Appunti per Algebra Computazionale B

Agnese Gini

a.a.2016-17

Indice

Capitolo 1

Accoppiamenti su una curva ellittica

Gli accoppiamenti su una una curva ellittica giocano un ruolo molto importante nello studio delle di questa struttura. In questo capitolo definiremo l'accoppiamento di Weil e quello di Tate Lichtenbaum, ne tratteremo le proprietà principali e forniremo un algoritmo per calcolarli.

1.1 Weil Pairing

L'accoppiamento di Weil è uno dei principali strumenti per lo studio delle curve ellittiche.

Teorema 1.1 (Definizione accoppiamento di Weil)**.** Sia (E, O) una curva ellittica definita su un campo K e m un intero positivo coprimo con la caratteristica del di K. Allora esiste un accoppiamento

$$e_m \colon E[m] \times E[m] \longrightarrow \mu_m$$

detto **accoppiamento di Weil** che soddisfa le seguenti proprietà:

i. e_m è bilineare in ogni variabile;

ii. e_m è alternante e in particolare $e_m(T,T) = 1$ per ogni $T \in E[m]$;

iii. e_m è non degenere, ossia se $e_m(S,T) = 1$ per ogni S allora $T = O$;

iv. e_m è Galois invariante, ossia $e_m(S,T)^\sigma = e_m(S^\sigma, T^\sigma)$.

v. e_m è compatibile, ossia $e_{mm'}(S,T) = e_m(S,T)^{m'}$ per ogni m' intero coprimo con la caratteristica del campo.

Dimostrazione. [Sil08]§III.8. □

Ricordande che nelle ipotesi del teorema il gruppo di m torsione è isomorfo a

$$\mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z} :$$

Corollario 1.2. Sia $\{S,T\}$ una base di $E[m]$ allora $e_m(S,T) = \zeta$ è una radice m-esima primitiva dell'unità in K. In particolare se E è definita su un sottocampo L di K e $E[m] \subset E(L)$ allora $\mu_m \subseteq L$.

Corollario 1.3. Sia E una curva ellittica su $\mathbb{Q}$ allora $E[m] \nsubseteq E(\mathbb{Q})$ per ogni $m \geq 3$.

Ricordiamo, tralasciando i dettagli, che dati due punti $S, T \in E[m]$ classicamente si definisce

$$e_m(S,T) = \frac{g(X+S)}{g(X)}$$

dove $g \in \bar{K}(E)^*$ tale che $\mathrm{div}(g) = m^*(T) - m^*(O)$. Dal punto di vista computazionale questa definizione non è ottimale, perciò si usa una definizione equivalente. Consideriamo due divisori $D_S, D_T \in \mathrm{Div}^0(E)$ tali che $\mathrm{sum}(D_S) = S$ e $\mathrm{sum}(D_T) = T$ e che abbiano supporto disgiunto; visto che S e T sono punti di m torsione, esistono due funzioni $f_T, f_S \in \bar{K}(E)^*$ tali che

$$\mathrm{div}(f_S) = mD_S \text{ e } \mathrm{div}(f_T) = mD_T \tag{1.1}$$

allora si ha[1] che

$$e_m(S,T) = \frac{f_T(D_S)}{f_S(D_T)}.$$

Infatti consideriamo $S', T', R \in E$ tali che

$$mT' = T \qquad mS' = S \qquad S' \neq \pm T \qquad 2R = T' - S'$$

e definiamo

$$D_T = (T) - (O) \qquad D_S = (S + mR) - (mR)$$

Esistono $f_T, f_S \in \bar{K}(E)^*$ che soddisfano 1.1 e come nella costruzione standard $g_T, g_S \in \bar{K}(E)^*$ tali che

$$g_T^m = f_T \circ m \qquad g_S^m = f_S \circ m.$$

Allora

$$\begin{aligned}\frac{f_T(D_S)}{f_S(D_T)} &= \frac{f_T(S+mR)}{f_T(mR)} \cdot \frac{f_S(0)}{f_S(T)} \\ &= \frac{f_T(m(S'+R))}{f_T(mR)} \cdot \frac{f_S(m0)}{f_S(mT')} \\ &= \left(\frac{g_T(S'+R)}{g_T(R)} \cdot \frac{g_S(0)}{g_S(T')}\right)^m\end{aligned}$$

[1] Siano C una curva liscia, $D = \sum_{P \in C} n_P(P)$ un divisore e $f \in \bar{K}(C)^*$ una funzione il cui supporto è disgiunto da D. Allora

$$f(D) := \prod_{P \in C} f(P)^{n_P}.$$

Osserviamo che $m\text{div}(g_T \circ \tau_{iT}) = \text{div}(f_T \circ \tau_{iT'})$, dove con τ_P indichiamo la traslazione per P, e dunque

$$\begin{aligned}\frac{f_T(D_S)}{f_S(D_T)} &= \prod_{i=0}^{m-1} \frac{g_T(R+(i+1)S')}{g_T(R+iS')} \cdot \frac{g_S(iS')}{g_S(T'+iS')} \\ &= \frac{g_T(R+mS')}{g_T(R)} \prod_{i=0}^{m-1} \cdot \frac{g_S(iS')}{g_S(T'+iS')} = \\ &= \frac{g_T(R+S)}{g_T(R)} \\ &= e_m(S,T).\end{aligned}$$

Esempio 1. Una scelta possibile per i divisori è preso $Q \in E$

$$D'_S = (S-Q) - (-Q)$$

$$D'_T = (T+Q) - (Q)$$

in tal modo

$$e_m(S,T) = \frac{f_T(D'_S)}{f_S(D'_T)} = \frac{f_T(S-Q)}{f_T(-Q)} \frac{f_S(Q)}{f_S(Q+T)}. \tag{1.2}$$

Grazie a questa definizione possiamo infatti calcolare l'accoppiamento di Weil usando l'algoritmo di Miller:

Teorema 1.4. Consideriamo una curva ellittica in forma di Weierstrass

$$E\colon\ y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

e siano $P = (x_P, y_P)$ e $Q = (x_Q, y_Q)$ punti di E non zero.

(a) Sia λ è la pendenza della retta per P e Q (o della tangente a E se coincidono) e sia $h_{P,Q} \in K(E)^*$

$$h_{P,Q} := \begin{cases} \frac{y-y_P-\lambda(x-x_P)}{x+x_P+x_Q-\lambda^2-a_1\lambda+a_2} & \lambda \neq \infty \\ x - x_P & \lambda = \infty \end{cases}$$

allora $\text{div}(h_{P,Q}) = (P) + (Q) - (P+Q) - (O)$.

(b) **Algoritmo di Miller**. Consideriamo $N \geq 1$ un intero la cui espansione binaria è

$$N = \epsilon_0 + \epsilon_1 \cdot 2 + \cdots + \epsilon_t \cdot 2^t$$

con $\epsilon_i \in \{0,1\}$ ed $\epsilon_t = 1$. Allora l'algoritmo che segue restituisce f_P tale che

$$\text{div}(f_P) = N(P) - ([N]P) - (N-1)(O).$$

Algorithm 1.1 Algoritmo di Miller

Input: E, N e P come nel Teorema 1.4.
Output: f tale che $\mathrm{div}(f) = N(P) - ([N]P) - (N-1)(O)$.

$T := P$ e $f := 1$
for $i = t-1, \ldots, 0$ **do**
  $f = f^2 \cdot h_{T,T}$
  $T = 2T$
  if $\epsilon_i = 1$ **then**
    $f = f \cdot h_{T,P}$
    $T = T + P$
  end if
end for
return f

Dimostrazione. (a) Supponiamo primi che $\lambda \neq \infty$ e $r\colon\ y = \lambda x + \nu$ la retta per P e Q (o la tangente). $r \cap E = \{P, Q, -P-Q\}$ dunque

$$\mathrm{div}(r(x,y)) = (P) + (Q) + (-P-Q) - 3(O)$$

Inoltre

$$\mathrm{div}(x - x_{P+Q}) = (P+Q) + (-P-Q) - 2(O)$$

allora

$$\mathrm{div}(h_{P,Q}) = \mathrm{div}\left(\frac{y - \lambda x - \nu}{x - x_{P+Q}}\right) = (P) + (Q) - (P+Q) - (O)$$

Nel caso $\lambda = \infty$ basta osservare che $P + Q = O$.

(b) Dal punto (a) abbiamo

$$\mathrm{div}(h_{T,T}) = 2(T) - (2T) - (O)$$

$$\mathrm{div}(h_{T,P}) = (T) + (P) - (T+P) - (O)$$

Consideriamo un'istanza del ciclo for per i fissato. All'inizio del ciclo avremo certi $T = T_i^s$ e $f = f_i^s$ e alla fine T_i^e e f_i^e; il valore di T viene raddoppiato e se $\epsilon_i = 1$ gli viene aggiunto P, ossia

$$T_i^e = 2T_i^s + \epsilon_i P.$$

Analogamente

$$f_i^e = f_i^{s2} \cdot h_{T_i^s, T_i^s} \cdot h_{2T_i^s, P}^{\epsilon_i}$$

Allora

$$\begin{aligned}
\mathrm{div}(f_i^e) &= 2\mathrm{div}(f_i^s) + div(h_{T_i^s, T_i^s}) + \epsilon_i \mathrm{div}(h_{2T_i^s, P}) \\
&= 2\mathrm{div}(f_i^s) + 2(T_i^s) - (2T_i^s) - (O) + \epsilon_i((2T_i^s) + (P) - (2T_i^s + P) - (O)) \\
&= 2\mathrm{div}(f_i^s) + 2(T_i^s) - (2T_i^s + \epsilon_i P) + \epsilon_i (P) - (1 + \epsilon_i)(O).
\end{aligned}$$

Usando che $T_i^e = T_{i-1}^s$ e $f_i^s = f_{i-1}^s$ (i decresce) le relazioni sopra sono

$$T_{i-1}^s - 2T_i^s = \epsilon_i P.$$

$$\mathrm{div}(f_{i-1}^s) - 2\mathrm{div}(f_i^s) = 2(T_i^s) - (2T_i^s + \epsilon_i P) + \epsilon_i(P) - (1+\epsilon_i)(O).$$

e ci permettono di semplificare la scrittura dei valori alla fine dei cicli

$$\begin{aligned} T_0^e &= \epsilon_0 P + 2T_0^s \\ &= \epsilon_0 P + \sum_{i=1}^{t-1} 2^i (T_{i-1}^s - 2T_i^s) + 2^t T_{t-1}^s \\ &= \epsilon_0 P + \sum_{i=1}^{t-1} 2^i \epsilon_i P + 2^t T_{t-1}^s \\ &= \sum_{i=0}^{t} 2^i \epsilon_i P \\ &= NP \end{aligned}$$

dove abbiamo le condizioni iniziali e la scrittura in base due di N.
Infine otteniamo quindi

$$\begin{aligned} \mathrm{div}(f_0^e) &= 2\mathrm{div}(f_0^s) + 2(T_0^s) - (T_0^e) + \epsilon_0(P) - (1+\epsilon_0)(O) \\ &= \sum_{i=1}^{t-1} 2^i (\mathrm{div}(f_{i-1}^s) - 2\mathrm{div}(f_i^s)) + 2(T_0^s) - (NP) + \epsilon_0(P) - (1+\epsilon_0)(O) \\ &= \sum_{i=1}^{t-1} 2^i (2(T_i^s) - (T_{i-1}^s) + \epsilon_i(P) - (1+\epsilon_i)(O)) + \\ &+ 2(T_0^s) - (NP) + \epsilon_0(P) - (1+\epsilon_0)(O) \\ &= 2^t (T_{t-1}^s) + \sum_{i=0}^{t-1} 2^i \epsilon_i(P) - \sum_{i=0}^{t-1} 2^i \epsilon_i(O) - (NP) \\ &= N(P) - (N-1)(O) - (NP). \end{aligned}$$

□

Calcolo del Weil pairing

Siano $S, T \in E[m]$ e $Q \in E$. Per calcolare l'accoppiamento di Weil possiamo seguire i seguenti passi:

1. Con l'algoritmo di Miller calcola f_S e f_T;
2. Sceglie R opportuno
3. Detta $r_{A,B}$ la retta passante per A e B qualsiasi, pone

$$a_S = f_S \frac{r_{S-R,-(S-R)}^m}{r_{S,-R}^m} \quad \text{e} \quad a_T = f_T \frac{r_{T+R,-(T+R)}^m}{r_{T,R}^m}$$

in modo che

$$\mathrm{div}(a_S) = mD_S = m(S-R) - m(-R) \quad \text{e} \quad \mathrm{div}(a_T) = mD_T = m(T+R) - m(R)$$

4. Calcola
$$e_m(S,T) = \frac{a_T(D_S)}{a_S(D_T)} = \frac{f_T(S-R)}{f_T(-R)}\frac{f_S(R)}{f_S(T+R)}.$$

Correttezza. Affinché $f(D)$ si ben definito è richiesta la disgiunzione dei supporti, quini non si può usare direttamente f_S e f_T. Vogliamo una funzione $a_S \in K(E)$ tale che

$$\begin{aligned}
\mathrm{div}(a_S) = mD_S &= m(S-R) - m(-R)\\
&= m[(S-R) + (-S+R) - (-S+R) - (-R) - (S) + 3(O) + (S) - 3(O)]\\
&= m[(S-R) + (-S+R) - 2(O) - (-S+R) - (-R) - (S) + 3(O) + (S) - (O)]\\
&= m[(S-R) + (-S+R) - 2(O)] - m[(-S+R) - (-R) - (S) + 3(O)] + [m(S) - m(O)]\\
&= \mathrm{div}(r^m_{S-R,-(S-R)}) - div(r^m_{S,-R}) + \mathrm{div} f_S\\
&= \mathrm{div}\left(f_S \frac{r^m_{S-R,-(S-R)}}{r^m_{S,-R}}\right)
\end{aligned}$$

Analogamente:

$$\begin{aligned}
\mathrm{div}(a_T) = mD_T &= m((T+R) - m(R)\\
&= m[(T+R) + (-T-R) - (-T-R) - (R) - (T) + 3(O) + (T) - 3(O)]\\
&= m[(T+R) + (-T-R) - 2(O) - (-T-R) - (R) - (T) + 3(O) + (T) - (O)]\\
&= m[(T+R) + (-T-R) - 2(O)] - m[(-T-R) + (R) + (T) - 3(O)] + [m(S) - m(O)]\\
&= \mathrm{div}(r^m_{T+R,-(T+R)}) - div(r^m_{T,R}) + \mathrm{div} f_T\\
&= \mathrm{div}\left(f_T \frac{r^m_{T+R,-(T+R)}}{r^m_{T,R}}\right)
\end{aligned}$$

Osservazione 1.1. L'algoritmo di Miller, leggermente modificato, può essere usato anche per valutare in $R \in E$ $f(P) \in K(E)$, il cui divisore è $N(P) - N(0)$, valutando via via $h_{T,T}(R)$ e $h_{P,T}(R)$. In questo modo i due punti vengono accorpati in un'unica computazione, in tempo lineare.

1.2 Tate-Lichtenbaum Pairing

Un altro accoppiamento che può essere utilizzato sia per la teoria sia nelle applicazioni crittografiche e computato più efficientemente dell'accoppiamento di Weil è quello di Tate-Lichtenbaum. Visto che noi siamo interessati alle curve ellittiche definite sui campi finiti, per non cadere in tecnicismi, lo introdurremo solo per $K = \mathbb{F}_q$ con q potenza di un primo:

Definizione 1.5. Sia $E/\mathbb{F}_q$ una curva ellittica e $n \geq 1$ tale che $n|q-1$. L'**accoppiamento di Tate-Lichtenbaum** è

$$\begin{array}{rccc}
\tau_n\colon & E(\mathbb{F}_q)[n] \times \frac{E(\mathbb{F}_q)}{nE(\mathbb{F}_q)} & \longrightarrow & \mu_n\\
& (P,Q) & \longmapsto & e_n(P, R-\phi(R))
\end{array}$$

Dove ϕ è il $q-$frobenius e $R \in E(\bar{\mathbb{F}}_q)$ tale che $nR = Q$.

