residui quadratici?
4. Per ogni residuo quadratico, trovare le corrispondenti radici quadrate.
5. Supponiamo di utilizzare RSA in $\mathbb{Z}_{35}$ con $e = 7$. Qual è il valore di d?

Esercizio B.12. Sia $E : Y^2 = X^3 + 1$ su $\mathbb{Z}_5$. Verificare che si tratta di una curva ellittica. Determinare l'ordine del punto $(2, 2)$ nel gruppo $E(\mathbb{Z}_5)$.

Esercizio B.13. Consideriamo la curva $E : Y^2 \equiv X^3 + X + 2 \pmod{p}$ per $p = 5$. Elencare i punti di $E(\mathbb{Z}_5)$. Sia $P = (1, 2)$; calcoliamo $P + P = 2P$. Infine, mostrare che $E(\mathbb{Z}_5)$ è ciclico e che P è un generatore.

Letture consigliate

[BZ10] Richard P. Brent e Paul Zimmermann. "An $O(M(n)\log n)$ Algorithm for the Jacobi Symbol". In: *ANTS*. 2010, pp. 83–95.

[ES98] Shawna Meyer Eikenberry e Jonathan Sorenson. "Efficient Algorithms for Computing the Jacobi Symbol". In: *J. Symb. Comput.* 26.4 (1998), pp. 509–523.

[JJ05] Gareth A. Jones e Josephine M. Jones. *Elementary Number Theory*. Springer-Verlang, 2005.

[Lan02] Serge Lang. *Algebra*. Springer-Verlang Berlin, 2002.

[Sch85] René Schoof. "Elliptic Curves over Finite Fields and the Computation of Square Roots mod$p$". In: *Math. Comp. 44(170)* (1985), pp. 483–494.

[Sil86] Joseph H. Silverman. *Arithmetic of Elliptic Curves*. Springer-Verlang, 1986.

[Sin97] Simon Singh. *Fermat's Last Theorem: The Story of a Riddle that Confounded the World's Greatest Minds*. Fourth Estate, 1997.

[Ven09] Daniele Venturi. *Introduction to Algorithmic Number Theory*. Rapp. tecn. ECCC TR09-62. SAPIENZA University of Rome, 2009. URL: `http://eccc.hpi-web.de/report/2009/062/`.

[Was08] Lawrence C. Washington. *Elliptic Curves: Number Theory and Cryptography*. Seconda edizione. Chapman & Hall (CRC Press), 2008.

[Wil95] Andrew Wiles. "Modular elliptic curves and Fermat's Last Theorem". In: *Annals of Mathematics* 142 (1995), pp. 443–551.

APPENDICE

C

Problemi computazionali

«Quanto fa uno più uno più uno più uno più uno più uno più uno più uno più uno più uno?». «Non saprei», disse Alice. «Ho perso il conto». «Non è in grado di fare l'addizione», disse la Regina Rossa.

Lewis Carroll, Attraverso lo Specchio e quel che Alice vi trovò [Car71]

Come abbiamo avuto modo di vedere in diverse occasioni, l'ipotesi di base su cui si regge tutta la teoria della crittografia moderna è l'esistenza di alcuni problemi computazionali ritenuti in qualche modo "difficili" da risolvere (in determinate condizioni). Questa appendice offre una panoramica su alcuni di questi problemi, discutendo la loro origine e studiando alcuni degli algoritmi più efficienti conosciuti per attaccarli.

Guida per il lettore. Il primo problema computazionale di cui ci occuperemo (nel Paragrafo C.1) è quello di determinare se un dato numero è primo oppure no. Ciò è molto importante nel contesto di alcuni crittosistemi a chiave pubblica, ad esempio il cifrario RSA (cf. Paragrafo 6.2). Come vedremo, in realtà, questo problema non è un problema difficile, in quanto è sempre risolvibile in tempo polinomiale.

Nel Paragrafo C.2 passeremo a studiare alcuni algoritmi per fattorizzare interi della forma $p \cdot q$, dove p e q sono primi di grande dimensione (diciamo $\approx$ 50 cifre decimali).

Nel Paragrafo C.3, infine, studieremo il problema del logaritmo discreto, che richiede, dato un elemento g^x in un gruppo finito $\mathbb{G}$, di calcolare l'esponente x.

Per una trattazione esaustiva si rimanda a [CP01]. Una panoramica si trova anche in [Sch08; Ste08; Ven09].

Venturi D.: Crittografia nel Paese delle Meraviglie
DOI 10.1007/978-88-470-2481-6_17, © Springer-Verlag Italia 2012

C.1 Test di primalità

Un numero intero $p > 1$ è detto *primo* se p ha come divisori solamente 1 e sé stesso. Lo studio della differenza tra numeri primi e numeri composti (ovvero numeri che sono esprimibili come prodotto di primi) sembra avere origini molto antiche. Euclide è stato il primo a mostrare che ogni numeri intero positivo è esprimibile come prodotto di primi, oppure è primo esso stesso.

Teorema C.1 (Euclide, Libro VII Proposizioni 31 e 32). *Ogni numero intero maggiore di* 1 *è esprimibile come prodotto di primi oppure è esso stesso un primo.*

Dimostrazione. Per induzione. Supponiamo che l'asserto sia vero per tutti gli interi inferiori ad n. Se n è primo non c'è niente da dimostrare. Se n non è primo, esistono due interi n_1, n_2 tali che $n = n_1 \cdot n_2$, con $n_1, n_2 < n$. Ma per l'ipotesi induttiva n_1 e n_2 devono essere esprimibili come prodotto di primi (oppure essere loro stessi primi); sostituendo tali espressioni, otteniamo che anche n è esprimibile come prodotto di primi. □

Quanti sono i numeri primi? La risposta a questa domanda è stata data ancora da Euclide.

Teorema C.2 (Euclide, Libro IX Proposizione 20). *Ci sono infiniti numeri primi.*

Dimostrazione. Per assurdo supponiamo che l'insieme dei primi sia finito e pari a $\{p_1, \ldots, p_k\}$. Consideriamo il numero:

$$n = \prod_{i=1}^{k} p_i + 1.$$

Tale numero non è divisibile per nessuno dei numeri $p_1, \ldots, p_k$. D'altro canto n deve essere divisibile per qualche primo oppure essere primo a sua volta (per il Teorema C.1). Dunque esiste un primo p_{k+1} diverso da $p_1, \ldots, p_k$ e ci sono infiniti primi. □

Gauss è stato il primo a mostrare che la scomposizione in fattori primi di un numero intero è unica.

Teorema C.3 (Teorema fondamentale dell'aritmetica). *Ciascun numero naturale è esprimibile in uno ed un sol modo come prodotto di primi.*

Dimostrazione. Per induzione. Supponiamo che l'asserto sia valido per ogni numero naturale strettamente minore di n. Se n è esso stesso primo non c'è niente da dimostrare. Sia allora n composto e supponiamo che esso ammetta due scomposizioni distinte in fattori primi, vale a dire:

$$n = p \cdot q \cdot r \cdots = p' \cdot q' \cdot r' \cdots,$$

dove $p, q, r, \ldots$ e $p', q', r', \ldots$ sono tutti primi. Osserviamo che nessun primo può comparire in entrambe le scomposizioni in quanto, se così fosse, potrebbe essere cancellato generando due scomposizioni distinte di un numero minore di n, contro l'ipotesi induttiva.