Proposizione 1.6. L'accoppiamento di Tate-Lichtenbaum è ben definito, non degenere e bilineare.

Dimostrazione. Prima di tutto osserviamo che $n(R-\phi(R)) = nR-\phi(nR) = Q-Q = 0$ dato che il q frobenius fissa $E(\mathbb{F}_q)$ ed è un omomorfismo. Inoltre non dipende dalla scelta di R supponiamo infatti di avere $R' \in E$ tale che $nR' = Q$. Allora detto $T = R-R'$

$$R' - \phi(R') = R + T - \phi(R) - \phi(T)$$

E quindi per bilinearità del Weil pairing e che $P \in E(\mathbb{F}_q)$

$$\begin{aligned} e_n(P, R'-\phi(R')) =& e_n(P, R-\phi(R))\frac{e_n(P,T)}{e_n(P,\phi(T))} \\ =& e_n(P, R-\phi(R))\frac{e_n(P,T)}{e_n(\phi(P),\phi(T))} \\ =& e_n(P, R-\phi(R))\frac{e_n(P,T)}{e_n(P,T)^{\phi}} \\ =& e_n(P, R-\phi(R)) \end{aligned}$$

Dove nell'ultima uguaglianza abbiamo usato che $n|q-1$ e quindi $\mu_n \in \mathbb{F}_q^*$.
Per la buona definizione rimane da far vedere che τ_n non dipende dalla classe modulo $nE(\mathbb{F}_q)$ nella seconda entrata. Visto che n è un omomorfismo allora preso $Q' \in Q + nE(\mathbb{F}_q)$, si ha che $S = Q-Q' \in E(\mathbb{F}_q)$ e $nS = O$, dunque ci basta far vedere che

$$\tau_n(P,S) = 1$$

Ma preso $S' \in\in E(\mathbb{F}_q)$ tale che $nS' = S$

$$\tau_n(P,S) = e_n(P, S'-\phi(S')) = e_n(P, S'-S') = 1.$$

La bilinearità nella prima entrata discende direttamente dal Teorema 1.1; prendiamo $Q_1, Q_2 \in E(\mathbb{F}_q)/nE(\mathbb{F}_q)$

$$\begin{aligned} \tau_n(P, Q_1+Q_2) =& e_n(P, R_1+R_2-\phi(R_1)-\phi(R_2)) \\ =& e_n(P, R_1-\phi(R_1))e_n(P, R_2-\phi(R_2)) \\ =& \tau_n(P,Q_1)\tau_n(P,Q_2). \end{aligned}$$

La non degenericità è più laboriosa e quindi daremo solo l'idea (per i dettagli vedere [Was08]§11.7). Chiaramente il Tate-Lichtenbaum pairing è non degenere nella prima entrata, vorremmo dire quindi che se

$$\tau_n(P,Q) = e_n(P, R-\phi(R)) = 1 \ \forall P \in E(\mathbb{F}_q) \Rightarrow Q \in nE(\mathbb{F}_q).$$

L'ipotesi ci dice che $R-\phi(R) \in (\phi-1)E[n]$ dunque esiste $T \in E[n]$ tale che $R-\phi(R) = \phi(T)-T$ e dunque $\phi(R+T) = R+T \in (\mathbb{F}_q)$. Ma $Q = nR = nR+O = n(R+T)$ e quindi $Q \in nE(\mathbb{F}_q)$.

□

Per il calcolo effettivo dell'accoppiamento di Tate ci si può ricondurre alla strategia di Miller, tuttavia questo risulta meno oneroso perché richiede metà delle valutazioni (quantità che su larga scala è incidente).
Per semplificare quanto segue non calcoliamo direttamente il valore $\tau_n(P,Q)$ come nella definizione ma utilizzeremo una scrittura equivalente:

$$\begin{aligned}\tau_n(P,Q) &= e_n(P, R-\phi(R)) \\ &= \frac{1}{e_n(P,R-\phi(R))^{-1}} \\ &= \left(\frac{1}{e_n(P,R-\phi(R))}\right)^{-1} \\ &= \big(e_n(R-\phi(R),P)\big)^{-1} = e_n(\phi(R)-R,P)\end{aligned}$$

Ricordando la prima definizione dell'accoppiamento di Weil prendiamo f_P tale che $\mathrm{div}(f_P) = n(P) - n(O)$ e $g \in \mathbb{F}_q(E)$ tale che $f_P \circ [n] = g^n$, allora

$$\tau_n(P,Q) = e_n(\phi(R)-R,P) = \frac{g(X+\phi(R)-R)}{g(X)} \in \mu_n$$

per ogni $X \in E$. Scegliendo $X = R$ e usando che Weil è Galois invariante

$$\tau_n(P,Q) = \frac{g(\phi(R))}{g(R)} = \frac{g(R)^{\phi}}{g(R)} = \frac{g(R)^q}{g(R)} = g(R)^{q-1} = (f\circ[n])(R)^{\frac{q-1}{n}} = f(Q)^{\frac{q-1}{n}}.$$

Facciamo adesso vedere che, sotto certe ipotesi, utilizzare è possibile utilizzare sia l'uno che l'altro accoppiamento.

Lemma 1.7. Sia ℓ un primo tale che

- $\ell \mid q-1$ $\quad\triangleright \mu_\ell \subseteq \mathbb{F}_q$
- $\ell \mid \#E(\mathbb{F}_q)$ $\quad\triangleright E(\mathbb{F}_q)[\ell] \neq \varnothing$
- $\ell^2 \nmid \#E(\mathbb{F}_q)$ $\quad\triangleright E(\mathbb{F}_q)[\ell]$ ciclico?

Sia P un generatore di $E(\mathbb{F}_q)[\ell]$, allora $\tau_\ell(P,P) = \zeta$ una radice primitiva $\ell-$esima dell'unità.

Dimostrazione. Supponiamo che $\tau_\ell(P,P) = 1$, allora per ogni u intero $\tau_\ell(uP,P) = 1^u = 1$ dunque visto che τ_ℓ è non degenere $P \in \ell E(F_q)$. Allora esiste $Q \in E(F_q)$ tale che $\ell Q = P$ e visto che $P \in E(\mathbb{F}_q)[\ell]$ si ha che $\ell^2 Q = \ell P = O$. Per la terza ipotesi su ℓ si deve aver però che $Q = O$ e $P = O$, assurdo perché P è un generatore. Usando che ℓ è primo si ha la tesi. □

Proposizione 1.8. Sia E una curva ellittica su un campo finito $\mathbb{F}_q$ e ℓ un primo tale che

- $\ell \mid \#E(\mathbb{F}_q)$
- $E[\ell] \nsubseteq E(\mathbb{F}_q)$
- $\ell \nmid q(q-1)$

Allora $E[\ell] \subseteq E(\mathbb{F}_{q^m})$ se e solo $q^m \equiv 1 \mod \ell$.

Dimostrazione. Se $E[\ell] \subseteq E(\mathbb{F}_{q^m})$ allora $\mu_m \subseteq \mathbb{F}_{q^m}$ dunque $\ell \mid q^m - 1$.
Viceversa supponiamo che $q^m \equiv 1 \mod \ell$ e siano $Q \in E[\ell] \setminus E(\mathbb{F}_q)$ e $P \in E(\mathbb{F}_q)$, un punto di ordine ℓ. Allora $\{P, Q\}$ è una base di $E[\ell]$ e possiamo descrivere l'azione del $q-$ frobenius ϕ sui punti di ℓ torsione in tali coordinate con

$$M = \begin{pmatrix} 1 & b \\ 0 & d \end{pmatrix}$$

in particolare deve valere che

$$d + 1 \equiv tr(\phi) = a = q + 1 - \#E(\mathbb{F}_q) \equiv q + 1 \mod \ell$$

e quindi $d \equiv q \mod \ell$.
Dato che il q^m-frobenius è ϕ^m si ha che la sua azione è rappresentata da

$$M^m = \begin{pmatrix} 1 & b \\ 0 & q \end{pmatrix} = \begin{pmatrix} 1 & b\frac{q^m-1}{q-1} \\ 0 & q^m \end{pmatrix}$$

$E[\ell] \subseteq E(\mathbb{F}_{q^m})$ se e solo se l'azione del q^m-frobenius è banale, perciò, se e solo se $M = I \mod \ell$ se e solo se $q^m \equiv 1 \mod \ell$. □

1.3 Pairing-Friendly Elliptic Curves

In generale quando utilizziamo gli accoppiamenti in ambito crittografico, specialmente quando è sfruttato il problema del logaritmo discreto (ECDLP) [2], gli attacchi sono più efficienti quando si prendono curve ellittiche E definite sui campi finiti tali che $E(\mathbb{F}_q)$ contenga un punto il cui ordine sia un primo N abbastanza grande e tale che il grado di immersione[3] d di N in F_q non sia troppo grande. Curve di questo tipo sono dette **curve ellittiche Pairing-Friendly**.
Il tipo di interazione tra queste grandezze dipende dalle istanze del problema, ma è sempre opportuno bilanciare la difficoltà del ECDLP con la difficoltà di ECL in $\mathbb{F}^*_{q^d}$.

[2] Vedi Capitolo 4.
[3] Ossia il minimo d tale che $E[N] \subseteq E(\mathbb{F}_{q^d})$.

Capitolo 2

Calcolo dell'ordine di curve ellittiche su campi finiti

Sia (E, O) una curva ellittica su un campo finito $\mathbb{F}_q$. Fissato delle coordinate di Weierstrass, si ha che i punti razionalo sono del tipo (x, y) con $x, y \in \mathbb{F}_q$ oppure il punto all'infinito O e visto che le possibili scelte di x e y sono finite si ha che la cardinalità $\#E(F_q) < \infty$.
Più avanti vedremo come conoscere l'ordine della curva è utile, intanto visto che sappiamo che è un gruppo finito e conosciamo la struttura dei gruppi di torsione possiamo dire qualche cosa:

Teorema 2.1. Sia E una curva ellittica su un campo finito $\mathbb{F}_q$, dove $q = p^r$ potenza di un primo p. Allora

$$E(\mathbb{F}_q) \simeq \mathbb{Z}/n\mathbb{Z} \quad \text{o} \quad E(\mathbb{F}_q) \simeq \mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/n_2\mathbb{Z}$$

con $n, n_1, n_2 \in \mathbb{Z}^+$ e $n_1 \mid n_2$.

Dimostrazione. Il teorema di struttura ci dice che un gruppo abeliano finito isomorfo a

$$\mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_s\mathbb{Z}$$

con $n_1, \ldots, n_s \in \mathbb{Z}^+$ e $n_1 \mid \cdots \mid n_s$. Fissato un indice i in $\{1, \ldots, s\}$ la componente n_i-esima deve avere n_1 elementi il cui ordine divide n_1. Allora $(\mathbb{F}_q)$ deve avere n_1^i elementi il cui ordine è n_1. Per la natura dei gruppi di $m-$torsione allora $r \leq 2$. □

Il più importante teorema per quanto riguarda la cardinalità di una curva ellittica su un campo finito è il Teorema di Hasse, il quale da un stima che in alcuni casi è sufficiente per determinare l'ordine del gruppo.

Teorema 2.2 (Hasse)**.** Sia E una curva ellittica su un campo finito $\mathbb{F}_q$, dove $q = p^r$ potenza di un primo p.
Allora

$$|\#E(\mathbb{F}_q) - q - 1| \leq 2\sqrt{q}.$$

Una volta dimostrati un po' di fatti la prova del teorema è una semplice osservazione. Richiamo i risultati necessari per la prova del teorema di Hasse la cui trattazione si può trovare su [Sil08] o [Was08] e alcuni fatti sulle curve ellittiche su campi finiti ci serviranno più in là:

Teorema 2.3. Sia E una curva ellittica su un campo finito $\mathbb{F}_q$ e $q = p^r$ potenza di un primo p.

1. L'endomorfismo di Frobenius $\phi_q \in \mathrm{End}(E)$ è puramente inseparabile e di grado q.
2. Se $n \geq 1$ allora
   (a) $\ker(\phi_q^n - 1) = E(\mathbb{F}_{q^n})$,
   (b) $\phi_q^n - 1$ è separabile e $\deg(\phi_q^n - 1) = \#E(\mathbb{F}_{q^n})$.

Dimostrazione. (*Teorema 2.2*)
Il morfismo di frobenius è puramente inseparabile perciò l'isogenia $\phi - 1$ è separabile. Allora

$$\#E(\mathbb{F}_q) = \ker(\phi - 1) = \deg_s(\phi - 1) = \deg(\phi - 1)$$

Infine, visto che il grado è una forma quadratica definita positiva si ha che

$$|\#E(\mathbb{F}_q) - q - 1| = |\deg(\phi - 1) - \deg\phi - \deg 1| \leq 2\sqrt{\deg\phi \cdot \deg 1} = 2\sqrt{q}.$$

□

Teorema 2.4. Sia E una curva ellittica su un campo finito $\mathbb{F}_q$.

1. Sia $a \in \mathbb{Z}$ tale che
   $$a = q + 1 - \#E(\mathbb{F}_{q^n})$$
   la **traccia del Frobenius**, allora
   $$\phi_q^2 - a\phi_q + q = 0 \in \mathrm{End}(E)$$
   ed a è l'unico intero tale che $a \equiv tr((\phi_q)_m) \mod m$ per ogni $m \in \mathbb{N}$ tale che $\gcd(m, p) = 1$.
2. Date $\alpha, \beta \in \mathbb{C}$ radici di $p(T) = T^2 + aT + q$, allora
   $$\#E(\mathbb{F}_{q^n}) = q^n + 1 - (\alpha^n + \beta^n).$$

Ricordiamo

Definizione 2.5. Il **simbolo di Legendre** associato al primo $p \geq 2$

$$\left(\frac{x}{p}\right) := \begin{cases} 1 & \exists t \in \mathbb{F}_p^* \colon t^2 \equiv x \mod p \\ -1 & \nexists t \in \mathbb{F}_p^* \colon t^2 \equiv x \mod p \\ 0 & x = 0 \end{cases}$$

e per $q = p^r$ il **simbolo di Legendre generalizzato** è dato $x \in \mathbb{F}_q$

$$\left(\frac{x}{\mathbb{F}_q}\right) := \begin{cases} 1 & \exists t \in \mathbb{F}_q^* \colon t^2 \equiv x \mod p \\ -1 & \nexists t \in \mathbb{F}_q^* \colon t^2 \equiv x \mod p \\ 0 & x = 0 \end{cases}$$

Teorema 2.6. Sia E una curva ellittica su un campo finito $\mathbb{F}_q$ in forma di Weierstrass $y^2 = f(x)$ con $f(x) = x^3 + Ax + B$, allora

$$\#E(\mathbb{F}_q) = q + 1 + \sum_{x \in \mathbb{F}_q} \left(\frac{f(x)}{\mathbb{F}_q}\right)$$

2.1 Ordine di un punto

Un'idea per calcolare l'ordine del gruppo, dato che conosciamo a priori informazioni sulla struttura, è quella di sfruttare il teorema di Lagrange e studiare l'ordine dei suoi elementi. Cercare ordine di un punto (ma anche trovare un punto) di una curva ellittica è processo dispendioso. Prima di dare un metodo per calcolare effettivamente la cardinalità di $E(\mathbb{F}_q)$ occupiamoci di questo problema, così da capire il motivo per l'algoritmo di Schoof evita di manipolare esplicitamente dei punti.
Sia E una curva ellittica su un campo finito $\mathbb{F}_q$ e $P \in E(\mathbb{F}_q) \setminus \{O\}$. L'**ordine** di P è il minimo intero positivo k tale che $kP = O$. Il teorema di Hasse ci dice che l'ordine N di $E(\mathbb{F}_q)$ giace in un intervallo di ampiezza $4\sqrt{q}$, dunque se troviamo un punto P_1 il cui ordine ha un solo multiplo in tale intervallo tale multiplo sarà proprio N. In realtà può aver senso listare un certo numero di punti i cui ordini hanno un numero finito di multipli in tale intervallo.

In primo luogo è lecito chiedersi se un punto come P_1 esiste. Un risultato di Mestre ci dice che per primi sufficientemente grandi la risposta è positiva. Per poter enunciare questo risultato ci serve una definizione:

Definizione 2.7. Sia E una curva ellittica su un campo finito $\mathbb{F}_p$ e $d \in \mathbb{F}_p^*$ che non sia un quadrato. Supponiamo che $E\colon\ y^2 = x^3 + Ax + B$. Chiameremo **twist quadratico** la curva

$$E'\colon\ y^2 = x^3 + Ad^2x + Bd^3$$

Osservazione 2.1. Vale che:

- $j(E) = j(E')$
- Esiste una trasformazione di E' in E su $\mathbb{F}_p(\sqrt{d})$
- $E'(\mathbb{F}_p) = p + 1 + a$.

Proposizione 2.8. Sia $p > 229$ un primo e E una curva ellittica sul campo finito $\mathbb{F}_p$. Allora o E o E', il twist quadratico, possiede un punto il cui ordine ha un unico multiplo in $(p+1-2\sqrt{p}, p+1+2\sqrt{p})$ $(p+1-2\sqrt{p}, p+1+2\sqrt{p})$.

Prima di procedere notiamo che questa strategia può avere qualche problema quando $E(\mathbb{F}_q)$ è della forma $\left(\mathbb{Z}/\ell\mathbb{Z}\right)^2$ (con ℓ primo). In tal caso infatti entrambi i generati del gruppo hanno ordine ℓ e quindi l'unica cosa che si può dedurre è che $\ell \mid N$. Fortunatamente questo caso è raro:

Proposizione 2.9. Sia E una curva ellittica su un campo finito $\mathbb{F}_q$ tale che

$$E(\mathbb{F}_q) \simeq \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}$$

per un certo intero n. Allora vale una delle seguenti opzioni:

- $q = n^2 + 1$
- $q = n^2 \pm n + 1$

- $q = (n \pm 1)^2$

Dimostrazione. Il teorema di Hasse ci dice che $n^2 = 1 + q + a$ con $|a| \leq 2\sqrt{q}$. Dico che $a \equiv 2 \mod n$: consideriamo la caratteristica p del campo, si ha che $p \nmid n$ altrimenti dovrebbero esserci p^2 punti in $E[p]$ che però è isomorfo $\mathbb{F}_p$ o un punto; dunque dato che $\mu_n \subseteq \mathbb{F}_q^*$ ($E[n] \in E(\mathbb{F}_q)$) esiste $k \in \mathbb{N}^+$ tale che $kn = q - 1$ e quindi

$$a \equiv q + 1 - n^2 \equiv kn + 2 - n^2 \equiv 2 \mod n.$$

Allora $a = nh + 2$ con $h \in \mathbb{Z}$ e

$$n^2 = q + 1 - nh - 2 = q - 1 - nh$$

$$q = n^2 + nh + 1$$

e ricordando che

$$|q + 1 - n^2| = |2 + hn| \leq 2\sqrt{q}$$

si ha $|h| \leq 2$ da cui la tesi. □

Osservazione 2.2. Supponiamo che $E(F_q)$ si isomorfa a $\mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/n_2\mathbb{Z}$ con $n_1 \mid n_2$. Allora l'ordine di ogni elemento divide n_2. *Quale è la probabilità scegliendo un po' di punti random che il minimo comune multiplo degli ordini sia n_2?*
Siano P_1, P_2 punti di ordine rispettivamente n_1, n_2 e tali che ogni $P \in E(\mathbb{F}_q)$ si scriva unicamente come $P = a_1P_1 + a_2P_2$ con $0 \leq a_i < n_i$. Consideriamo ℓ un primo tale che $\ell \mid n_2$. Scegliamo random un punto P, la probabilità che $\ell \nmid a_2$ è $1 - 1/\ell$, in particolare se questo succede l'ordine del punto è divisibile per il massimo e tale che $\ell^e \mid \#E(\mathbb{F}_q)$. Se $\ell >> 0$ la probabilità che questo avvenga è alta e quindi quasi tutti i primi grandi che dividono n_2 dividono l'ordine di un punto random. Questo vuol dire che facendo un po' di queste scelte dovremmo avere la giusta potenza di questi primi.

Osservazione 2.3. Fissato un punto, se cerchiamo k tale che $kP = O$ usando il teorema di Hasse una strategia rudimentale è provare tutti gli interi compresi in $(q + 1 - 2\sqrt{q}, q + 1 + 2\sqrt{q})$. Questo procedimento costa intorno ai $4\sqrt{q}$ passi.

2.1.1 Baby Step-Giant Step

L'osservazione 2.3 ci da un bound da migliorare, esibiamo adesso una procedura che fa proprio questo.

Algorithm 2.1 Baby Step-Giant Step,ordine di un punto

Input: E curva ellittica su $\mathbb{F}_q$ e $P \in E(\mathbb{F}_q)$.
Output: $M \in \mathbb{Z}^+$ tale che $MP = O$.

1: $Q := (q+1)P$
2: sceglie $m > \sqrt[4]{q}$ (buon approssimante)
3: calcola l'insieme $J = \{jP\}_{j=0,\dots,m-1}$
4: trovato=FALSE e $h = -m$
5: **while** trovato=TRUE & $h \leq m$ **do**
6: $R = Q + 2mhP$
7: **if** $\exists j : \pm R = jP \in J$ **then**
8: trovato=TRUE
9: **else** h++
10: **end if**
11: **end while**
12: $M = q + 1 + 2mh \mp j$
13: fattorizza $M = \prod p_i^{e_i}$
14: **for** $i = 1, \dots, r$ **do**
15: **if** $(M/p_i)P = O$ **then**
16: $M = M/p_i$ e ricomincia da (13)
17: **end if**
18: **end for**
19: **return** M

Correttezza. 2.1.1

(a) *Perché c'è un match e il ciclo while termina?*

Lemma 2.10. Sia $a \in \mathbb{Z}$ tale che $|a| \leq 2m^2$ allora esistono $c, d : -m \leq c, d \leq m$ tali che $a = c + 2md$.

Dimostrazione. Sia $c \equiv a \mod m$ e $d = (a-c)/2m$ allora

$$|d| \leq \frac{2m^2 + m}{2m} < m + 1.$$

□

Usando la notazione del lemma con a la traccia del frobenius quindi abbiamo

$$Q + 2hmP = (q + 1 + 2hm)P \overset{h=-d}{=} (q + 1 - a + c)P = (M + c)P = cP.$$

(b) *Perché 12-18 restituisce l'ordine di P?*
Supponiamo $t \neq M$ sia l'ordine di P allora $k \mid M$ strettamente e quindi esiste un fattore primo p di M che divide M/t allora $t \mid M/p$ e quindi $(M/p)P = 0$.

□

Possiamo fare alcune osservazioni riguardo l'efficienza:

- Per migliorare l'algoritmo in termini di spazio si può conservare invece che i jP solamente le loro coordinate x e calcolare y solo in caso di coincidenza.
- I punti si possono calcolare per somme successive usando l'associatività

$$\begin{cases} (j+1)P = jP + P \\ Q + (h+1)2mP = (Q + h2mP) + 2mP \end{cases}$$

- Fattorizzare un intero è un problema non banale, il passo 13 può essere sostituito con "trova i fattori primi di M minori di un certo B''.

Complessità. 2.1.1 L'algoritmo impiega circa $2m + 2m = 4\sqrt[4]{q}$ passi. □

Se vogliamo stimare $\#E(\mathbb{F}_q)$:

Algorithm 2.2 Ordine della curva BG

Input: E curva ellittica su $\mathbb{F}_q$.
Output: $\#E(\mathbb{F}_q)$.

1: calcola random l'insieme di punti $I = \{P_j\}_{j=1,\dots,t}$
2: **for** $i = 1, \dots, t$ **do**
3: Usando Baby Step- Giant Step calcolare M_i
4: **end for**
5: Calcola $L = \text{lcm}\{M_i : i = 1, \dots, t\}$
6: **return** l'unico multiplo di $L \in (q + 1 - 2\sqrt{q}, q + 1 + 2\sqrt{q})$.

2.2 Algoritmo di Schoof

Data una curva ellittica E: $y^2 = x^3 + Ax + B$ su un campo finito $\mathbb{F}_q$, l'algoritmo di Schoof permette di calcolare la cardinalità di $E(\mathbb{F}_q)$ senza interpellare alcun punto.
L'idea è quella di sfruttare il calcolo nei gruppi di torsione.

2.2.1 Division polynomials

Dallo studio isogenie di una curva ellittica E su un campo K si ottiene che i gruppi di torsione sono di soli due tipi possibili:

- se $(m, \text{char}\, K) = 1$ o $\text{char}\, K = 0$ allora $E[m] = \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$
- se $p = \text{char}\, K$ e $r \geq 1$ allora $E[p^r] = \mathbb{Z}/p^r\mathbb{Z}$ oppure $E[p^r] = \{O\}$

Di questo fatto esiste anche una dimostrazione costruttiva che permette di scrivere esplicitamente le funzioni razionali che esprimono la moltiplicazione intera una volta fissate delle coordinate di Weierstrass.

Consideriamo A, B parametri fissati chiameremo **polinomi di divisione** $\psi_m \in \mathbb{Z}[x, y, A, B]$ costruiti induttivamente come segue:

$$\begin{aligned}
\psi_0 &= 0 \\
\psi_1 &= 1 \\
\psi_2 &= 2y \\
\psi_3 &= 3x^4 + 6Ax^2 + 12Bx - A^2 \\
\psi_4 &= 4y(x^6 + 5Ax^4 + 20Bx^3 - 5A^2x^2 - 4ABx - 8B^2 - A^3) \\
\psi_{2m+1} &= \psi_{m+2}\psi_m^3 - \psi_{m-1}\psi_{m+1}^3 && m \geq 2 \\
\psi_{2m} &= (2y)^{-1}(\psi_m)(\psi_{m+2}\psi_{m-1}^2 - \psi_{m-2}\psi_{m+1}^2) && m \geq 3
\end{aligned}$$

Alcuni risultati ([Was08] 3.2):

Lemma 2.11.

$$\psi_m \in \begin{cases} \mathbb{Z}[x, y^2, A, B] & m \equiv 1 \mod 2 \\ 2y\mathbb{Z}[x, y^2, A, B] & m \equiv 0 \mod 2 \end{cases}$$

Definizione 2.12.

$$\begin{aligned}
\phi_m &= x\psi_m^2 - \psi_{m+1}\psi_{m-1} \\
\omega_m &= (4y)^{-1}(\psi_{m+2}\psi_{m-1}^2 - \psi_{m-2}\psi_{m+1}^2)
\end{aligned}$$

Lemma 2.13. $\phi_m \in \mathbb{Z}[x, y^2, A, B]$ e

$$\omega_m \in \begin{cases} y\mathbb{Z}[x, y^2, A, B] & m \equiv 1 \mod 2 \\ \mathbb{Z}[x, y^2, A, B] & m \equiv 0 \mod 2 \end{cases}$$

Lemma 2.14.

- $\deg_x \phi_m = m^2$ e $lc_x\phi_m = 1$
- $\deg_x \psi_m^2 = m^2 - 1$ e $lc_x\psi_m^2 = m^2$

Teorema 2.15. Sia $E\colon y^2 = x^3 + Ax + B$ una curva ellittica su K di caratteristica diversa da 2 e $P = (x, y) \in E$. Se $n \geq 0$ allora

$$nP = \left(\frac{\phi_n(x)}{\psi_n^2(x)}, \frac{\omega_n(x, y)}{\psi_n^3(x, y)}\right)$$

Corollario 2.16. Sia $E\colon y^2 = x^3 + Ax + B$ una curva ellittica su $\mathbb{F}_q$ e $P = (x, y) \in E$. Se $m \geq 0$ dispari allora

$$P \in E[m] \iff \psi_m(x) = 0$$

2.2.2 Algoritmo

L'idea è calcolare la traccia del Frobenius a modulo abbastanza primi. *Come?* Ricordiamo che

$$\phi_q^2 - a\phi_q + q$$

è la mappa nulla. Se $P = (x, y) \in E$ allora

$$\phi_q^2 P - a\phi_q P + qP = O$$

$$(x^{q^2}, y^{q^2}) + (x_q, y_q) = a(x^q, y^q)$$

Studiando questa relazione sui punti di ℓ torsione possiamo trovare delle relazioni modulo ℓ. *Perché questo è conveniente?* Grazie ai division polynomials (teorema 2.15) per lavorare in $E[\ell]$ ci basta ridurre le relazioni modulo ψ_ℓ.

Correttezza. 2.2.2

(2-4): Se $\ell = 2$ dobbiamo capire se se la curva ha o meno ordine pari infatti $q+1$ è dispari e dunque $a \equiv \#E(\mathbb{F}_q) \mod 2$. Perciò bisogna controllare se ci sono o meno punti di 2 torsione. Vista l'equazione di Weierstrass un tale punto è del tipo $(e, 0)$ dunque trovarlo è equivalente a trovare una radice di $x^3 + Ax + B = 0$ in $\mathbb{F}_q$.
Come possiamo calcolarlo effettivamente?
Ricordiamo che

$$\mathbb{F}_q = \{x \in \overline{\mathbb{F}_q} \mid \ x^q - x = o\}$$

ci basta controllare se $x^q - x$ e $x^3 + Ax + B$ hanno radici comuni facendo un gcd.

(5): Sia $\ell \in \{2, 3, 5, \ldots, L\} \setminus \{\mathrm{char}\mathbb{F}_q\}$, consideriamo $P \in E[\ell]$ e posto q_ℓ il rappresentante di Q modulo ℓ in notazione bilanciata si ha

$$(x', y') = (x^{q^2}, y^{q^2}) + q_\ell(x, y) = (x^{q^2}, y^{q^2}) + (x_{q_\ell}, y_{q_\ell}) = a(x^q, y^q)$$

dunque studiando la coincidenza dei due membri si può trovare il valore di $a \mod \ell$.

(8-18): Dapprima supponiamo che $(x^{q^2}, y^{q^2}) \neq \pm(x_{q_\ell}, y_{q_\ell})$. In questo modo possiamo escludere a priori che $a \equiv 0 \mod \ell$ e confrontare (x', y') solo con i gli $(x_j^q, y_j^q) = j\phi(x, y)$ per $j \in \mathbb{Z}$ non multipli di ℓ.
Con lo scopo di ridurre il più possibile il dispendio di risorse, dapprima calcoliamo solo le componenti x per $j = 1, \ldots, (\ell - 1)/2$ e in caso di coincidenza le componenti y infatti $(x', y') = (x_j, \pm y_j) = \pm(x_j, y_j) = \pm j(x, y)$ ci dà il rappresentante in notazione bilanciata.
Come possiamo fare questi conti effettivamente?
Usando la formula della somma per punti distinti (è qui che uso l'ipotesi fatta all'inizio del punto) si trova che il coefficiente angolare della retta è

$$m = \frac{y^{q^2} - y_{q_\ell}}{x^{q^2} - x_{q_\ell}}$$

$$x' = m^2 - x^{q^2} - x_{q_\ell}$$

osserviamo però che dette $r_{1,i}(x), r_{2,i}(x)$ le funzioni razionali che definiscono la moltiplicazione per i e utilizzando l'equazione della curva

$$\begin{aligned}(y^{q^2} - y_{q_\ell})^2 &= (y^{q^2-1} - r_{2,q_\ell}(x))^2 y^2 \\ &= (y^{q^2-1} - r_{2,q_\ell}(x))^2 (x^3 + Ax + B) \\ &= ((x^3 + Ax + B)^{(q^2-1)/2} - r_{2,q_\ell}(x))^2 (x^3 + Ax + B)\end{aligned}$$

Allora $x' \in \mathbb{F}_q(x)$ e ha senso lavorare modulo ψ_ℓ.
A questo per fare il confronto tra le prime due coordinate ci basta controllare se $(x' - x_j^q) \equiv 0 \mod \psi_\ell$[1]. Volendo utilizzare la stessa strategia per confrontare le seconde coordinate, bisogna ricondurci a polinomi nella sola variabile x e in effetti

$$\frac{y'}{y}, \frac{y_j^q}{y} \in \mathbb{F}_q(x).$$

(19): Ci rimangono da vagliare i casi $(x^{q^2}, y^{q^2}) = \pm(x_{q_\ell}, y_{q_\ell})$.