Senza perdita di generalità supponiamo che p e p' siano i primi più piccoli delle due scomposizioni. Siccome n è composto, abbiamo $n \geq p^2$ ed $n \geq (p')^2$; ma p e p' sono distinti e quindi una delle due uguaglianze deve essere vera in senso stretto, ovvero $p \cdot p' < n$. Consideriamo ora il numero $n - p \cdot p'$; siccome tale numero è minore di n, deve ammettere un'unica scomposizione in fattori primi. È evidente d'altra parte che sia p che p' dividono $n - p \cdot p'$, in quanto entrambi dividono n e $p \cdot p'$. Di conseguenza:

$$n - p \cdot p' = p \cdot p' \cdot q'' \cdot r'' \cdots,$$

dove $q'', r'', \ldots$ sono tutti primi. Segue che $p \cdot p'$ deve essere un fattore di n, il che è impossibile, in quanto abbiamo già osservato che i primi $p, q, r, \ldots$ e $p', q', r', \ldots$ devono essere distinti. Concludiamo che la fattorizzazione di n è unica. □

Fissiamo un intero n e chiediamoci quanti sono i primi minori o uguali ad n. Il teorema dei numeri primi (*Prime Number Theorem*, PNT) afferma che per n abbastanza grande ci sono circa $n/\ln n$ primi in $[1, n]$.

Teorema C.4 (Teorema dei numeri primi). *Sia $\pi(n)$ il numero di primi strettamente minori di n. Abbiamo:*

$$\lim_{n \to \infty} \frac{\pi(n) \ln n}{n} = 1.$$

La prova di questo fatto è complessa ed esula dagli scopi del testo. In generale domande di questo tipo vengono studiate in teoria analitica dei numeri. Si veda [Apo76] per un'introduzione.

Il compito di stabilire se un dato intero è un primo oppure no, è molto importante in crittografia. Nel seguito discuteremo due algoritmi: il test di

Miller-Rabin ed il test di Agrawal, Kayal e Saxena. Il primo algoritmo è probabilistico; in particolare non è escluso che un numero composto passi il test venendo identificato come primo (tuttavia ciò avviene solo con bassa probabilità). Il secondo algoritmo è stato il primo test di primalità ad essere contemporaneamente deterministico e polinomiale (mostrando così che il problema di stabilire la primalità di un numero è in **P**). Un altro algoritmo molto importante (basato sulle curve ellittiche) è dovuto a Goldwasser e Killian [GK86] ed è stato poi migliorato da Atkin e Morain [AM93]; seppure il test sia deterministico, l'analisi della complessità è probabilistica (ma si congettura che sia polinomiale). Distingueremo test probabilistici (in cui la risposta è corretta con alta probabilità) e test deterministici (in cui la risposta è sempre corretta).

Il test di Miller e Rabin. L'algoritmo di Miller-Rabin è probabilistico: esso è in grado di stabilire con elevata probabilità (ed in tempo polinomiale) se un dato numero n *non* è primo. D'altro canto, se n passa il test, non possiamo essere certi che esso sia primo. È necessario quindi ripetere l'algoritmo diverse volte, rendendo così trascurabile la probabilità che un numero composto non sia riconosciuto come tale. L'idea originale è dovuta ad Artjuhov[Art67]. Rabin[Rab80] ha proposto la versione probabilistica dell'algoritmo; indipendentemente Miller [Mil76] ha mostrato che, sotto alcune ipotesi, il test può essere trasformato in deterministico.

In un certo senso il test di Miller-Rabin può essere visto come una generalizzazione del seguente *test di Fermat*. Per il piccolo teorema di Fermat (cf. Teorema B.11), se p è un primo, allora per ogni $x \in \mathbb{Z}_p^*$, abbiamo $x^{p-1} \equiv 1 \pmod{p}$. Pertanto, dato un numero n da testare, potremmo pensare di scegliere $x \in \mathbb{Z}_n^*$ e controllare se per caso $x^{n-1} \equiv 1 \pmod{n}$. Il problema di questa strategia è che il piccolo teorema di Fermat fornisce solo una condizione necessaria per la primalità di n; esistono infatti numeri — detti *pseudo-primi di Fermat* in base x — tali che $x^{n-1} \equiv 1 \pmod{n}$, anche quando n è composto (cf. Esercizio C.1). In realtà c'è di peggio: esistono numeri — detti *numeri di Carmichael* — che soddisfano la relazione di cui sopra per *qualsiasi* valore di $x \in \mathbb{Z}_n^*$, indipendentemente dal fatto che n sia primo o composto (cf. Esercizio C.2). L'idea di Miller-Rabin è quella di aumentare la probabilità di successo del test di Fermat.

Lemma C.5 (Radici dell'unità modulo p). *Sia $p > 2$ un primo. Non esistono radici non-banali di 1 modulo p.*

Dimostrazione. Osserviamo che sia 1 che -1 restituiscono sempre 1 quando sono elevati al quadrato modulo p; queste sono le radici banali dell'unità. Supponiamo che $x \in \mathbb{Z}_p$ sia una radice non-banale dell'unità modulo p. Allora

$$x^2 \equiv 1 \pmod{p} \quad \Rightarrow \quad (x+1)(x-1) \equiv 0 \pmod{p}.$$

Siccome x è non-banale, abbiamo $x \not\equiv \pm 1 \pmod{p}$, vale a dire $x+1$ ed $x-1$ sono coprimi con p, ovvero né $x+1$ né $x-1$ sono divisibili per p. Ma allora p non può neanche dividere il prodotto tra $x+1$ ed $x-1$ e possiamo concludere:

$$(x+1)(x-1) \not\equiv 0 \pmod{p},$$

che è una contraddizione. Concludiamo che le uniche radici dell'unità modulo p sono quelle banali. □

Lemma C.6 (Algoritmo di Miller-Rabin). *Sia n un primo dispari. Scriviamo $n-1 = 2^s d$, con d un intero dispari ed $s \geq 1$. Per ogni $x \in \mathbb{Z}_n^*$ si ha che $x^d \equiv 1 \pmod{n}$ oppure $x^{2^r d} \equiv -1 \pmod{n}$ per qualche $0 \leq r < s$.*

Dimostrazione. Siccome n è primo, il piccolo teorema di Fermat ci dice che $x^{n-1} \equiv 1 \pmod{n}$, ovvero n è congruo all'unità in $\mathbb{Z}_n^*$. Se ora calcoliamo ripetutamente la radice quadrata di x^{n-1}, il Lemma C.5 ci dice che otterremo sempre $+1$ oppure -1. Quando il risultato è -1, la seconda uguaglianza è verificata.