(20-31): Se $(x^{q^2}, y^{q^2}) = -(x_{q_\ell}, y_{q_\ell})$ per ogni $P \in E[\ell]$ allora $E[\ell] = \{0\}$ e quindi $a = 0 \mod \ell$.
Invece se $(x^{q^2}, y^{q^2}) = (x_{q_\ell}, y_{q_\ell})$ per ogni $P \in E[\ell]$ allora

$$\begin{aligned}\phi_q^2(x,y) &= q(x,y) \\ a\phi(x,y) &= 2q(x,y) \\ a^2 q(x,y) &= a^2\phi^2(x,y) = (2q)^2(x,y)\end{aligned}$$

perciò se q è un quadrato modulo ℓ esiste w tale che $q \equiv w^2 \mod \ell$ dunque da $a^2 q \equiv a^2 w^2 \equiv (2q)^2 \equiv 4w^4$ otteniamo che $a \equiv \pm 2w \mod \ell$. Per decidere il segno, possiamo usare il fatto che per ogni punto

$$(\phi_q^2 - q)(x,y) = (\phi_q - w)(\phi_q + w)(x,y) = O$$

Prendendo quindi un punto $P \in E[\ell]$ si deve avere che o $P' = (\phi_q + w)P = O$ oppure $(\phi_q - w)P' = O$; in particolare deve esistere un punto T tra questi due tale che $\phi_q T = \pm wT$ e dunque, per capire il segno e se questo caso si verifica, ci basta sapere se per qualche $(x,y) \in E[\ell]$ vale

$$(x^q, y^q) = \pm(x_w, y_w) = (x_w, \pm y_w). \tag{2.1}$$

Come possiamo fare questi conti effettivamente?
La prima cosa da controllare è se q è un residuo modulo ℓ (20) e trovare eventualmente la sua radice w. Per far questo si possono utilizzare alcune tecniche note come il crivello quadratico. In caso affermativo bisogna capire se esiste un punto il $E[\ell]$ che soddisfa l'Equazione 2.1; utilizzando la strategia già vista nei punti soprastanti, prima si controlla la prima coordinata (23) ed eventualmente la seconda per controllare il segno (25).

[1] Le radici di ψ_ℓ sono semplici visto che la cardinalità di $E[\ell] \setminus O$ è $(\ell^2 - 1)/2$ che per il Lemma 2.14 è anche il grado di ϕ_ℓ.

Osservazione 2.4. Qui è importante usare il gcd e non la congruenza modulo ψ_ℓ perché ci serve che esista un punto e non che ogni punto soddisfi l'equazione. Infatti dire che la traccia del Frobenius è $2w$ non garantisce che per ogni P

$$\phi P = \pm wP$$

come ad esempio se la matrice di $\phi|_{E[\ell]}$ è

$$\begin{bmatrix} w & 1 \\ 0 & w \end{bmatrix}$$

(34): È una banale applicazione del teorema cinese del resto.

□

Complessità. 2.2.2 Facendo un'analisi prettamente teorica, nel senso che implementando in maniera furba questo numero può essere migliorato, la complessità è $O(\log^8 q)$.
Diamo un'idea del perché. Per calcolare $\phi(P)$ e $\phi^2(P)$ dobbiamo elevare a potenza x e y per ogni primo della lista S, ossia dobbiamo esponenziarli in

$$R = {}^{\mathbb{F}_q[x,y]}\!\!\diagup_{(y^2-(x^3+Ax+B),\psi_\ell)}$$

il che richiede $O(\log q)$ moltiplicazioni: i polinomi di R hanno grado in x minore di $(\ell^2-1)/2$ che è dell'ordine $O(\log^2 q)$ e quindi ogni moltiplicazione richiede $O(\log^4 q)$ operazioni in $\mathbb{F}_q$. A sua volta in termini di bit le operazioni in $\mathbb{F}_q$ costano $O(\log^2 q)$, per un totale quindi di $O(\log^7 q)$ per ogni primo $\ell \in S$. La cardinalità di S è di ordine logaritmico in q e dunque la complessità è $O(\log^8 q)$. □

Osservazione 2.5. Se q è un numero grande invece di elevare alla q-esima potenza si può lavore con un rappresentante

$$x_q \equiv x^q \mod x^3 + Ax + B.$$

Algorithm 2.3 Algoritmo di Schoof

Input: E: $y^2 = x^3 + Ax + B$ curva ellittica su $\mathbb{F}_q$.
Output: $\#E(\mathbb{F}_q) = q + 1 - a$.
Note: con $Num(f)$ indichiamo la procedura che restituisce il numeratore della frazione f.

$S = \{2,3,5,\ldots,L\} \setminus \{\text{char}\,\mathbb{F}_q\}$ in modo che $\prod_{\ell\in S}\ell > 4\sqrt{q}$
if $\ell = 2$ **then**
    $a \equiv 0 \mod 2 \iff \gcd(x^3 + Ax + B, x^q - x) \neq 1$
end if
for $\ell \in S$ **do**
    trova q_ℓ il rappresentante di Q modulo ℓ in notazione bilanciata
    calcola la coordinata x di

$$(x', y') = (x^{q^2}, y^{q^2}) + q_\ell(x, y)$$

    for $j = 1,\ldots,(\ell-1)/2$ **do**
        calcola la coordinata x di $(x_j, y_j) = j(x, y)$
        if $(x' - x_j^q) \equiv 0 \mod \psi_\ell$ **then**
            calcola y e y_j
            if $(y' - y_j^q)/y \equiv 0 \mod \psi_\ell$ **then**
                $a \equiv j \mod \ell$
            else $a \equiv -j \mod \ell$
            end if
        else j++
        end if
    end for
    if $j = (\ell+1)/2$ **then**
        if $\nexists w^2 \equiv q \mod \ell$ **then**
            $a \equiv 0 \mod \ell$
        else
            if $\gcd(Num(x_q - x_w), \psi_\ell) = 1$ **then**
                $a \equiv 0 \mod \ell$
            else $G = \gcd(Num((y^q - y_w)/y), \psi_\ell)$
                if $G \neq 1$ **then**
                    $a \equiv 2w \mod \ell$
                else $a \equiv -2w \mod \ell$
                end if
            end if
        end if
    end if
end for
calcola $a \mod \prod_{\ell\in S}\ell$ e tale che $|a| < 2\sqrt{q}$
return $q + 1 - a$

Capitolo 3

Curve ellittiche supersingolari

L'anello degli endomorfismi endomorfismi di una curva ellittica è ha una classe di isomorfismo ben definita

Proposizione 3.1. Sia E una curva ellittica su un campo K allora vale una delle seguenti:

- $\mathrm{End}(E) \simeq \mathbb{Z}$
- $\mathrm{End}(E)$ è isomorfo ad un ordine di campo di numeri quadratico immaginario.
- $\mathrm{End}(E)$ è isomorfo ad un ordine di un'algebra di quaternioni.

Conoscere il gruppo degli endomorfismi può aiutare nello studio della curva, ad esempio sapere che è del secondo tipo può facilitare ECDLP.
Nel caso in cui K abbia caratteristica zero la terza opzioni non si verifica mai, invece nel caso dei campi finiti si. In questo capitolo faremo veder come, dal punto di vista della crittoanalisi, questa eventualità sia positiva. Per fare questo introdurremo delle proprietà equivalente e alcuni criteri per capire quando si verifica o meno.

Teorema 3.2. Sia E una curva ellittica su un campo $\mathbb{F}_q$ con $p = \mathrm{char}\mathbb{F}_q > 0$. Allora sono equivalenti i seguenti fatti:

i. $E[p^r] = \{O\}\ \forall r \geq 1$.

ii. Dato ϕ_q il q-frobenius, allora la sua isogenia duale $\widehat{\phi_q}$ è (puramente) inseparabile.

iii. L'endomorfismo di moltiplicazione per p è puramente inseparabile e $j(E) \in \mathbb{F}_{q^2}$.

iv. $\mathrm{End}(E)$ è isomorfo ad un ordine di un'algebra di quaternioni.

Definizione 3.3. Sia E una curva ellittica su un campo $\mathbb{F}_q$, se valgono le condizioni i.-iv. del teorema 3.2 diremo che è **supersingolare** o che ha **invariante di Hasse** O, altrimenti è detta **ordinaria** o con invariante di Hasse 1.

Osservazione 3.1. Notiamo che essere supersingolare è un proprietà che non dipende dal campo i definizione ma è una proprietà della curva definita sulla chiusura algebrica.

3.1 Calcolo invariante di Hasse

Sebbene le caratterizzazioni date siano pregni di significato a livello teorico, è utile avere dei criteri pratici per stabilire quando una curva è supersingolare.

Teorema 3.4. Sia E una curva ellittica su un campo $\mathbb{F}_q$ con $p = \mathrm{char}\,\mathbb{F}_q > 2$.

(a) Se $E\colon y^2 = f(x)$ con $f \in \mathbb{F}_q[x]$ polinomio cubico le cui radici sono distinte. Allora E è supersingolare se e solo se il coefficiente di x^p nello sviluppo di $f(x)^{\frac{p-1}{2}}$ è nullo.

(b) Sia $E = E_\lambda\colon y^2 = x(x-1)(x-\lambda)$, curva ellittica in forma di Legendre, con $\lambda \in \overline{\mathbb{F}_q} \setminus \{0,1\}$ e sia $m = (p-1)/2$ e

$$H_p(t) = \sum_{i=0}^{m} \binom{m}{i}^2 t^i$$

Allora E è supersingolare se e solo se $H_p(\lambda) = 0$.

È interessante anche capire quanto spesso si ottiene una curva supersingolare.

Corollario 3.5. Il polinomio $H_p(t)$ ha radici distinte. In particolare esiste solo una curva supersingolare in caratteristica 3, mentre se $p > 3$ il numero si curve supersingolari a meno di isomorfismo (sulla chiusura algebrica) è

$$\left[\frac{p}{12}\right] + \begin{cases} 0 & if\ p \equiv 1 \mod 12 \\ 1 & if\ p \equiv 5 \mod 12 \\ 1 & if\ p \equiv 7 \mod 12 \\ 2 & if\ p \equiv 11 \mod 12 \end{cases}$$

Per la dimostrazione nei dettagli vedere [Sil08] §V.4 oppure [Was08] §4.6. Diamo però una veloce scorsa ai fatti, alcuni dei quali servono anche per la dimostrazione ma sono importanti anche di per sé.

Proposizione 3.6. Sia E una curva ellittica su un campo $\mathbb{F}_q$. E è supersingolare se e solo se la traccia del q-frobenius a è congrua a zero modulo p.

Dimostrazione. Osserviamo che $a = \phi_q + \widehat{\phi_q}$ o meglio $\widehat{\phi_q} = -a + \phi_q$. Per il Corollario 5.5 [Sil08] una mappa del tipo $n + m\phi_q$ è separabile se e solo se $p \nmid n$, da cui la tesi. □

Corollario 3.7. Se $p \geq 5$ ed E è definita su $\mathbb{F}_p$ allora è supersingolare se e solo se $a = 0$. In tal caso $\#E(\mathbb{F}_q) = p + 1$.

Proposizione 3.8. Sia $p \geq 5$. Allora la curva ellittica $y^2 = x^3 + 1$ su $\mathbb{F}_p$ è supersingolare se e solo se $p \equiv 2 \mod 3$ e la curva ellittica $y^2 = x^3 + x$ su $\mathbb{F}_p$ è supersingolare se e solo se $p \equiv 3 \mod 4$.

Proposizione 3.9. Se q è dispari e $q \equiv 2 \mod 3$ allora tutte le curve ellittica E con equazione di Weierstrass $y^2 = x^3 + B$ con $B \in \mathbb{F}_q^*$ sono supersingolari.

Osservazione 3.2. Vediamo l'idea per dimostrazione il teorema 3.4.a.
Dal teorema 2.6 si ha che

$$\#E(\mathbb{F}_q) = q + 1 + \sum_{x \in \mathbb{F}_q} \left(\frac{f(x)}{\mathbb{F}_q} \right) = q + 1 + \sum_{x \in \mathbb{F}_q} f(x)^{\frac{q-1}{2}}$$

ricordando che vale

$$\sum_{x \in \mathbb{F}_q} x^i = \begin{cases} -1 & if\ q-1 \mid i \\ 0 & if\ q-1 \nmid i \end{cases}$$

e usando che il polinomio f è cubico l'unico termine che sopravvive dopo la somma è il coefficiente di x^{q-1} che indicheremo con A_q. Perció data la relazione

$$q + 1 - a = q + 1 + A_q$$

deve valere $a \equiv -A_q \mod p$.
Infine A_p il coefficiente di x^p nello sviluppo di $f(x)^{\frac{p-1}{2}}$ si ha che[1] per ogni $r > 0$

$$A_{p^{r+1}} = A_{p^r} A_p^{p^r}$$

e quindi $A_p \equiv 0 \mod p$ se e solo se $A_q \equiv 0 \mod p$ se e solo se $a \equiv 0 \mod p$ se e solo se E è supersingolare.

3.2 Vantaggi computazionali

Uno dei motivi per cui sono interessanti le curve supersingolari è che la motiplicazione intera può essere fatta in alcuni casi più veolocemente.
consideriamo E una curva supersingolare su $\mathbb{F}_q$ e un suo punto $P = (x, y) \in E(\mathbb{F}_{q^n})$ per qualche intero n positivo e tendenzialmente grande. Sia $k \in \mathbb{N}$, vogliamo calcolare kP. L'algoritmo standar è

[1] si usa ancora il fatto che f è cubico.

Algorithm 3.1 Moltiplicazione per raddoppiamenti successivi

Input: E curva ellittica su $\mathbb{F}_q$, $k \in \mathbb{N}$ e $P = (x, y) \in E(\mathbb{F}_{q^n})$.
Output: kP.

1: $a = k$, $B = O$ e $C = P$
2: **while** $a \neq 0$ **do**
3: **if** a pari **then**
4: $a = a/2$
5: $B = B + C$
6: $C = 2C$
7: **end if**
8: **if** a dispari **then**
9: $a = a - 1$
10: $B = B + C$
11: $C = C$
12: **end if**
13: **end while**
14: **return** B

che ha costo logaritmico.
Tuttavia nel caso di una curva supersingolare possiamo fare meglio. Supponiamo che $a = 0$, allora

$$\phi_q^2 + q = 0$$

dunque

$$qP = -\phi_q^2 P = (x^{q^2}, -y^{q^2})$$

e

$$q^i P = (-1)^i \phi_q^{2i} P = (x^{q^{2i}}, (-1)^i y^{q^{2i}})$$

e quindi per fare la moltiplicazione per q ci si può ridurre all'elevamento a potenza sul campo finito. In generale quello che si fa è:

Algorithm 3.2 Moltiplicazione intera curve supersingolari

Input: E: $y^2 = X^3 + Ax + B$ curva ellittica su $\mathbb{F}_q$ supersingolare, $k \in \mathbb{N}$ e $P = (x, y) \in E(\mathbb{F}_{q^n})$.
Output: kP.

1: Espandi k in base q

$$k = k_0 + k_1 q + \cdots + k_s q^s$$

con $0 \leq k_i < q$
2: calcola $k_i P = (x_i, y_i)$ per ogni i
3: calcola $q^i k_i P = (x_i^{q^2}, (-1)^i y_i^{q^2})$
4: **return** $\sum_{i=0}^{s} q^i k_i P$

Vedremo nei capitoli successivi ulteriori vantaggi, o svantaggi a seconda della prospettiva, di questo tipo di curve.

3.3 Riduzione modulo p

Sia E una curva ellittica su $\mathbb{Z}$ con moltiplicazione complessa, ossia tale che l'anello degli endomorfismi è strettamente più grande di $\mathbb{Z}$ e visto che la caratteristica qui è zero sará isomorfo ad un ordine di campo di numeri quadratico immaginario $\mathbb{Q}(\sqrt{-d})$.
Consideriamo $p \nmid d$ un primo dispari e tale che la curva ridotta modulo p

$$\begin{array}{rccc} \mathrm{red}_p: & E(\mathbb{Q}) & \to & \tilde{E}(\mathbb{Z}/p\mathbb{Z}) \\ & [x,y,z] & \mapsto & [x,y,z] \mod p \end{array}$$

sia una curva ellittica (in pratica stiamo dicendo che completando a $\mathbb{Q}_p$ si ha *buona riduzione*). Allora $\tilde{E}(\mathbb{Z}/p\mathbb{Z})$ è supersingolare se e solo se $-d$ non è un residuo quadratico modulo p. Questo vuol dire che in questo caso circa la metà delle curve ridotte sono supersingolari; se invece $\mathrm{End}(E) \simeq \mathbb{Z}$ l'insieme delle curve a riduzione supersingolare è più sparso.q

Capitolo 4

Problema del logaritmo discreto

Problema del logaritmo discreto LDP: Dato un gruppo $(G,\cdot)$ e due elementi a,b, dire se esiste e trovare $k \in \mathbb{Z}$ tale che $a^k = b$.

Sebbene sia generale il problema del **logaritmo discreto** spesso è dato sui campo finiti $\mathbb{F}_q$, in tale contesto allora ha senso non tanto trovare un intero k ma una classe modulo $q-1$. In questo capitolo studieremo soprattutto il **problema del logaritmico discreto su curve ellittiche ECDLP**, il quale è molto utile in ambito crittografico. In questo capitolo presenteremo prima l'index calculus che è il migliore algoritmo sui campi finiti, poi alcune strategie generali e infine particolari attacchi sulle curve ellittiche.

4.1 Index Calculus

L'index calculus è l'analogo del crivello quadratico per la fattorizzazione. L'idea è quella di risolvere il problema del logaritmo su istanze meno complesse e dispendiose e combinarle (usando l'algebra lineare) in modo opportuno.
Introduciamo come prima cosa una nozione che ci serve e che non è propria solo di questo algoritmo ma di molte procedure:

Definizione 4.1. Sia $x \in \mathbb{Z}$ un intero e $B \in \mathbb{N}^+$, diremo che è B**-liscio** se si fattorizza in modo unico come prodotto di numeri primi ℓ minori di B

$$x = \prod_{\ell < B} \ell^{e_\ell}.$$

A grandi linee il procedimento dell'index calculus è il seguente. Sia p un numero primo e g un generatore di $\mathbb{F}_p^*$ e $b \in \mathbb{F}_p^*$ un elemento qualsiasi :

1. risolviamo il problema del logaritmo per un po' di $a_i \in \mathbb{F}_p^*$

   $$g^{k_i} \equiv a_i = \prod_{\ell < B} \ell^{e_{\ell,i}} \mod p$$

   che siano B-lisci per un certo B fissato in modo da ottenere relazioni lineari tra i logaritmi dei primi minori di B

   $$k_i \equiv \sum_{\ell < B} e_{\ell,i} L(\ell) \mod p-1$$

2. troviamo $j \in \{1, \dots, p-1\}$ tale che $g^{-j} b = c$ sia B-liscio e quindi

$$c = \prod_{\ell < B} \ell^{e_{\ell,c}} \mod p$$

da cui

$$L(b) - j \equiv L(c) \equiv \sum_{\ell < B} e_{\ell,c} L(\ell) \mod p-1$$

dove notiamo $L(b) = k$ è il numero che stiamo cercando;

3. facendo combinazioni lineari delle equazioni

$$k_i \equiv \sum_{\ell} e_{\ell,i} L(\ell) \mod p-1$$

risolviamo

$$k \equiv j + \sum_{\ell} e_{\ell,c} L(\ell) \mod p-1.$$

Osservazione 4.1.

- Il problema originario viene riportato alla manipolazioni di equazioni lineari.
- Il costo temporale è subesponenziale $O(\exp(\sqrt{2 \ln p \ln \ln p}))$.
- Quello che stiamo usando "pesantemente" è che siamo in un dominio a fattorizzazione unica.

4.2 Attacchi generali LDP

Qui utilizzeremo la notazione additiva perché abbiamo in mente un curva ellittica, tuttavia queste strategie valgono su un qualsiasi gruppo sensato. A meno di diverse indicazioni, assumeremo P un generatore del gruppo e che l'ordine N del gruppo sia noto (nel caso delle curve ellittiche abbiamo l'algoritmo di Schoof ad esempio).

4.2.1 Baby Step-Giant Step

Il problema del calcolo dell'ordine di un punto P su una curva ellittica non è altro un'istanza particolare di ECDLP:

$$kP = O.$$

La stategia *Baby Step-Giant Step* descritta nel capitolo precedente può essere reinterpretata nel caso generale.

Algorithm 4.1 Baby Step-Giant Step LDP

Input: G gruppo, $P, Q \in G$ e ordine del gruppo N.
Output: $k \in \mathbb{Z}^+$ tale che $kP = Q$.

1: sceglie $m \geq \sqrt{N}$ (buon approssimante) e calcola mP
2: calcola l'insieme $J = \{jP\}_{j=0,\dots,m-1}$
3: trovato=FALSE e $h = 0$
4: **while** trovato=TRUE & $h \leq m$ **do**
5: $R = Q - mhP$
6: **if** $\exists j$: $R = jP \in J$ **then**
7: trovato=TRUE
8: **else**h++
9: **end if**
10: **end while**
11: $k = hm + j \mod N$
12: **return** k

Correttezza. 4.2.1 *Perché c'è un match e il ciclo while termina?*
Visto che $m^2 \geq N$ allora $k = hm + j$ modulo N restituisce un numero minore $k < N \leq m^2$ e quindi possiamo scrivere $k = k_0 + mk_1$ con $k \equiv k_0 \mod m$. Allora ponendo $h = k_1$ e $j = k_0$ si ha il match infatti

$$Q - hmP = Q - k_1 mP = kP - k_1 mP = k_0 P = jP.$$

□

Complessità. 4.2.1 L'algoritmo richede $O(\sqrt{N})$ passi e $O(\sqrt{N})$ spazio memoria. □

Osservazione 4.2.

- Per calcola i baby step e i giant step possiamo usare la proprietà associativa.
- Anche se non conosciamo esattamente N ci basta un upper buond (nelle curve ellittiche possiamo usare il teorema di Hasse).
- Per risparmiare spazio nel caso delle curve ellittiche possiamo utilizzare la notazione bilanciata e la prima coordinata e poi usare la seconda per il segno.

4.2.2 ρ di Pollard e Metodo λ

Uno degli svantaggi del metodo Baby Step-Giant Step è il costo in termini di memoria. ρ di Pollard e Metodo λ riducono quest'ultimo impiegando circa lo stesso tempo.

Capiamo l'idea di fondo. Sia G un gruppo finito di ordine N e $f: G \to G$ una funzione pseudocasuale. Costruiamo una successione:

$$\begin{cases} a_0 \in G \\ a_n = f(a_{n-1}) \end{cases}$$

Visto che il gruppo è finito, da un certo punto in poi la funzione diventerà periodica, ossia per $n > h$ abbastanza grande $a_n = a_{n+T}$. Questo h ovviamente dipende da a_0, ma al caso pessimo è $\sqrt{N}$.
Abbiamo trovato un metodo che dopo un certo numero di passi ci dà un match. Alcune osservazioni:

- Invece di registrare tutta la successione se ne posso lanciare due con velocità multipla: ad esempio $\{a_i\}$ e $\{a_{2i}\}$ e controllare di volta in volta se $a_i = a_{2i}$. Questo richiede magari più passi, ma non sostanzialmente di più.

- Un'altra strategia è quella di registrare i punti *distinti*, ossia che soddisfano una certa proprietà che nel caso delle curve ellittiche può essere ad esempio la prima coordinata nulla.

Discutiamo quindi la scelta di f più opportuna rispetto al nostro obiettivo: trovare $k \in \mathbb{Z}$ tale che $kP = Q$.
Consideriamo una partizione di G in s sottogruppo (normalmente si sceglie $s \simeq 20$)

$$G = S_1 \sqcup \cdots \sqcup S_s$$

e due interi $2s$ interi $a_i, b_i \mod N$ tali che

$$M_i = a_i P + b_i Q \in G.$$

siano distinti. Allora definiamo per ogni $g \in S_i$

$$f(g) = g + M_i.$$

Siamo pronti per descrivere i passi dell'algoritmo ρ **di Pollard**.

1. Scegliamo a $a_0, b_0 \mod N$ e poniamo $P_0 = a_0 P + b_0 Q$

2. Calcoliamo la successione $P_i = f(P_{i-1})$ con punto iniziale P_0 finché non troviamo $P_r = P_t$. Allora
$$u_r P + v_r Q = u_t P + v_t Q$$
e quindi
$$(u_r - u_t)P = (v_t - v_r)Q$$

3. Calcola $\gcd(v_t - v_r, N) = d$
$$k \equiv (v_t - v_r)^{-1}(u_r - u_t) \mod \frac{N}{d}$$

4. Calcoliamo $(k+i)P$ per $i = 0, \ldots, d-1$ finché non troviamo Q.

Spieghiamo perché è corretto e facciamo alcune osservazioni:

Osservazione 4.3 (ρ di Pollard).

- *Come si realizza effettivamente il passo 2?*
  Dato un punto R trovare la scrittura in termini di P, Q non è sempre possibile né facile. In questo caso pero ad ogni passo, vista la condizione iniziale, si ha che se $P_i \in S_l$

$$\begin{aligned} P_{i+1} &= P_i + a_l P + b_l Q = u_i P + v_i Q + a_l P + b_l Q \\ &= (a_l + u_i)P + (b_l + v_i)Q \\ &= u_{i+1} P + v_{i+1} Q \end{aligned}$$

  Allora più che la successioni di punti ci basta tenere traccia delle coordinate:

$$\begin{cases} (a_0, b_0) \\ (u_{i+1}, v_{i+1}) = (a_{j_i}, b_{j_i}) + (u_i, v_i) \end{cases}$$

- Normalmente i gruppi che si vanno a considerare hanno pochi fattori o addirittura N è primo, in quest'ultimo caso o $d = N$ e quindi non abbiamo ottenuto informazioni oppure $d = 1$ e quindi il passo 3 è banale.

Il **metodo** λ si basa sullo stesso principio messo però in parallelo. Invece di scegliere un punto iniziale si scelgono $P_0^1, \ldots, P_0^r$ e si calcolano le rispettive sequenze, magari su macchine diverse,

$$P_i^j = f(P_{i-1}^j)$$

per $j = 1, \ldots, r$ e si controllano le coincidenze tra queste.
Anche in questo caso invece di conservare tutte le successioni può essere conveniente registrare solo i punti distinti.

Osservazione 4.4. Notiamo che sebbene abbiano gli stessi costi Baby Step-Giant Step e ρ di Pollard si basano su principi totalmente diversi: il primo è un **algoritmo deterministico** il secondo è un **algoritmo probabilistico**.

4.2.3 Poligh-Hellman

Per l'attacco PH supponiamo di conoscere N l'ordine di $P \in G$ e la sua fattorizzazione intera

$$N = \prod q^e$$

e dato un punto $Q \in G$ vogliamo trovare k che risolvere LDP per P, Q.
L'idea è trovare che k modulo q^e per ogni fattore e poi usare il teorema cinese del resto. In particolare fissato q^e

$$k \equiv k_0 + k_1 q + \cdots + k_{e-1} q^{e-1} \mod q^e$$

vogliamo trovare i k_i.

Algorithm 4.2 Poligh-Hellman

Input: G gruppo, $P, Q \in G$ e N ordine di P, $F = \{(q,e) \mid q^e \text{ divide esattamente } N\}$.
Output: $k \in \mathbb{Z}^+$ tale che $kP = Q$.

1: **while** $F \neq \varnothing$ **do**
2: $T = \{j\left(\frac{N}{q}P\right) \mid j = 0,\dots,q-1\}$
3: Calcola $k_0 \in \{0,\dots,q-1\} : \frac{N}{q}Q = k_0\frac{N}{q}P$
4: **for** $i = 0,\dots,e-1$ **do**
5: $Q_i = Q_{i-1} - k_{i-1}q^{i-1}P$
6: Calcola $k_i \in \{0,\dots,q-1\} : \frac{N}{q^{i+1}}Q_i = k_i\frac{N}{q}P$
7: **end for**
8: $k \equiv k_0 + k_1 q + \cdots + k_{e-1}q^{e-1} \mod q^e$
9: $F = F \setminus \{(q,e)\}$
10: **end while**
11: trova k con il TCR
12: **return** k

Correttezza. 4.2.3

(3:) $\frac{N}{q}Q \in T$, infatti

$$\frac{N}{q}Q = \frac{N}{q}kP = \frac{N}{q}(k_0 + k_1 q + \dots)P = \frac{N}{q}k_0P + (k_1 + \dots)NP = \frac{N}{q}k_0P \in T$$

(6:) Usando il punto precedente come base per l'induzione si ha

$$\begin{aligned}\frac{N}{q^{i+1}}Q_i &= \frac{N}{q^{i+1}}(Q_{i-1} - k_{i-1}q^{i-1}P)\\ &= \frac{N}{q^{i+1}}(k_i q^i + k_{i+1}q^{i+1} + \dots)P\\ &= \frac{N}{q}k_iP + (k_{i+1}q^i + \dots)NP\\ &= \frac{N}{q}k_iP \in T\end{aligned}$$

□

Osservazione 4.5. Se i primi che dividono N sono piccoli l'algoritmo funziona bene, se invece esiste un fattore che in termini di bit è vicino a N il calcolo di T diventa equipollente a LDP.

In crittografia noi siamo interessanti a gruppi su cui DLP è difficile. Per far sì che il sistema sia resistente all'attacco PH quindi serve che in G esista un elementi il cui ordine è diviso da un primo q molto grande. In questo caso è lecito chiedersi: *quale è la probabilità di trovare un tale punto random?* La probabilità che l'ordine sia divisibile per q è

$$1 - \frac{1}{q}$$

e quindi più grande è q maggiore è la probabilità.
Trovato un tale P' il cui ordine è mq si può lavorare su G' il gruppo generato da mP' che ha ordine esattamente q.

4.3 ECDLP: attacchi con i pairings

Le curve ellittiche sono dei gruppi che hanno anche un altro tipo di struttura. In questa sezione faremo vedere due attacchi specifici per le curve ellittiche sui campi finiti.

4.3.1 Attacco MOV

L'attacco MOV (Menezes, Okamoto,Vanstone) utilizza il Weil pairing su una curva ellittica $E(\mathbb{F}_q)$ per trasporre il problema del logaritmo discreto sul campo finito $\mathbb{F}_q$, dove si può usare l'index calculus.

Sia E un curva ellittica su $\mathbb{F}_q$ e $P,Q \in E(\mathbb{F}_q)$: vogliamo trovare k tale che $kP = Q$. La prima cosa da fare è trovare un criterio che certifichi che un tale k esista.

Lemma 4.2. Se N è l'ordine di P e $\gcd(N,q) = 1$. Allora

$$\exists k\colon kP = Q \iff NQ = O \wedge e_N(P,Q) = 1$$

Dimostrazione. Se $kP = Q$ allora $NQ = kNP = O$ e $e_N(P,Q) = e_N(P,P)^k = 1^k = 1$. Viceversa visto che $\gcd(N,q) = 1$ allora $E[N] = \mathbb{Z}_N^2$ possiamo prendere R che completi P a base. Allora esistono due interi tali che

$$Q = aP + bR.$$

Per il Teorema 1.1

$$1 = e_N(P,Q) = e_N(P, aP + bR) = e_N(P,P)^a e_N(P,R)^b = e_N(P,R)^b$$

allora $e_N(P,R)^b = 1$. Dal fatto che sono una base si ha che il loro accoppiamento è una radice primitiva N-esima dell'unità

$$e_N(P,R)^b = \zeta_N^b = 1$$

e quindi $N \mid b$. Perciò $Q = aP + bR = aP + b'NR = aP$ dal momento in cui $NR = O$. □

I gruppi di torsione sono definiti sulla chiusura algebrica, così come i $\mu_N \subseteq \overline{\mathbb{F}_q}$. In pratica noi però vogliamo lavorare però su campi finiti, possiamo sfruttare allora il Corollario 1.2: ci basterà lavorare su $E(\mathbb{F}_{q^m})$ per m abbastanza grande affinché quest'ultima contenga tutto il gruppo di N torsione.