Supponiamo d'altra parte di aver continuato ad estrarre le radici quadrate di x^{n-1} senza che la seconda uguaglianza sia mai verificata; ciò equivale a dire che $x^{2^r d} \not\equiv -1 \pmod{p}$ per ogni $r = 0, 1, \cdots, s-1$. Notare che siccome anche x^d è una radice quadrata dell'unità, deve aversi $x^d \equiv \pm 1 \pmod{n}$. Comunque per $r = 0$ abbiamo ottenuto $x^d \not\equiv -1 \pmod{n}$, quindi se la seconda uguaglianza non vale deve valere la prima. □

Analogamente al caso dei numeri pseudo-primi in base x, possiamo definire i numeri *pseudo-primi forti* in base x come gli interi dispari $n > 3$ per cui la condizione del Lemma C.6 è soddisfatta. Se scegliamo a caso $x \in \mathbb{Z}_n^*$ e la condizione del Lemma C.6 non è verificata, possiamo concludere subito che n è composto. D'altro canto, se tale condizione vale per un dato x, non possiamo concludere che n è primo, in quanto n potrebbe essere uno pseudo-primo forte in base x. La speranza è che stavolta la frazione di numeri composti che passino il test sia più bassa. In effetti posto

$$\mathcal{B} = \left\{ x \in \mathbb{Z}_n^* : \; x^d = 1 \text{ oppure } x^{2^r d} = -1 \text{ per qualche } 0 \leq r < s \right\},$$

si può dimostrare che (per ogni intero dispari composto $n > 9$) si ha $(\#\mathcal{B})/\varphi(n) \leq 1/4$.

Questo fatto può essere usato per costruire il seguente test di primalità probabilistico. Si sceglie un valore a caso $x \in \mathbb{Z}_n^*$ e si verifica se $x \in \mathcal{B}$. Quando ciò accade si conclude che n è composto, altrimenti n è probabilmente primo. In una singola istanza la probabilità che $x \in \mathcal{B}$ per n composto è al più $1/4$; ripetendo il test k volte tale valore può essere ridotto a 4^{-k} e quindi reso piccolo a piacere.

Notare che per verificare che x sia un elemento di $\mathcal{B}$, bisogna elevare $x \in \mathbb{Z}_n^*$ ad un esponente non più grande di n. Ciò richiede $O(\log(n)(\log(n))^{\mu})$, con $\mu = 2$ se si fa riferimento agli algoritmi di moltiplicazione ordinaria in $\mathbb{Z}_n$ e $\mu = 1 + \varepsilon$ usando le tecniche di moltiplicazione veloce [Ber08].

Il test di Agrawal, Kayal e Saxena. L'algoritmo AKS [AKS04] (dalle iniziali dei suoi inventori) ha dimostrato per la prima volta che il compito di stabilire la primalità di un dato intero è in **P**. L'ingrediente chiave è contenuto nel seguente teorema, la cui dimostrazione esula dagli scopi del testo.

Teorema C.7 (Algoritmo AKS). *Sia $n > 0$ un intero dispari ed r un primo. Supponiamo che sia verificato quanto segue:*

(i) n non è divisibile per nessuno dei primi $\leq r$;

(ii) l'ordine di n in $\mathbb{Z}_r^*$ è:

$$\operatorname{ord}(n) \geq \left(\frac{\log(n)}{\log(2)}\right)^2;$$

(iii) per ogni $0 \leq a \leq r$, possiamo scrivere:

$$(X + a)^n = X^n + a \qquad \text{in } \mathbb{Z}_n[X]/\left(\frac{X^r - 1}{X - 1}\right).$$

Allora n è potenza di un primo.

Per un primo r, si definisce l'r-simo *polinomio ciclotomico*[93]:

$$\Phi_r(X) = \frac{X^r - 1}{X - 1} = X^{r-1} + \cdots + X^2 + X + 1.$$

[93] Tali polinomi sono connessi al problema di dividere un cerchio unitario in segmenti uguali, ovvero il problema di iscrivere poligoni regolari in un cerchio di raggio 1.

In questo modo, l'anello

$$\mathbb{Z}_n[X]/\left(\Phi_r(X)\right) = \left\{a_{r-2}X^{r-2} + \cdots + a_1 X + a_0 : \ a_i \in \mathbb{Z}_n\right\}$$

contiene tutti i polinomi di grado al più $r-2$ con coefficienti in $\mathbb{Z}_n$. Pertanto rappresentare un singolo elemento di tale anello richiede $O(r \log n)$ bit.

Osserviamo che quando n è primo i coefficienti del polinomio $(X+a)^n$ sono tutti nulli modulo n, quindi, usando il piccolo teorema di Fermat (cf. Teorema B.11), possiamo concludere:

$$(X+a)^n = \sum_{i=0}^{n} \binom{n}{i} X^{n-i} a^i = X^n + a^n = X^n + a \qquad \text{in } \mathbb{Z}_n[X].$$

Purtroppo valutare $(X+a)^n$ per ogni a è troppo costoso quando n è grande: l'idea di Agrawal, Kayal e Saxena è quella di calcolare $(X+a)^n$ modulo l'r-simo polinomio ciclotomico $\Phi_r(X)$.

Si può trasformare il Teorema C.7 in un test di primalità, come segue:

1. Controlla[94] che n non sia potenza di qualche intero.
2. Provando i valori $r = 2, 3, \ldots$ trova il più piccolo primo r che non divide n e che non divide nessuno degli elementi $n^i - 1$ (con $i = 0, 1, \ldots, \left(\frac{\log(n)}{\log(2)}\right)^2$). In simboli:
$$r \nmid n \cdot \prod_{i=1}^{\left(\frac{\log(n)}{\log(2)}\right)^2} (n^i - 1).$$
3. Controlla che la condizione (iii) del Teorema C.7 sia verificata.
4. Se n passa il Test, concludi che n è primo, altrimenti concludi che n è composto.

Dimostriamo che il test funziona. Banalmente se n è primo, passerà il test per il piccolo teorema di Fermat. Supponiamo d'altra parte che n passi il test.

[94] Tale controllo può essere effettuato alla seguente maniera. Sia n della forma $n = m^d$ con $d > 1$ ed $m \geq 2$, ovvero

$$n = m^d \geq 2^d \quad \Rightarrow \quad d \leq \frac{\log(n)}{\log(2)}.$$

Per $d = 2, 3, \ldots, \frac{\log(n)}{\log(2)}$ approssimiamo, con una certa precisione, il valore $n^{1/d}$ in $\mathbb{R}$ e poniamo $m = \text{round}(n^{1/d}) \in \mathbb{Z}$. Verifichiamo quindi che $m^d \neq n$. Il costo è $O(\log^4(n))$.