Vediamo adesso l'algoritmo in dettaglio:

Algorithm 4.3 Attacco MOV

Input: $P, Q \in E$ curva ellittica su $\mathbb{F}_q$, tali per cui ECDLP abbia soluzione e N ordine di P.
Output: k: $kP = Q$.
Note: Tutti i calcoli si intendono svolti su $E(\mathbb{F}_{q^m})$ per m opportuno.

1: $G = 1$
2: **while** $G = N$ **do**
3: Scegli $S \in E(\mathbb{F}_{q^m})$ random.
4: Calcola ordine M di S
5: $d = \gcd(N, M)$
6: $T = \frac{M}{d} S$
7: Calcola $e_N(P, T) = \zeta_P$
8: Calcola $e_N(Q, T) = \zeta_Q$
9: Trova k' tale che $(\zeta_P)^{k'} = \zeta_Q$ in $\mathbb{F}_{q^m}^*$
10: $k = TCR(G, d, k, k')$
11: $G = \gcd(G, d)$
12: **end while**
13: **return** k

Correttezza. 4.3.1

(3-6): Per poter utilizzare l'accoppiamento di Weil abbiamo bisogno di punti in $E[N]$.

(7-9): Visto che l'ordine di T è d allora $\zeta_P, \zeta_Q \in \mu_d$, dunque usando la bilinearità si ha che

$$\zeta_d^b = \zeta_Q = e_N(Q, T) = e_N(kP, T) = \zeta_P^k = \zeta_d^{ak}$$

e quindi $ka \equiv b \mod d$.

(2),(10-13): Per poter determinare k ci serve che la relazione valga modulo N, dunque usando il TCR possiamo ricostruirla a partire da un po' di $d \mid N$ tali che $\gcd(d_i \mid i \in I) = N$.

□

Complessità. 4.3.1 Il costo del corpo del ciclo è dato da calcolo di due accoppiamenti più il costo della soluzione del logaritmico discreto sui campi finiti, per cui rimandiamo rispettivamente al capitolo 1.1 e alle sezioni precedenti di questo capitolo.
Quello che è opportuno qui valutare è invece il numero di cicli While. Facciamo una stima della probabilità che $d > 1$, che è il caso in cui il loop non è "inultile". Sappiamo che esistono $n_1 \mid n_2$ tali che

$$E(\mathbb{F}_{q^m}) = \mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/n_2\mathbb{Z}$$

prendiamo $\{B_1, B_2\}$ una base con n_i ordine di B_i. Allora

$$S = uB_1 + vB_2$$

e se ℓ è un primo tale che $\ell^e \parallel N$ allora esiste $f \geq e$ tale che $\ell^f \parallel n_2$, perciò se $\ell \nmid v$ l'ordine di S M deve essere diviso perℓ^f e quindi $\ell^e \mid \gcd(N, M) = d$, ossia siamo nel caso buono. La probabilità che ℓ non divida v è $1 - 1/\ell$. Se N ha fattori primi molto grandi allora la probabilità che $d > 1$ è più vicina ad 1. □

Facciamo alcune osservazioni. Abbiamo ribadito più volte che per poter fare le operazioni sui campi finiti ci può servire estendere il campo. Una buona strategia contro l'attacco MOV è quella di scegliere una curva per cui l'm opportuno è molto grande e estendere il campo comporta dei costi equivalenti a risolvere ECDLP con delle moltiplicazioni.
Purtroppo se la curva è supersingolare questo tentativo fallisce:

Proposizione 4.3. Sia E una curva ellittica su $\mathbb{F}_q$ tale che la traccia del frobenius sia nulla. Allora se esiste $P \in E(\mathbb{F}_q)$ di ordine N si ha $E[N] \subseteq E(\mathbb{F}_{q^2})$.

Dimostrazione. Facciamo vedere che $E[N]$ è fissato da ϕ_{q^2}. Sia $S \in E[N]$ qualsiasi, allora $\phi_{q^2}(S) = -qS$ visto che $a = 0$; inoltre $N \mid \#E(\mathbb{F}_q) = q+1$ si ha che $-q \equiv -q+q+1 \equiv 1 \mod N$ e dunque

$$\phi_{q^2}(S) = -qS = S.$$

□

Osservazione 4.6. Se $a \neq 0$, l'idea sopra funziona comunque probabilmente con $m = 3, 4, 6$.

4.3.2 Attacco Frey-Rüch

L'accoppiamento di Tate-Lichtenbaum a livello computazionale è più conveniente di quello di Weil. Frey e Rüch hanno pensato di ricalcare la strategia dell'attacco MOV usando però Tate-Lichtenbaum.
Come nel Lemma 1.7, sia ℓ un primo tale che $\ell \mid q-1$, $\ell \mid \#E(\mathbb{F}_q)$ e $\ell^2 \nmid \#E(\mathbb{F}_q)$ e sia P un generatore di $E(\mathbb{F}_q)[\ell]$. Vogliamo trovare k tale che $kP = Q$:

$$\tau_\ell(Q,P) = \tau_\ell(kP,P) = \tau_\ell(P,P)^k = \zeta_\ell^k.$$

Risolvendo il problema logaritmo discreto su $\mathbb{F}_q$ quindi possiamo determinare $k \mod \ell$.

Anche in questo caso è da domandarsi: *come scegliere una curva ellittica resistente all'attacco Frey-Rüch?*
Bisogna scegliere in modo che esista un punto il cui ordine sia un primo $\ell >> 0$ e $\ell \nmid q-1$; dobbiamo prevenire, come nell'attacco MOV, che questa seconda ipotesi venga a cadere e quindi che $q^m \not\equiv 1 \mod \ell$.

A conclusione, ricordando la Proposizione 1.8, sottolineiamo che se il punto che prendiamo ha ordine un primo le ipotesi affinché tutto funzioni sono per l'attacco MOV e equivalenti a quelle del'attacco Frey-Rüch, ma che tendenzialmente si usa il secondo perché su ampia scala è meno costoso.

4.4 Curve anomale

Con un occhio alla crittografia, è interessante trovare delle curve resistenti agli attacchi proposti. Un'idea è quella di fare in modo che gruppi di torsione siano speciali; abbiamo

già dimostrato che il caso delle curve supersingolari è sconveniente, resta però il caso ad esempio in cui

$$\#E(\mathbb{F}_q) = q$$

in cui l'ordine di ogni punto è necessariamente una potenza della caratteristica del campo.

Definizione 4.4. Una curva ellittica su $\mathbb{F}_q$ è detta **curva anomala** se

$$\#E(\mathbb{F}_q) = q.$$

Osservazione 4.7. Nota bene che essere anomala dipende dal campo su cui si lavora.

Purtroppo anche per le curve anomale esiste una procedura che rende ancora più facile il conto. La traccia che si segue in questo caso è questa: si solleva la curva su $\mathbb{Z}$ e studiando la riduzione di quest'ultima si ottengono informazioni sull'esponente. Diamo gli elementi per la costruzione precisa dell'algoritmo.

Teorema 4.5. Sia E una curva ellittica su $\mathbb{F}_p$ con p un primo in forma di Weierstrass

$$y^2 = x^3 + Ax + B$$

e $P, Q \in E(\mathbb{F}_p)$. Allora esistono $\tilde{A}, \tilde{B}, x_1, x_2, y_1, y_2 \in \mathbb{Z}$ tale che

$$\tilde{E}\colon\ y^2 = x^3 + \tilde{A}x + \tilde{B}$$

sia un curva ellittica e $\tilde{P} = (x_1, y_1), Q = (x_2, y_2) \in \tilde{E}(\mathbb{Q})$ con

$$\tilde{A} \equiv A \qquad \tilde{B} \equiv B \qquad \tilde{P} \equiv P \qquad \tilde{Q} \equiv Q \qquad \mod p.$$

Dimostrazione. [Was08] § 5.4 □

Osservazione 4.8. Non è vero che se $kP = Q$ allora $k\tilde{P} = \tilde{Q}$.

Indicando con $v_p\colon \mathbb{Q} \to \mathbb{Z}$ la valutazione p-adica nella notazione del teorema definiamo

$$\tilde{E}_r := \{(x, y) \in \tilde{E} \mid v_p(x) \leq -2r \wedge v_p(y) \leq -3r\} \cup \{O\}$$

Teorema 4.6. Sia $\tilde{E}$ una curva ellittica data da $\tilde{E}\colon\ y^2 = x^3 + \tilde{A}x + \tilde{B}$, p un primo e $\tilde{A}, \tilde{B} \in \mathbb{Z}$ e $r \in \mathbb{N}^+$. Allora

a) $\tilde{E}_r < \tilde{E}(\mathbb{Q})$

b) se $(x, y) \in \tilde{E}(\mathbb{Q})$

$$v_p(x) < 0 \iff v_p(y) < 0$$

e in tale caso esiste $r \leq 1$ $v_p(x) = -2r$ e $v_p(y) = -3r$

c) La mappa

$$\begin{array}{rccc} \lambda_r\colon & \tilde{E}_r/\tilde{E}_{5r} & \longrightarrow & \mathbb{Z}/p^{4r}\mathbb{Z} \\ & (x, y) & \longmapsto & p^{-r}\frac{x}{y} \\ & O & \longmapsto & 0 \end{array}$$

è un omomorfismo iniettivo.

d) Se $(x,y) \in \tilde{E}_r \setminus \tilde{E}_{r+1}$ allora $\lambda_r(x,y) \not\equiv 0 \mod p$.

Data una curva ellittica $E'\colon y^2 = x^3 + Cx + D$ su $\mathbb{Q}$ con $C, D \in \mathbb{Z}$, la **mappa di riduzione** $mod\, p$ è

$$\begin{array}{rccc} \mathrm{red}_p\colon & E'(\mathbb{Q}) & \to & E'(\mathbb{Z}/p\mathbb{Z}) \\ & [x,y,z] & \mapsto & [x,y,z] \mod p \end{array}$$

Usando che la riduzione manda rette in rette si mostra che $E'(\mathbb{Z}/p\mathbb{Z})$ è un curva di Weierstrass (non per forza liscia) e che la mappa di riduzione è un omomorfismo con $E'_{ns}(\mathbb{Z}_p)$. Nel caso che stiamo studiando noi, visto che il gcd tra il discriminate in di E e p sono coprimi, quando solleviamo come nel Teorema 4.5 e riduciamo modulo p il discriminate rimane nonnulo e la curva ridotta è liscia.
Riassumendo

Corollario 4.7. $\mathrm{red}_p\colon \tilde{E}(\mathbb{Q}) \to \tilde{E} \mod p$ è un omomorfismo e

$$\ker(\mathrm{red}_p) = \tilde{E}_1.$$

Siamo pronti per descrivere l'algoritmo:

Algorithm 4.4 ECDLP curve anomale

Input: Sia $E\colon y^2 = x^3 + Ax + B$ una curva ellittica anomala su $\mathbb{F}_p$ con p un primo di Weierstrass e $P, Q \in E(\mathbb{F}_p)$.
Output: $k\colon kP = Q$.

1: Solleviamo P, Q, E a $\tilde{P}, \tilde{Q}, \tilde{E}$
2: $\tilde{P}_1 = p\tilde{P}$
3: $\tilde{Q}_1 = p\tilde{Q}$
4: **if** $P_1 \in \tilde{E}_2$ **then**
5: Ritorna ad 1:
6: **else**
7: $\ell_1 = \lambda_1(P_1)$
8: $\ell_2 = \lambda_1(Q_1)$
9: **end if**
10: $k = \ell_2/\ell_1 \mod p$
11: **return** k

Correttezza. 4.4

(7-8): *Perché $P_1, Q_1 \in \tilde{E}_1$?*
Per il Corollario 4.7 la mappa di riduzione è un omomorfismo quindi $\mathrm{red}_p(P_1) = \mathrm{red}_p(pP) = p\,\mathrm{red}_p(P) = O$ visto che p è l'ordine del gruppo. Analogo per Q_1.

(10): *Perché possiamo invertire ℓ_2?*
Per ipotesi $P_1 \notin \tilde{E}_2$ e quindi per il Teorema 4.6.d $\lambda_1(P_1) \not\equiv 0 \mod p$.

(11): *Perché $k \mod p$ risolve ECDLP?*
In primo luogo osserviamo che, essendo p anche l'ordine del gruppo, trovare un

qualsiasi rappresentante di $k \mod p$ è tale che $kP = Q$. In secondo luogo consideriamo

$$\tilde{K} = k\tilde{P} - \tilde{Q}$$

si ha che $\mathrm{red}_p(\tilde{K}) = O$ e quindi $\tilde{K} \in \tilde{E}_1$. Perciò è ben definita

$$\lambda_1(\tilde{K}) \in \mathbb{Z}/p^4\mathbb{Z}$$

inoltre

$$\lambda_1(\tilde{K}) \equiv \lambda_1(k\tilde{P} - \tilde{Q}) \equiv k\lambda_1(\tilde{P}) - \lambda_1(\tilde{Q}) \mod p$$

moltiplicando a destra e sinistra per p

$$\begin{aligned}
0 &\equiv p\lambda_1(\tilde{K}) && (4.1)\\
&\equiv pk\lambda_1(\tilde{P}) - p\lambda_1(\tilde{Q}) && (4.2)\\
&\equiv k\lambda_1(p\tilde{P}) - \lambda_1(p\tilde{Q}) && (4.3)\\
&\equiv k\lambda_1(\tilde{P}_1) - \lambda_1(\tilde{Q}_1) && (4.4)\\
&\equiv k\ell_1 - \ell_2 \mod p && (4.5)
\end{aligned}$$

- *È necessario che la curva sia anomala affinché l'algoritmo funzioni?*
  In 2-3. Altrimenti se $\#E(\mathbb{F}_p) = M \neq p$, affinché $P_1, Q_1 \in \tilde{E}_1$ bisogna porre $\tilde{P}_1 = M\tilde{P}$ e $\tilde{Q}_1 = M\tilde{Q}$. Tuttavia se come nell'equazione 4.1 moltiplichiamo per M nessuno ci assicura che $M\lambda_1(\tilde{K}) \equiv 0 \mod p$.

□

Per completezza aggiungiamo questo paragrafo ma rimandiamo a [Was08] §5.4.
Questo algoritmo nella pratica ha un problema: se p è grande le coordinate di $\tilde{P}_1$ sono troppo grandi. Una soluzione possibile è lavorare solo con la coordinata x, che generalmente ha numeratore e denominatore sui p^2 digits.
In ogni caso, a noi interessa solo $x/y \mod p$ e l'idea è quella di lavorare modulo p^2. Questo però è problematico perché $x(p\tilde{P})$ ha p^2 al denominatore perciò $\tilde{P}_1 \equiv O \mod p^2$.
Riportiamo l'algoritmo modificato senza dettagli:

Algorithm 4.5 ECDLP curve anomale II

Input: Sia E: $y^2 = x^3 + Ax + B$ una curva ellittica anomala su $\mathbb{F}_p$ con p un primo di Weierstrass e $P, Q \in E(\mathbb{F}_p)$.
Output: k: $kP = Q$.

1: Solleviamo P, Q, E a $\tilde{P}, \tilde{Q}, \tilde{E}$
2: $\tilde{P}_2 = (p-1)\tilde{P} = (x', y') \mod p^2$
3: $\tilde{Q}_2 = (p-1)\tilde{Q} = (x'', y'') \mod p^2$
4: $m_1 = p\frac{y'-y_1}{x'-x_1}$
5: $m_2 = p\frac{y''-y_2}{x''-x_2}$
6: **if** $v_p(m_2) < 0 \vee v_p(m_1) < 0$ **then**
7: Ritorna a 1:
8: **end if**
9: $k = m_1/m_2 \mod p$
10: **return** k

Capitolo 5

Crittosistemi e algoritmi di firma

Il fatto che ECDLP sia un problema *hard*, è una cosa utile nell'ambito della crittografia perché può essere sfruttato per la costruzione di sistemi crittografici. Prima di descriverne alcuni ricordiamo il *basic setup* comune.
Un **crittosistema** è una sestupla $(\mathscr{A},\mathscr{P},\mathscr{C},\mathscr{K},\mathscr{E},\mathscr{D})$ dove

$\mathscr{A}$: è l'alfabeto

$\mathscr{P}$: è l'insieme dei messaggi (in chiaro) ammissibili o **plaintexts**

$\mathscr{C}$: è l'insieme dei messaggi cifrati o **ciphertexts**

$\mathscr{K}$: è l'insieme del chiavi o **keys**

$\mathscr{E}$: è l'insieme delle funzioni di cifratura o **encryption functions**

$$E_k : \mathscr{P} \to \mathscr{C}$$

per $k \in \mathscr{K}$

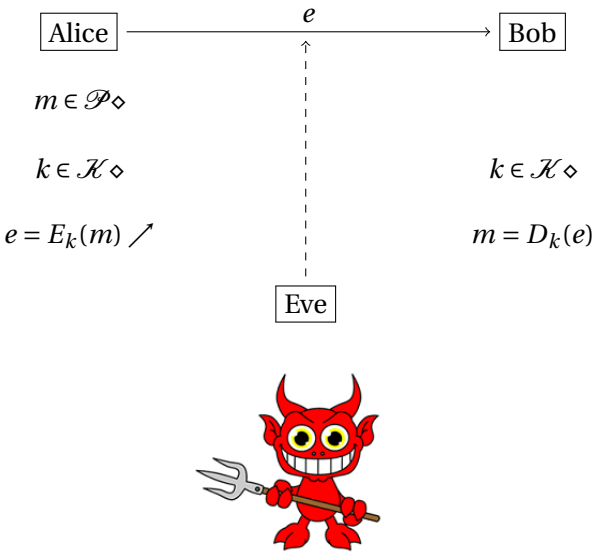
$\mathscr{E}$: è l'insieme delle funzioni di decifratura o **decryption functions**

$$D_k : \mathscr{C} \to \mathscr{P}$$

per $k \in \mathscr{K}$

dove per ogni $k \in \mathscr{K}$ $D_k \circ E_k = id_{\mathscr{C}}$.

La situazione di solito è la seguente: Alice vuole mandare un messaggio a Bob, ma vuole che lo possa leggere solo lui, ma Eve (*"the evendropper"*) tenta di inteccettarlo; allora Alice non invia il messaggio in chiaro lungo il canale di comunicazione, ma *cifrato*.



Si distinguono due tipi di crittosistema:

- a **cifratura simmetrica**. Sia Alice che Bob posseggono una chiave di cifratura e decifratura che possono essere correlate o la stessa.
- a **chiave pubblica**. Non ci sono contatti precedenti tra Alice e Bob, ma quest'ultimo pubblica una chiave che Alice deve usare per la cifratura. In questo caso deve essere difficile dedurre dalla chiave di cifratura quella di decifratura.

Osservazione 5.1. I sistemi simmetrici di solito sono più veloci quindi quello che si fa è usare un crittosistema a chiave pubblica per lo scambio delle chiavi e poi uno simmetrico per lo scambio del messaggio.

5.1 Diffie-Hellman

Il protocollo di Diffie-Hellman solitamente è utilizzato per lo scambio delle chiavi.

1. Alice e Bob si accordano su una curva ellittica E su $\mathbb{F}_q$ su cui ECDLP è difficile e $P \in E(F_q)$ tale che l'ordine di P sia un intero molto grande.
2. Alice sceglie $a \in \mathbb{Z}$ e calcola $P_a = aP$ e lo manda a Bob.
3. Bob sceglie $b \in \mathbb{Z}$ e calcola $P_b = bP$ e lo manda a Alice.
4. A partire abP estraggono un chiave (es. i primi 256 bit di $x(abP)$ o una funzione hash).

Attaccare questo crittosistema simmetrico consiste nel risolvere il

Diffie-Hellman problem: Data una curva ellittica E su $\mathbb{F}_q$ e $P, aP, bP, abP \in E(\mathbb{F}_q)$ e un altro punto in $Q \in E(\mathbb{F}_q)$ dire se $Q = abP$.

In alcuni casi Diffie-Hellman può essere attaccato usando l'accoppiamento di Weil. Consideriamo il seguente esempio per capire come:

Esempio 2. Consideriamo la curva ellittica $E\colon y^2 = x^3 + 1$ su $\mathbb{F}_q$ per $q \equiv 2 \mod 3$. Dalla proposizione 3.9 sappiamo che è supersingolare e poi nell'anello degli endomorfismi troviamo

$$\begin{array}{rccc} \alpha\colon & E & \to & E \\ & (x,y) & \mapsto & (\omega x, y) \\ & O & \mapsto & O \end{array}$$

con $\omega \in \mathbb{F}_{q^2}$ radice terza primitiva dell'unità.
α è un isomorfismo e lo possiamo usare per definire a partire dall'm-esimo accoppiamento i Weil

$$\tilde{e}_m(P,R) = e_m(P,\alpha(R))$$

per $P,R \in E[m]$.
Supponiamo di voler risolvere il DHP (con la stessa notazione della definizione) e che P abbia ordine n. Vale il seguente fatto:

Lemma 5.1. Se $3 \nmid n$ allora $\tilde{e}_n(P,P)$ radice n-esima primitiva dell'unità.

Dimostrazione. Se facciamo vedere che $\{P,\alpha(P)\}$ sono una base di $E[n]$ abbiamo la tesi. Supponiamo che esistano $c,d \in \mathbb{Z}$ tali che $cP = d\alpha(P)$; allora

$$\alpha(dP) \in E(\mathbb{F}_q)$$

se $dP = O$ allora $cP = O$ e quindi $c \equiv 0 \mod n$, altrimenti scriviamo $dP = (x,y)$ e dunque $cP = (\omega x, y) \in E(\mathbb{F}_q)$. Visto che $x \in \mathbb{F}_q$ allora $\omega x \notin \mathbb{F}_q$ se e solo se $x = 0$; gli unici punti di E con tale ascissa sono $(0,\pm 1)$ che però hanno ordine 3. □

Se $Q = tP$, noi ci chiediamo se $t \equiv ab \mod n$.

1. Se $e_n(P,Q) = 1$ siamo sicuri[1] che un tale t esiste.
2. Allora
$$\begin{cases} \tilde{e}_n(aP,bP) = \tilde{e}_n(P,P)^a b = \tilde{e}_n(P,abP) \\ \tilde{e}_n(P,Q) = \tilde{e}_n(P,tP) = \tilde{e}_n(P,P)^t \end{cases}$$
la risposta è affermativa se questi due valori coincidono.

Abbiamo un certificato senza calcolare neanche un logaritmo!

5.1.1 Diffie-Hellman Tripartito

Juox ha avuto l'intuizio di sfruttare la modifica dell'esempio 2 per costruire un sistema di scambio chiavi tra più di due utenti:

1. Alice, Bob e Chris scelgono tre chiavi private rispettivamente $a,b,c \mod n$
2. Ognuno calcola $P_i = iP$ con $i = a$ o b o c e lo pubblica
3. La chiave privata è $\tilde{e}_n(P,abcP)$.

Osservazione 5.2. $\tilde{e}_n(P,abcP) = \tilde{e}_n(aP,bP)^c = \tilde{e_n}(cP,bP)^a = \tilde{e}_n(aP,cP)^b$.

[1] Lemma 4.2.

5.2 Massey-Omura

Il protocollo Massey-Omura permette lo scambio di messaggi lungo un canale pubblico senza il bisogno di uno scambio di chiavi.

1. Alice e Bob si accordano su una curva E su $\mathbb{F}_q$ tale che ECDLP sia difficile e $\#E(\mathbb{F}_q) = N$
2. Alice rappresenta il messaggio con $M \in \#E(\mathbb{F}_q)$ e sceglie un intero m_A coprimo con N e invia $M_1 = m_A M$
3. Bob sceglie un intero m_B coprimo con N e invia $M_2 = m_B M_1$
4. Alice calcola $M_3 = {m_A}^{-1} M_2$ e lo manda a Bob
5. Bob calcola $M = {m_B}^{-1} M_3$.

Correttezza.

$$\begin{aligned} M = {m_B}^{-1} M_3 = {m_B}^{-1} {m_A}^{-1} M_2 = {m_B}^{-1} {m_A}^{-1} m_B M_1 \\ = {m_B}^{-1} {m_A}^{-1} m_B M_1 = {m_B}^{-1} {m_A}^{-1} m_B m_A M. \end{aligned}$$

Attacco. Eve conosce $(M_1, M_2, M_3, E(\mathbb{F}_q))$, scegliendo $a = {m_A}^{-1}$, $b = {m_B}^{-1}$ e $P = m_A m_b M$ si ha che questa è un'istanza di DHP.

5.2.1 Rappresentazione di un messaggio come un punto

In Massey-Omura e in altri metodi che seguono si richiede di *rappresentare il messaggio con un punto di una curva ellittica.* Esponiamo qui il **metodo di Kobliz** quando p è un primo.

Algorithm 5.1 Kobliz-rappresentazione di un messaggio con un punto

Input: E: $y^2 = x^3 + Ax + B$ su $\mathbb{F}_p$ e $m \in \mathscr{P}$ un messaggio intero $0 \leq m < 100$.
Output: $M \in E$.
Output: Sqrt calcola la radice quadrata su $\mathbb{F}_p$.

1: **for** $j = 0, \ldots, 99$ **do**
2: $x_j = 100m + j$
3: $s_j = x_j^3 + Ax_j + B$
4: **if** $\left(\frac{s_j}{p}\right) = 1$ **then**
5: y_j =Sqrt(s_j)
6: **return** $M = (x_j, y_j)$
7: **end if**
8: **end for**

Per estrarre il messaggio originario da m ci basta osservare che

$$m = \left[\frac{x_j}{100}\right]$$

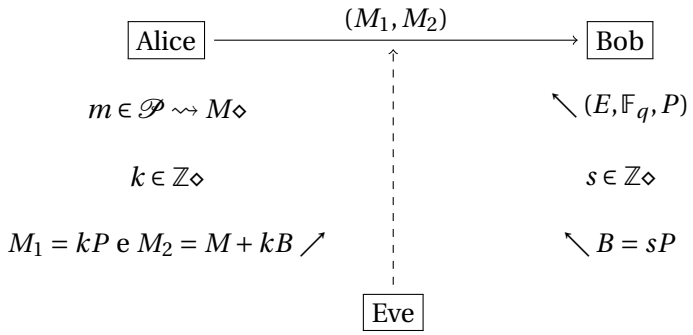
Perché il ciclo termina? Il ciclo termina approssimativamente quasi certamente: la probabilità che s_j (un elemento random) sia un residuo quadratico è circa $1/2$, perciò la probilità di non trovare M è 2^{-100}.

5.3 ElGamal, crittosistema a chiave pubblica

Il crittosistema di ElGamal è un crittosistema a chiave pubblica basato sul ECDLP.

Fase di cifratura:

1. Bob sceglie come chiave pubblica una curva ellittica E su $\mathbb{F}_q$ e un suo punto P che abbia ordine un primo grande.
2. Bob sceglie come chiave privata $s \in \mathbb{Z}$ e calcola $B = sP$.
3. Bob pubblica B
4. Alice codifica il messaggio come $M \in E(\mathbb{F}_q)$
5. Alice sceglie $k \in \mathbb{Z}$ e calcola
$$M_1 = kP$$
e
$$M_2 = M + kB$$
6. Alice invia (M_1, M_2)



Fase di decifratura:

1. Bob calcola $M = M_2 - sM_1$.

Correttezza.
$$M_2 - sM_1 = M + kB - skP = M + kB - kB = M$$

Attacco. Eve conosce $(E, \mathbb{F}_q, P, B, M_1, M_2)$, ci sono due possibilità:

- Calcola s da P, B e calcola $M = M_2 - sM_1$.
- Calcola k da P, M_1 e calcola $M = M_2 - kB$.

Osservazione 5.3. Alice ogni volta deve cambiare k, altrimenti Eve se ne accorge da M_1 e può estrapolare informazioni una volta uno dei due messaggi diventa pubblico.

5.4 Firma digitale

In alcuni contesti può essere importante avere la garanzia che unI certo messaggio sia autentico ossia che provenga effettivamente da chi dica di essere il mittente. I protocolli di firma digitale fanno proprio questo: associano al messaggio delle informazioni che garantiscono l'identità del mittente. Il modello è simile a quello dei crittosistemi ma con qualche piccola modifica. Questa volta abbia Samantha (the **signer**) che vuole provare la sua identità a Victor (the **verifier**) e Eve che tenta di spacciarsi per Sam.

5.4.1 ElGamal

L'idea del crittosistema di ElGamal può essere rivisitato come protocollo di firma.

Setup:

1. Sam sceglie un curva ellittica, un suo punto di ordine N e una funziona

   $$f\colon E(\mathbb{F}_q) \to \mathbb{Z}$$

   con immagine ampia e quasi iniettiva

2. Sam sceglie un intero $a \in \mathbb{Z}$ segreto e calcola $B = aA$

3. Sam pubblica pubblica $(E, \mathbb{F}_q, A, N, B, f)$

   Sam ⟶ Vic

   $(E, \mathbb{F}_q, A, N, B, f)$ ↗

   $a \in \mathbb{Z}$ ◇

Sia dato un messaggio codificato da $m \in \mathbb{Z}$.

Firma:

1. Sam sceglie $k \in \mathbb{Z}$ coprimo con N
2. Sam calcola $R = kA$
3. Sam calcola $s = k^{-1}(m - f(R)a) \mod N$
4. Sam invia (m, R, s)

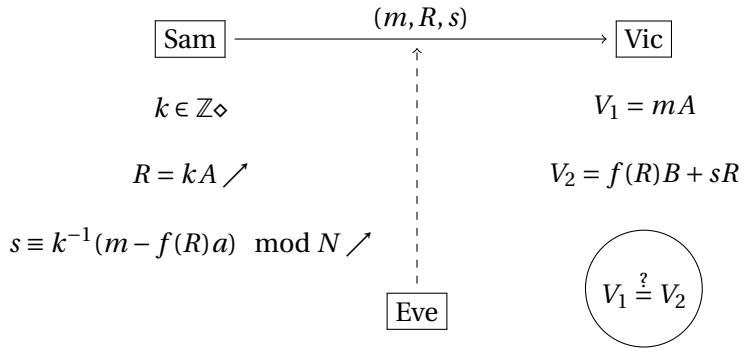
Verifica:

1. Vic calcola

   $$V_1 = mA$$

   $$V_2 = sR + f(R)B$$

2. Se $V_1 = V_2$ certifica l'identità



Correttezza.

$$mA = (m - f(R)a)A + f(R)aA = k^{-1}(m - f(R)a)R + f(R)B = sR + f(R)B$$

Attacco. Le osservazioni sono simili a quelle del crittosistema ElGamal e i possibili attacchi il logaritmo. Sam deve cambiare k ogni volta, dati infatti due triple (m,R,s) e (m',R,s') si ha

$$sk = m - f(R)a \mod N$$

$$s'k = m' - f(R)a \mod N$$

da cui

$$k(s-s') \equiv m - m' \mod N$$

e si $\gcd(s-s',N) = 1$ abbiamo trovato k e quindi a.
Oppure a Eve basta trovare un coppia R e s coerenti (tanto il messaggio è a piacere) ma se fissa R e cerca R deve risolvere un logaritmo, se invece fissa S un'equazione in x,y che si verifica essere di pari difficoltà.