È sufficiente controllare che tutte le condizioni del Teorema C.7 siano rispettate per poter concludere che n sia primo (perché abbiamo già controllato nel punto (1) dell'algoritmo che esso non è potenza di un primo). La condizione (iii) è sicuramente verificata, in quanto essa coincide con il punto (3) del test; anche la condizione (i) è verificata, per come abbiamo definito r al punto (2). Sia infine o l'ordine di n modulo r, ovvero $n^o \equiv 1 \pmod{r}$ (il che ha senso in quanto $r \nmid n$); allora $r|(n^o - 1)$. Nel punto (2) dell'algoritmo abbiamo verificato che r non divide gli elementi $n^i - 1$ per $i = 0, 1, \ldots, \left(\frac{\log(n)}{\log(2)}\right)^2$, così che possiamo concludere:

$$o > \left(\frac{\log(n)}{\log(2)}\right)^2,$$

che è esattamente la condizione (ii) del Teorema C.7.

Si può dimostrare che $r \leq O(\log^5 n)$. La parte più costosa del punto di vista computazionale è quella in cui dobbiamo calcolare $(X + a)^n$ in $\mathbb{Z}_n[X]/(\Phi_r(X))$ per ogni $a = 0, \ldots, r - 1$. Siccome un elemento nell'anello è rappresentabile con $O(r \log n)$ bit, valutare l'n-sima potenza costa $O(\log n(r^2 \log^2 n))$ se ci si riferisce alle tecniche standard di moltiplicazione. Poiché dobbiamo ripetere per ogni possibile $a = 0, \ldots, r - 1$ la complessità nel caso peggiore sarà $O(r^3 \log^3 n) = O(\log^{18} n)$. Tale valore si abbassa a $O(\log^{12+\epsilon} n)$ se si fa riferimento alle tecniche di moltiplicazione veloce [Ber08].

C.2 Fattorizzazione di interi

Come abbiamo visto, il teorema fondamentale dell'aritmetica assicura che ogni numero intero sia scomponibile in fattori primi in modo univoco. È possibile in qualche modo trovare tale scomposizione? Quando il numero da fattorizzare è molto grande, non esiste (ad oggi) alcun algoritmo efficiente per risolvere il problema. Questo è il motivo per cui alcuni crittosistemi come RSA (cf. Paragrafo 6.2) sono basati sull'ipotesi che fattorizzare alcuni interi sia difficile.

Per testare la robustezza del cifrario RSA, vengono pubblicate periodicamente delle "sfide" in cui si richiede di fattorizzare un intero di certe dimensioni. L'ultimo numero fattorizzato [Kle+10] — della forma $p \cdot q$ con 232 cifre — ha richiesto circa due anni usando alcune tra le reti di super-computer più potenti al mondo (distribuite tra la Svizzera, il Giappone ed i Paesi Bassi).

Sia $p|n$ il più piccolo primo divisore di n (il numero che vogliamo fattorizzare). È ovvio che $p \leq \sqrt{n}$. L'idea più semplice è quella di scegliere un valore casuale $d < n$ e sperare che $\gcd(d, n) > 1$. Siccome la probabilità che il numero scelto

sia divisibile per p è $1/p$, e siccome il costo associato al calcolo del massimo comun divisore è $O(\log^3 p)$ (cf. Corollario B.5), possiamo concludere che la complessità è $O(p \log^3 n)$ che nel caso peggiore (ovvero quando $p \approx \sqrt{n}$) è già esponenziale in n.

Un'altra idea semplice è quella di cercare p semplicemente dividendo n per tutti i primi minori uguali della $\sqrt{n}$. Siccome dobbiamo eseguire tante divisioni quanti sono i primi minori o uguali a p — ovvero $\approx p/\log p$ per il teorema dei numeri primi (cf. Teorema C.4) — abbiamo una complessità $O(\log^2 n \cdot p/\log p)$ che è ancora esponenziale quando $p \approx \sqrt{n}$.

Nel resto di questo paragrafo discutiamo qualche algoritmo più sofisticato: l'algoritmo di Pollard $p-1$ e la sua estensione sulle curve ellittiche (detta ECM) dovuta a Lenstra.

Pollard $p-1$. L'idea alla base dell'algoritmo di Pollard è abbastanza semplice. Purtroppo come vedremo, esso consente di trovare un fattore non banale di n solo in casi molto "fortunati". Nonostante ciò la sua versione sulle curve ellittiche ha dato vita ad uno degli algoritmi più efficienti noti.

Sia n l'intero da fattorizzare. Fissiamo una costante B che rappresenti un limite per il numero di passi che siamo disposti a compiere. Calcoliamo

$$M = \prod_{\substack{p_i \leq B \\ p_i \text{ primo} \\ p_i^{e_i} \leq B}} p_i^{e_i}, \tag{C.1}$$

avendo indicato con e_i l'esponente del fattore primo p_i nella scomposizione di B. Ad esempio se $B = 20$, abbiamo $M = 2^4 \cdot 3^2 \cdot 5 \cdot 7 \cdot 11 \cdot 13 \cdot 17 \cdot 19$.

Una conseguenza del teorema dei numeri primi (cf. Teorema C.4) è che $M \approx e^B$. Fissiamo un valore casuale $x \in \mathbb{Z}_n^*$ e calcoliamo $y = x^M \bmod n$ e $\gcd(y - 1, n) = d$. La speranza è che $1 < d < n$, così che abbiamo trovato un divisore non banale d di n.

Sia ad esempio $n = 10001$ e $B = 10$, così che $M = 2^3 \cdot 3^2 \cdot 5 \cdot 7 = 2520$. Per $x = 2 \in \mathbb{Z}_{10001}^*$, abbiamo:

$$\begin{aligned} y = x^M = 2^{2520} &\equiv 3579 \pmod{10001} \\ \gcd(3579 - 1, 10001) &= 73. \end{aligned}$$

In effetti $10001 = 73 \cdot 137$.

Indichiamo con p il divisore primo minimale di n; l'algoritmo riesce a calcolare p se $p | \gcd(y - 1, n)$, ovvero se $p | (y - 1)$ poiché p è per definizione un

divisore di n. Quando ciò avviene

$$p|(y-1) \quad \Rightarrow \quad y \equiv 1 \pmod{p} \quad \Rightarrow \quad x^M \equiv 1 \pmod{p}.$$

Per il piccolo teorema di Fermat (cf. Teorema B.11), quindi, è sufficiente che M sia un multiplo di $p-1$, ovvero $p-1$ deve dividere M. Ciò è possibile solo se *tutti* i divisori primi di $p-1$ sono più piccoli di B (per costruzione di M), ovvero se $p-1$ è B-liscio (B-*smooth*).

Definizione C.1 (Numeri B-lisci). Sia $B > 0$ un numero reale e $n > 0$ un intero. Diremo che n è B-liscio, se tutti i divisori primi di n sono più piccoli di B. ■

Ad esempio 100 è 10-liscio, poiché $100 = 5^2 \cdot 2^2$ e $2, 5 < 10$.

Possiamo pertanto concludere che l'algoritmo $p-1$ di Pollard ha successo solo se $p-1$ è B-liscio. Il costo computazionale è dato dal calcolo di $y = x^M \bmod n$ e del massimo comun divisore, ovvero:

$$O\left(\log(M)\log^2(n) + \log^3(n)\right) = O\left(B\log^2(n)\right).$$

In pratica, se $B \approx n^{1/10}$, si può vedere che la probabilità che n abbia un divisore primo p tale che $p-1$ è B-liscio, è molto bassa; pertanto l'algoritmo funziona bene solo in casi fortunati. Ovviamente, più i valori di B sono grandi, maggiore è la probabilità che un numero sia B-liscio; d'altro canto la complessità è però esponenziale in $\log(B)$.

Più in generale, l'algoritmo funziona se $x^M \equiv 1 \pmod{p}$, mentre $x^M \not\equiv 1 \pmod{q}$ per ogni altro divisore primo q di n.

Il metodo delle curve ellittiche. Sebbene l'algoritmo Pollard $p-1$ sia poco efficiente, esso costituisce il cuore dell'idea di Lenstra [Len87], che può essere vista come una traduzione nel linguaggio delle curve ellittiche dell'algoritmo di Pollard. L'algoritmo è detto metedo delle curve ellittiche (*Elliptic Curve Method*, ECM).

Come abbiamo visto, l'algoritmo di Pollard funziona quando $p-1$ è B-liscio, dove $p-1 = \#\mathbb{Z}_p^*$. Nell'algoritmo ECM sostituiremo $\mathbb{Z}_p$ con $E(\mathbb{Z}_p)$, essendo E una curva ellittica non-singolare (cf. Definizione B.9). Analogamente a quanto accade nell'algoritmo di Pollard, il valore di p non è noto e tutti i calcoli saranno quindi eseguiti in $E(\mathbb{Z}_n)$ (al posto di $\mathbb{Z}_n^*$).

Supponiamo per semplicità che $n = pq$, dove p e q sono primi distinti maggiori di 3. Il teorema del resto cinese (cf. Teorema B.13) implica che

$$\begin{aligned} \mathbb{Z}_n &\simeq \mathbb{Z}_p \times \mathbb{Z}_q \quad \text{(come anelli)} \\ \mathbb{Z}_n^* &\simeq \mathbb{Z}_p^* \times \mathbb{Z}_q^* \quad \text{(come gruppi)} \\ E(\mathbb{Z}_n) &\simeq E(\mathbb{Z}_p) \times E(\mathbb{Z}_q) \quad \text{(come gruppi)}. \end{aligned}$$

Pertanto l'insieme $E(\mathbb{Z}_n)$ eredita la struttura di gruppo abeliano: la maggior parte dei punti in $E(\mathbb{Z}_n)$ può essere addizionata usando le formule di Definizione B.10. Infatti la formula fallisce solo quando alcune quantità calcolate in $E(\mathbb{Z}_n)$ sono zero modulo p e diverse da zero modulo q (o viceversa). Quando ciò accade n, sarà fattorizzato.

Traduciamo l'algoritmo di Pollard usando il linguaggio delle curve ellittiche. Consideriamo la curva ellittica di equazione

$$E: \; Y^2 = X^3 + AX + B \qquad \gcd(n, 6) = 1 \text{ e } \gcd(\Delta_E, n) = 1,$$

con $\Delta_E = 4A^3 + 27B^2 \neq 0$. Notare che abbiamo richiesto che il discriminante Δ_E della curva non sia divisibile per nessuno dei divisori primi di n, ovvero che la curva E sia non-singolare[95] su $\mathbb{Z}_q$, per ogni divisore primo q di n. (Osserviamo che questo test preliminare è polinomiale, in quanto richiede solamente il calcolo di un massimo comun divisore. Inoltre se per caso $\gcd(\Delta_E, n) \neq 1$ abbiamo già trovato un fattore non banale di n.)

Scegliamo una costante B e calcoliamo il valore M in Eq. (C.1). Fissiamo quindi un punto a caso su $E(\mathbb{Z}_n)$ e proviamo a calcolare $[M]P = P + \cdots + P$ (M volte) in $E(\mathbb{Z}_n)$, il che corrisponde a calcolare $y = x^M$ nell'algoritmo di Pollard. Se ad un certo punto la formula per calcolare il coefficiente angolare λ in Eq. (B.3) fallisce (vale a dire se troviamo un valore non invertibile), allora abbiamo trovato un fattore di n. Ciò accade solo se

$$\begin{aligned} & [M]P = \infty \text{ in } E(\mathbb{Z}_p) \\ & [M]P \neq \infty \text{ in } E(\mathbb{Z}_q) \; \forall \text{ divisore primo } q \neq p \text{ di } n \\ \Leftrightarrow \quad & \#E(\mathbb{Z}_p) | M \quad \text{i. e. } \#E(\mathbb{Z}_p) \text{ è } B\text{-liscio}. \end{aligned}$$

[95] Siccome si può dimostrare che estrarre radici quadrate modulo n (quando n è composto) è equivalente a fattorizzare n (cf. Teorema 6.4), è necessario scegliere il punto P prima della curva E. Quindi, si sceglie prima $P = (x_P, y_P)$ ed un valore casuale per A; poi si calcola:

$$B = y_P^2 - x_P^3 - Ax_P.$$

Infine, si verifica che $\gcd(\Delta_E, n) = 1$.

Un esempio chiarisce immediatamente le idee. Sia $n = 35$ e supponiamo di usare $E : Y^2 = X^3 - X - 2$. È facile verificare che $\gcd(\Delta_E, 35) = 1$. Sia $M = 3$ e $P \in E(\mathbb{Z}_n)$, diciamo $P = (2, 2)$. Dobbiamo calcolare $[M]P$ in $E(\mathbb{Z}_n)$ e sperare che $[M]P = \infty$ in $E(\mathbb{Z}_p)$, ma $[M]P \neq \infty$ in $E(\mathbb{Z}_q)$ per ogni divisore primo $q \neq p$ di n. Ovviamente non conosciamo p, ma, se la condizione di cui sopra si verifica, è sufficiente calcolare $[M]P = [3]P = P + P + P$ in $E(\mathbb{Z}_n)$ per trovare un fattore di n. Cominciamo calcolando $[2]P$, usando le equazioni della Definizione B.10; abbiamo:

$$\lambda = \frac{3x_P^2 + A}{2y_P} = \frac{11}{4} = \frac{11 \cdot 9}{4 \cdot 9} \equiv 99 \equiv -6 \pmod{35}.$$

Quindi,

$$\begin{aligned} x_{2P} &= -x_P - x_P + \lambda^2 \equiv -3 \pmod{35} \\ y_{2P} &= -y_P - \lambda(x_{2P} - x_P) \equiv 3 \pmod{35}. \end{aligned}$$

Pertanto $[2]P = (-3, 3)$. Se ora tentiamo di calcolare $[3]P = 2P + P = (-3, 3) + (2, 2)$, otteniamo:

$$\lambda = \frac{3 - 2}{-3 - 2} = -\frac{1}{5},$$

e 5 non è invertibile in $\mathbb{Z}_{35}$. Abbiamo così trovato un fattore di $n = 5 \cdot 7$. È facile verificare che in effetti

$$\begin{aligned} [3]P &= (2, -2) + (2, 2) = \infty \text{ in } E(\mathbb{Z}_5) \\ [3]P &= (-3, 3) + (2, 2) \neq \infty \text{ in } E(\mathbb{Z}_7), \end{aligned}$$

e che $E(\mathbb{Z}_{35}) \simeq E(\mathbb{Z}_7) \times E(\mathbb{Z}_5)$.

Finora abbiamo solamente tradotto l'algoritmo di Pollard usando le curve ellittiche. Il punto di forza dell'algoritmo ECM sta nel fatto che ora però abbiamo un grado di libertà in più: se l'algoritmo fallisce per una data curva, possiamo semplicemente cambiare curva (senza incrementare B) e riprovare! (Ciò non è possibile in Pollard $p - 1$, in quanto $\mathbb{Z}_p^*$ è unico; se $p - 1$ non è B-liscio non resta altro da fare che aumentare B.)

Per stimare la complessità computazionale dell'algoritmo, bisogna stimare il numero di curve E che è necessario provare prima di avere successo. Ciò è possibile conoscendo: (i) la distribuzione del numero di curve ellittiche su $\mathbb{Z}_p$ in funzione del loro numero di punti e (ii) quanti sono i numeri B-lisci per un dato B. Si può dimostrare che questa analisi, nel caso in cui $p \approx \sqrt{n}$ fornisce una complessità $O(e^{\sqrt{\log(n) \log\log(n)}})$.

Tab. C.1. Algoritmi di fattorizzazione a confronto

Algoritmo	Complessità	Numero massimo di cifre
Divisione	$O\left(e^{\frac{1}{2}\log(n)}\right)$	1-15
ECM	$O\left(e^{\sqrt{\log(n)\log\log(n)}}\right)$	10-70
QS	$O\left(e^{\sqrt{\log(n)\log\log(n)}}\right)$	60-120
NFS	$O\left(e^{\log^{\frac{1}{3}}(n)(\log\log(n))^{\frac{2}{3}}}\right)$	120-200

Metodi di crivello. Gli algoritmi più sofisticati sono basati sui cosiddetti *metodi di crivello* che si basano su una vecchia idea già nota ai tempi di Fermat: si può sperare di calcolare un fattore di n se si riesce a scrivere n come differenza di due quadrati:

$$n = u^2 - v^2 = (u+v)(u-v).$$

Il crivello quadratico (*Quadratic Sieve*, QS) ed il crivello dei campi di numeri (*Number Field Sieve*, NFS) sono basati esattamente su questa osservazione. La Tab. C.1 mette a confronto gli algoritmi di fattorizzazione più importanti.

C.3 Il logaritmo discreto

Un altro problema centrale in crittografia è quello del logaritmo discreto. In questo paragrafo daremo una definizione formale del problema. Quindi studieremo un algoritmo semplice (ma non molto efficiente) per il calcolo di logaritmi discreti in $\mathbb{Z}_p^*$.

Logaritmi in $\mathbb{Z}_p^*$. Sia p un primo. Come abbiamo visto nell'Appendice B.2 il gruppo $\mathbb{Z}_p^*$ è ciclico e si chiama generatore un elemento g che ha ordine $p-1$ in $\mathbb{Z}_p^*$. Dato un tale generatore, ogni elemento $y \in \mathbb{Z}_p^*$ può essere espresso come $y = g^x$ per un qualche intero x.

Definizione C.2 (Logaritmo discreto in $\mathbb{Z}_p^*$). Con la notazione sopra introdotta, diremo che x è il logaritmo discreto di y rispetto alla base g e scriveremo $x = \log_g y$. ■

Osserviamo che per il piccolo teorema di Fermat (cf. Teorema B.11) possiamo sempre aggiungere multipli di $p-1$ all'esponente, ovvero:

$$y = g^x \equiv g^{x+k(p-1)} \pmod{p} \qquad \forall\ k \in \mathbb{Z}.$$

Segue che l'esponente $x = \log_g(y)$ è sempre un elemento di $\mathbb{Z}_{p-1}$.

È facile convincersi che il logaritmo discreto soddisfa le stesse proprietà del logaritmo canonico sui reali. In particolare se $y_1 = g^{x_1}$ ed $y_2 = g^{x_2}$, abbiamo $y_1 \cdot y_2 = g^{x_1+x_2}$ e quindi

$$\log_g(y_1 \cdot y_2) = x_1 + x_2 = \log_g(y_1) + \log_g(y_2).$$

D'altra parte, sia $h \neq g$ un altro generatore di $\mathbb{Z}_p^*$. Possiamo sempre scrivere $g = h^c$ per qualche $c \in \mathbb{Z}_{p-1}$. Dato allora un elemento $y = g^x = h^{cx}$, si ha:

$$\log_h(y) = cx = c \log_g(y) \quad \Rightarrow \quad \frac{\log_h(y)}{\log_g(y)} = c \in \mathbb{Z}_{p-1}^*,$$

ovvero è possibile cambiare la base del logaritmo da g ad h moltiplicando per una costante c che dipende solo da g ed h. Osserviamo infine che se g è un generatore di $\mathbb{Z}_p^*$, allora necessariamente $\log_g(-1) = \frac{p-1}{2}$, poiché g è un non-residuo quadratico modulo p.

Passi-nani e passi-giganti. Sia p un primo e g un generatore di $\mathbb{Z}_p^*$; dato un elemento $y = g^x \in \mathbb{Z}_p^*$ vogliamo calcolare $x \in \mathbb{Z}_{p-1}$. Il modo banale di fare ciò è l'attacco a forza bruta: calcoliamo le potenze successive di g — vale a dire $g^2, g^3, \ldots$ — fino a che non ritroviamo y e quindi x. La complessità è ovviamente $O(e^{\log p})$, esponenziale nella dimensione di p.

Il seguente algoritmo — detto algoritmo passi-nani passi-giganti (*Baby-Step Giant-Step*, BSGS) e dovuto a Shanks — consente di migliorare le prestazioni. (Lo stesso algoritmo può essere utilizzato per calcolare il numero di punti di una curva ellittica su $\mathbb{Z}_p^*$.) Si sceglie una costante $B = \lfloor \sqrt{p} \rfloor + 1$ e si scrive l'espansione in base B di x, ovvero $x = c_0 + c_1 B$. Siccome $x < B^2$ per costruzione, deve aversi $0 \leq c_0, c_1 \leq B$. Si eseguono quindi i passi-nani, calcolando i valori

$$y, yg^{-1}, \ldots, yg^{-B},$$

e memorizzando il risultato in una lista. Si eseguono poi i passi-giganti, calcolando i valori

$$g^B, (g^B)^2, \ldots, (g^B)^B,$$

e memorizzando il risultato in una seconda lista.

Siccome $y = g^x = g^{c_0} \cdot g^{c_1 B}$, basta cercare l'elemento c_0 nella prima lista e l'elemento c_1 nella seconda lista. In altre parole, siccome

$$yg^{-c_0} = (g^B)^{c_1} \qquad 0 \leq c_0, c_1 \leq B,$$

dobbiamo semplicemente cercare un elemento che sia *uguale nelle due liste*.

Il modo migliore per eseguire questa ricerca è quello di organizzare le liste in modo efficiente, ad esempio usando una tabella hash (*hash table* in inglese).[96] In questo modo la ricerca nelle due liste richiede tempo costante $O(1)$ e la complessità è dominata dalla dimensione della costante B:

$$O\left(\sqrt{p}\log^2(p)\right) = O\left(e^{\frac{1}{2}\log(p)}\log^2(p)\right).$$

La complessità è ancora esponenziale (ma comunque più bassa che nell'attacco a forza bruta).

Il metodo del calcolo dell'indice. Il calcolo dell'indice (*Index Calculus*) è il miglior algoritmo conosciuto per risolvere il problema del logaritmo discreto. Non daremo una descrizione di questo algoritmo. La complessità comunque è $O(e^{\sqrt{3\log(p)\log\log(p)}})$. È un problema aperto [SS98] quello di tradurre l'algoritmo usando il linguaggio delle curve ellittiche (il che potenzialmente potrebbe portare un notevole vantaggio in termini di complessità).

[96] Una tabella hash, è una struttura dati che usa una funzione hash (cf. Capitolo 4) per organizzare i dati in modo efficiente. Questa metodologia consente di ottenere tempo di accesso alla lista costante ed indipendente dalla dimensione della lista stessa [Knu98].

Esercizi

Esercizio C.1. Mostrare che 341 è uno pseudo-primo di Fermat in base 2, ma non in base 3. Mostrare che 91 è uno pseudo-primo di Fermat in base 3, ma non in base 2.

Esercizio C.2. Il seguente esercizio spiega il criterio di Korselt, che caratterizza completamente i numeri di Carmichael: un intero n è un numero di Carmichael se e solo se n è positivo, composto, privo di quadrati (i.e., per ogni primo p divisore di n si ha che p^2 non divide n) e per ogni primo p divisore di n si ha che $p-1|n-1$. Rispondere alle seguenti domande:

1. Usare il criterio di Korselt per dimostrare che non esiste un numero di Carmichael con solamente due fattori primi p_1, p_2.
2. Applicando il criterio di Korselt costruire un numero di Carmichael che abbia tre fattori primi p_1, p_2, p_3.
3. Dimostrare il criterio di Korselt.

Esercizio C.3. Verificare se il numero 341 è primo usando il test di Fermat ed il test di Miller-Rabin.

Esercizio C.4. Fattorizzare il numero 1001, usando il metodo $p-1$ ed il metodo delle curve ellittiche.

Esercizio C.5. Calcolare $\log_{11} 5$ in $\mathbb{Z}_{31}$ e $\log_5 27$ in $\mathbb{Z}_{103}$, usando l'algoritmo di Shanks.

Esercizio C.6. Alice e Bob usano un sistema di scambio chiavi di Diffie-Hellman basato sulla curva ellittica $E: Y^2 \equiv X^3 + 3X + 12 \pmod{23}$ che ha 21 punti. Pubblicano anche il punto $P = (2, 7)$. Successivamente Alice invia il messaggio $A = aP = (7, 10)$ e Bob invia il messaggio $B = bP = (9, 3)$.

1. Elencare i punti della curva.
2. Calcolare l'ordine del gruppo generato dal punto P.
3. Calcolare il numero segreto b usando l'algoritmo di Shanks.
4. Calcolare la chiave di sessione.

Letture consigliate

[AKS04] Manindra Agrawal, Neeraj Kayal e Nitin Saxena. "PRIMES is in **P**". In: *Annals of Mathematics* 160 (2004), pp. 781–793.

[AM93] Arthur O. L. Atkin e Francois Morain. "Elliptic Curves and Primality Proving". In: *Math. Comp. 61* (1993), pp. 29–68.

[Apo76] Tom M. Apostol. *Introduction to Analytic Number Theory*. Springer, 1976.

[Art67] M. M. Artjuhov. "Certain Criteria for Primality of Numbers Connected with the Little Fermat Theorem". In: *Acta Arith. 12* (1967), pp. 355–364.

[Ber08] Daniel J. Bernstein. *Fast Multiplication and its Applications, pp. 325–384 in Surveys in Algorithmic Number Theory*. J. P. Buhler and P. Stevenhagen, Math. Sci. Res. Inst. Publ. 44. Cambridge University Press, New York, 2008.

[Car71] Lewis Carroll. *Through the Looking-Glass, and What Alice Found There*. Edizione Italiana Einaudi. Traduzione di Alessandro Ceni. Mac Millan & Co., 1871.

[CP01] Richard Crandall e Carl Pomerance. *Prime Numbers: A Computational Perspective*. Springer, 2001.

[GK86] Shafi Goldwasser e Joe Kilian. "Almost All Primes Can Be Quickly Certified". In: *STOC*. 1986, pp. 316–329.

[Kle+10] Thorsten Kleinjung, Kazumaro Aoki, Jens Franke, Arjen K. Lenstra, Emmanuel Thomé, Joppe W. Bos, Pierrick Gaudry, Alexander Kruppa, Peter L. Montgomery, Dag Arne Osvik, Herman J. J. te Riele, Andrey Timofeev e Paul Zimmermann. "Factorization of a 768-Bit RSA Modulus". In: *CRYPTO*. 2010, pp. 333–350.

[Knu98] Donald Knuth. *The Art of Computer Programming 3: Sorting and Searching*. Addison-Wesley, 1998.

[Len87] Hendrik W. Lenstra. "Factoring Integers with Elliptic Curves". In: *Annals of Mathematics* 126 (1987), pp. 649–673.

[Mil76] Gary L. Miller. "Riemann's Hypothesis and Tests for Primality". In: *JCSS* 13.3 (1976), pp. 300–317.

[Rab80] Michael O. Rabin. "Probabilistic Algorithms for Testing Primality". In: *J. Number Theory* 12 (1980), pp. 128–138.

[Sch08] René Schoof. *Four Primality Testing Algorithms, pp. 101–126 in Surveys in Algorithmic Number Theory*. J. P. Buhler and P. Stevenhagen, Math. Sci. Res. Inst. Publ. 44. Cambridge University Press, New York, 2008.

[SS98] Joseph H. Silverman e Joe Suzuki. "Elliptic Curve Discrete Logarithms and the Index Calculus". In: *ASIACRYPT*. 1998, pp. 110–125.

[Ste08] Peter Stevenhagen. *The Number Field Sieve, pp. 83–100 in Surveys in Algorithmic Number Theory*. J. P. Buhler and P. Stevenhagen, Math. Sci. Res. Inst. Publ. 44. Cambridge University Press, New York, 2008.

[Ven09] Daniele Venturi. *Introduction to Algorithmic Number Theory*. Rapp. tecn. ECCC TR09-62. SAPIENZA University of Rome, 2009. URL: `http://eccc.hpi-web.de/report/2009/062/`.

Glossario

B-smooth: B-liscio. 482

Auxiliary Input Zero Knowledge: conoscenza nulla con input asiliario. 356

Baby-Step Giant-Step: passi-nani passi-giganti. 486

Bilinear Decisional Diffie-Hellman problem: problema decisionale bilineare di Diffie-Hellman. 267

blind signature: firma cieca. 411

Bounded Decoding Distance: decodifica a distanza limitata. 240

Bounded-Retrieval Model: modello della capacità limitata. 35

Bounded-Storage Model: modello dello spazio di memoria limitato. 35

broadcast: radiodiffusione circolare. 35, 408

Certificate Revocation List: lista di certificati revocati. 261

Certification Authority: autorità di certificazione. 258

chain rule: regola della catena. 437

Chinese Remainder Theorem: teorema del resto cinese. 454

Chosen Chipertext Attack: attacco a crittotesto scelto. 106

Chosen Plaintext Attack: attacco a messaggio scelto. 105

client: cliente. 336

Closest Vector Problem: problema del vettore più vicino. 238

commitment: impegno. 398

Common Reference String: stringa comune di riferimento. 370

Computational Diffie-Hellman problem: problema computazionale di Diffie-Hellman. 161

Decisional Diffie-Hellman problem: problema decisionale di Diffie-Hellman. 161

denial of service: interruzione del servizio. 283

e-cash: denaro digitale. 9

eavesdropping: spionaggio. 282

electronic voting: voto elettronico. 410

Elliptic Curve Cryptography: crittografia sulle curve ellittiche. 463

Elliptic Curve Method: metodo delle curve ellittiche. 482

Full Domain Hash: hash a dominio pieno. 211

fuzzy: sfocato. 272

greatest common divisor: massimo comun divisore. 444

group law: legge di gruppo. 466

hard-core: estremo. 57

hash table: tabella hash. 487

hash: polpettina fatta di avanzi di carne e verdure. 68

Hierarchical Identity-Based Encryption: cifrario su base identità di tipo gerarchico. 272

honest but curious: onesti ma curiosi. 403

Honest Verifier Zero Knowledge: conoscenza nulla con verificatore onesto. 355

Index Calculus: calcolo dell'indice. 487

Information Theory: teoria dell'informazione. 425

Initialization Vector: vettore di inizializzazione. 87

interleaving: interallacciamento. 283

Key Distribution Center: centro di distribuzione delle chiavi. 313

Key Encapsulation Mechanism: meccanismo di incapsulamento della chiave. 151

Key Generation Center: centro di generazione delle chiavi. 262

knapsack: sacco. 246

leakage: perdita. 9

Learning Parity with Noise: imparare la parità in presenza di rumore. 243

Learning With Errors: imparare in presenza di errori. 239

Linear Feedback Shift Register: Registro a scorrimento a retroazione lineare. 138

login: accesso. 293

malware: codice maligno. 35

man-in-the-middle: uomo-nel-mezzo. 283

meet-in-the-middle: incontro nel mezzo. 126

Message Authentication Code: codice autenticatore di messaggio. 184

min-entropy: entropia minima. 439

modification: modifica. 283

MultiParty Computation: computazione a parti multiple. 389

negligible: trascurabile. 12

Non-Interactive Zero Knowledge: conoscenza nulla non-interattiva. 371

Number Field Sieve: crivello dei campi di numeri. 485

Oblivious Transfer: trasferimento immemore. 395

One-Time Pad: blocco monouso. 33

One-Way Function: funzione unidirezionale. 54

One-Way Permutation: permutazione unidirezionale. 63

One-Way Trapdoor Function: funzione unidirezionale con botola. 148

Optimal Asymmetric Encryption Padding: cifrario asimmetrico ottimo con riempimento. 158

padding: riempimento. 86, 132

Password-based Chosen-basis CDH: problema CDH su base password a basi scelte. 338

perfect forward secrecy: sicurezza perfetta in avanti. 292

perfectly binding: pefettamente vincolante. 399

perfectly hiding: pefettamente celante. 399

Permutation boxes: scatole di permutazione. 122

Prime Number Theorem: teorema dei numeri primi. 475

Probabilistic Polynomial Time: probabilistico a tempo polinomiale. 12

Proof of Knowledge: prova di conoscenza. 362

prover: dimostratore. 294

PseudoRandom Function: funzione pseudocasuale. 110

Pseudorandom Generator: generatore pseudocasuale. 49

PseudoRandom Permutation: permutazione pseudocasuale. 116

Public Key Infrastructure: infrastruttura a chiave pubblica. 258

quadratic residue: residuo quadratico. 456

Quadratic Sieve: crivello quadratico. 485

Radio-Frequency IDentification: identificazione a radio-frequenza. 304

random variable: variabile aleatoria. 426

re-play: ri-uso. 283

receipt-freeness: assenza di ricevuta. 411

reflection: riflessione. 283

Registration Authority: autorità di registrazione. 264

round: turno. 119

salt and stretching: sale e stiramento. 329

Secure Hash Algorithm: algoritmo hash sicuro. 90

server: servente. 336

session hijacking: scambio della sessione. 282

Shortest Independent Vectors Problem: problema dei vettori più corti indipendenti. 238

Shortest Vector Problem: problema del vettore più corto. 238

Simulation Soundness: Validità di simulazione. 378

Smallest Integer Solution: soluzione agli interi più piccoli. 245

Special Honest Verifier Zero Knowledge: conoscenza nulla speciale con verificatore onesto. 362

Special Soundness: validità speciale. 362

Strong PseudoRandom Permutation: permutazione pseudocasuale forte. 117

Strong Special Honest Verifier Zero Knowledge: conoscenza nulla speciale in senso forte con verificatore onesto. 362

Subset Sum: somma di sottoinsiemi. 246

Substitution boxes: scatole di sostituzione. 122

threshold secret sharing: condivisione di segreti a soglia. 391

timestamp: francobollo temporale. 186

Trusted Third Party: terza parte fidata. 407

uncoercibility: incoercibilità. 411

Universal Composability: componibilità universale. 409

Weak PseudoRandom Function: funzione pseudocasuale debole. 138

Witness Indistinguishability: indistinguibilità di indizi. 356

Zero Knowledge: conoscenza nulla. 349

Indice analitico

UNITEXT – Collana di Informatica

Amministrazione avanzata di server Linux
Massimo Tartamella, Marco Sajeva, Benedetto Vassallo, Lorenzo Puccio
2004, X, 460 pp., ISBN 978-88-470-0234-0

Algoritmi. Lo spirito dell'informatica
David Harel, Yishai Feldman
2008, XXII, 616 pp., ISBN 978-88-470-0579-2

Qualità dei Dati. Concetti, Metodi e Tecniche
Carlo Batini, Monica Scannapieco
2008, XXII, 280 pp., ISBN 978-88-470-0733-8

Robotica mobile. Un'introduzione pratica
Ulrich Nehmzow; edizione italiana a cura di A. Chella, R. Sorbello
2008, XVI, 272 pp., ISBN 978-88-470-0385-9

Ricerca Operativa
Paolo Serafini
2009, XVI, 536 pp., ISBN 978-88-470-0845-8

Crittografia nel Paese delle Meraviglie
Daniele Venturi
2012, XIV, 500 pp., ISBN 978-88-470-2480-9