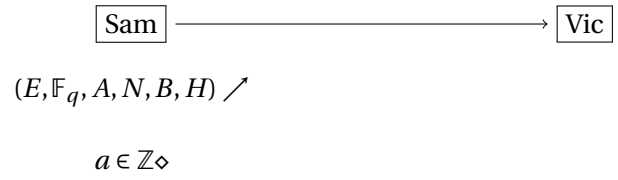
Un problema reale di questo ed altri protocolli è lo spazio. La tripla (m,R,s) occupa 3 volte i digits di m. Un metodo efficiente in questo caso è usare le funzioni hash.

Definizione 5.2. Una **funzione hash crittografica** è una funzione H che prendere stringhe di lunghezza fissata e tale che:

- data m è facile calcolare $H(m)$
- è **preimage resistan** o a immagine vuota, ossia dato y è computazionalmente impossibile trovare m tale che $H(m) = y$
- è **strongly collision free** o quasi iniettiva, ossia dati m,n è computazionalmente impossibile che $H(m) = H(n)$.

Nel protocollo di firma allora invece di usare m si può usare $H(m)$ e inviare $(H(m), R_H, s_H)$. Vain Duin ha proposto una variante che usa funzioni hash.

Setup:

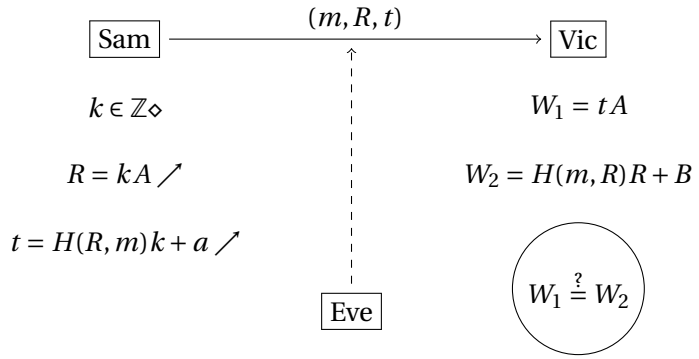


Firma:

1. Sam sceglie $k \in \mathbb{Z}$ coprimo con N
2. Sam calcola $R = kA$
3. Sam calcola $t = H(R,m)k + a \mod N$
4. Sam invia (m, R, t)

Verifica:

1. Vic calcola
$$W_1 = tA$$
$$W_2 = H(m,R)R + B$$
2. Se $W_1 = W_2$ certifica l'identità



5.4.2 ECDSA

ECDSA è una versione che utilizza le curve ellittiche del Digital Signature Algorithm che nella versione originale utlizza i gruppi moltipicativi. Samantha vuol firmare un documento $m \in \mathbb{Z}$.

Setup:

1. Sam sceglie un curva ellittica E su $\mathbb{F}_q$ e un suo punto G di ordine r un primo grande, dove
$$\#E(\mathbb{F}_q) = f \cdot r$$
con $f \in \mathbb{Z}$ piccolo (es. 1,2,4)
2. Sam sceglie un intero $a \in \mathbb{Z}$ segreto e calcola $Q = aG$

3. Sam pubblica pubblica $(E, \mathbb{F}_q, G, r, Q)$

Firma:

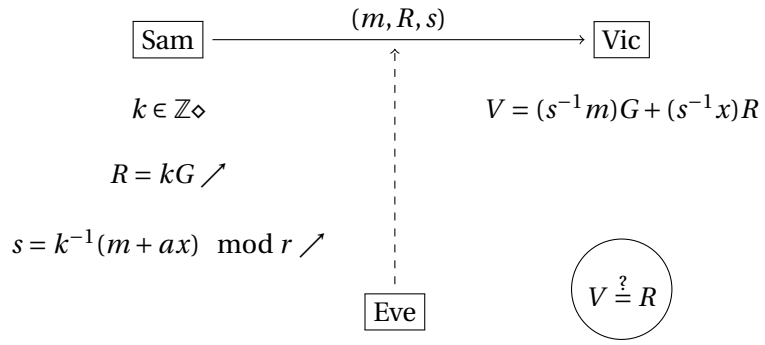
1. Sam sceglie $k \in \mathbb{Z}$ $1 \leq k < r$
2. Sam calcola $R = kG = (x, y)$
3. Sam calcola $s = k^{-1}(m + ax) \mod r$
4. Sam invia (m, R, s)

Verifica:

1. Vic calcola
$$u_1 = s^{-1}m \mod r$$
$$u_2 = s^{-1}x \mod r$$
$$V = u_1 G + u_2 Q$$
2. Se $V = R$ certifica l'identità

Correttezza.

$$V = s^{-1}mG + s^{-1}xQ = s^{-1}mG + s^{-1}xaG = s^{-1}(m + xa)G = kG = R$$



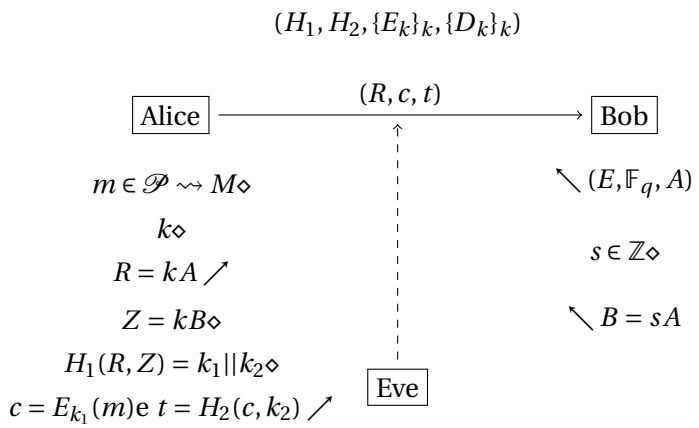
5.5 ECIES, crittosistema a chiave pubblica

The Elliptic Curve Integrated Encryption Scheme (ECIES) è un crittosistema a chiave pubblica.

Siano date due funzioni hash H_1, H_2 e una famiglia $\{E_k\}_k$ di funzioni di cifratura simmetrica dipendenti dalla chiave k e le rispettive funzioni di decifratura $\{D_k\}_k$.

Fase di cifratura:

1. Bob sceglie come chiave pubblica una curva ellittica E su $\mathbb{F}_q$ e un suo punto A che abbia ordine N
2. Bob sceglie come chiave privata $s \in \mathbb{Z}$ e calcola $B = sA$.
3. Bob pubblica B
4. Alice scarica la chiave pubblica
5. Alice sceglie $k \in \{1, \ldots, N-1\}$
6. Alice calcola
$$R = kA$$
e
$$Z = kB$$
7. Alice calcola $H_1(R, Z) = k_1 || k_2$ ottenendo due stringhe di lunghezza fissata concatenate
8. Alice dato m il messaggio calcola $E_{k_1}(m) = c$ e $t = H_2(c, k_2)$
9. Alice invia (R, c, t)



Fase di decifratura:

1. Bob calcola $Z = sR$
2. Bob calcola $H_1(R, Z)$
3. Bob calcola $H_1(c, k_2)$ e controlla $t = H_1(c, k_2)$
4. Bob calcola $m = D_{k_2}(c)$

Osservazione 5.4. Dal punto di vista computazionale questo algoritmo è vantaggioso perché non chiede di codificare il messaggio con un punto della curva. Inoltre il controlla al punto 3 della fase di decifratura garantisce che il messaggio sia effettivamente mandato da Alice (vedi attacco).

Attacco. Eve conosce potrebbe forzare Bob a decriptare un bel po' di messaggi e ottenere così informazioni su s. Infatti Eve potrebbe produrre del testo cifrato (c', k_2') e $t' = H_1(c', k_2')$, ma Bob grazie al passo 3 non è costetto ad usare D perché non conoscendo Z non può ottenere il giusto t'.

5.6 Boneh-Franklin, crittosistema basato sull'accoppiamento

Il crittosistema Boneh-Franklin è diverso dai crittosistemi descritti fino ad adesso. In questo caso Alice e Bob sono solo visti come utenti di un sistema più grande e il loro scambio di messaggi è legato alla loro identità all'interno di quest'ultimo. In particolare le nessuno dei due conosce la chiave che invece è gestita ad un livello superiore. Per fare un esempio Bob e Alice sono due utenti di un social network e Alice per mandare un messaggio si autentica, scrive il messaggio che viene cifrato e spedito, Bob lo riceve si autentica e può leggere il messaggio decifrato.
Formalizziamo questo concetto:

Definizione 5.3. Un **identity-based encryption scheme** (IBE) $\mathscr{E}$ è dato da quattro algoritmi (randomizzati):

- **Setup:** prende un parametro di sicurezza k e restituisce un sistema di parametri (*params*) e la **master-key**. *params* include $\mathscr{P}$ e $\mathscr{C}$ (come in 5). Il sistema di parametri può essere pubblico mentre la master-key è conosciuta solo dal **Public Key Generator** (PKG)[2]

- **Extract:** prende *params* in input e una stringa ID$\in \{0,1\}^*$ (*identità*) e restituisce d la chiave privata. Qui ID viene utilizzata come una chiave pubblica d la corrispondente chiave privata di decriptazione. Moralmente Extract estrae una chiave privata da una chiave pubblica.

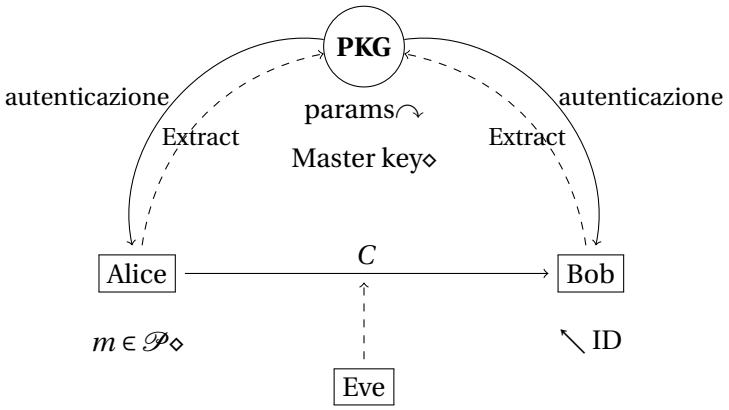
- **Encrypt:** prende *params*, ID e $M \in \mathscr{P}$ e restituisce $C \in \mathscr{C}$.

- **Decrypt:** prende *params*, $C \in \mathscr{C}$ e d e restituisce $M \in \mathscr{P}$.

Questi algoritmi devo soddisfare il *vincolo di consistenza standard*, ossia quando d è la chiave privata generata da `Extract` per ID, chiave pubblica, allora

$$\forall M \in \mathscr{P}\colon \texttt{Decrypt}(\texttt{params}, C, d) = M \text{ per } C = \texttt{Encrypt}(\texttt{params}, \text{ID}, M).$$

[2] In alcuni testi inveco di PKG si trova la **trusted authority** (TA).



Descriviamo un protocollo IBE che utilizza l'accoppiamento di Weil.

Setup:

1. PKG sceglie un primo $p = 6\ell - 1$ con ℓ un primo
2. PKG sceglie un punto P di ordine ℓ sulla curva ellittica

$$E(\mathbb{F}_p)\colon\ y^2 = x^3 + 1$$

3. PKG sceglie due funzioni hash:
   - H_1 prende stringhe di lunghezza fissata (ID) e restituisce un punto di ordine ℓ
   - H_2 prende un elemento $a \in \mathbb{F}_{p^2}$ di ordine ℓ e restituisce un stringa di lunghezza n (lunghezza messaggi ammissibili)
4. PKG sceglie $s \in \mathbb{F}_\ell^*$ (segreto) e calcola $P_{pub} = sP$
5. PKG pubblica pubblica $(E, p, H_1, H_2, n, P, P_{pub})$

Extract:

1. ID richiede la chiave privata a PKG
2. PKG calcola $Q_{ID} = H_1(\text{ID}) \in E$
3. PKG calcola $D_{ID} = sQ_{ID}$
4. PKG verifica l'identità di ID e in caso positivo invia D_{ID}

Encrypt:

1. Alice sceglie un messaggio M da mandare a Bob
2. Alice cerca ID di Bob e calcola Q_{ID}
3. Alice sceglie $r \in \mathbb{F}_\ell^*$
4. Alice calcola $g_{ID} = \tilde{e}_\ell(Q_{ID}, P_{pub})$

5. il testo cifrato è
$$c = (rP, M \oplus H_2(g_{ID}^r))$$
dove $\oplus$ è lo XOR binario.

Decrypt:

1. Bob riceve un messaggio (u, v)
2. Bob calcola $h_{ID} = \tilde{e}_\ell(D_{ID}, u)$
3. Bob calcola
$$M = v \oplus H_2(h_{ID}).$$

Correttezza. Basta osservare che

$$\begin{aligned} h_{ID} =& \tilde{e}_\ell(D_{ID}, u) = \tilde{e}_\ell(D_{ID}, rP) = \tilde{e}_\ell(D_{ID}, P)^r = \\ =& \tilde{e}_\ell(sQ_{ID}, P)^r = \tilde{e}_\ell(Q_{ID}, sP)^r = \tilde{e}_\ell(sQ_{ID}, P_p ub)^r = \\ =& g_{ID}^r. \end{aligned}$$

e che $2\oplus = id$.

Capitolo 6

Geometria algebrica reale computazionale

La geometria algebrica è quella branca della matematica che unisce lo studio di luoghi geometrici a quello di strutture algebriche. Classicamente tra le ipotesi che si fanno c'è anche quella che il campo base sia algebricamente chiuso, ma questa è in effetti una richiesta abbastanza restrittiva dal momento in cui, ad esempio, in molte applicazioni si richiede i punti che si vanno a considerare sono punti a coordinate intere o reali.
In questo capitolo siamo interessati ad aspetti computazionali, in particolare alla ricerca di soluzioni di sistemi di equazioni e disequazioni polinomiali a coefficienti reali, che si traduce vedremo come studio degli insiemi semialgebrici. Nell'ambito della geometria algebrica reale gli insiemi semialgebrici vedremo sono la base naturale su cui lavorare; in geometria algebrica classica questo ruolo è assunto dagli insiemi algebrici, ma lavorando su campi non algebricamente chiusi come i campi reali questi insiemi perdono una proprietà fondamentale: la proiezione di un sottoinsieme algebrico di $\mathbb{R}^n$ su un sottospazio non è algebrica. Vedremo in questo capitolo che invece i semialgebrici godono di questa e molte altre proprietà e poi studieremo un primo modo di decomporre questi spazi che ci permetterà anche di definire la dimensione di un semialgebrico.
Qui svilupperemo la teoria usando come campo $\mathbb{R}$ anche se, con pochissime accortezze, quello che diremo vale per qualsiasi campo reale chiuso.

6.1 Il metodo di Sturm per contare le radici

Dato un polinomio $p(X) \in \mathbb{R}[X]$ di grado n sappiamo che ha n radici complesse contate con molteplicità, ma a priori non sappiamo niente sul numero di radici reali. In questa sezione proproniamo un metodo per contare le radici in un determinato intervallo.

Polinomi con radici semplici

In questo paragrafo supporremo sempre che $p(X) \in \mathbb{R}[X]$ sia un polinomio con radici semplici[1].

[1] Lo possiamo verificare tramite il criterio della derivata ad esempio.

Definizione 6.1. Siano $p, g \in \mathbb{R}[X]$. Definiamo **sequenza di Sturm** di p e g la successione di polinomi $(p_0, \ldots, p_k)$ tali che $p_0 = p$, $p_1 = p'g$ e p_i è l'opposto del resto della divisione euclidea tra p_{i-2} e p_{i-1} ($p_i = p_{i-1}g_i - p_{i-2}$ con $g_i \in \mathbb{R}[X]$ e $deg(p_i) < deg(p_{i-1})$) per $i = 1 \ldots k$.
Presa inoltre $a \in \mathbb{R}$ non radice di p indichiamo con $v(p, g; a)$ il numero di cambi di segno della sequenza di Sturm di p e g valutata in a. Se è chiaro dal contesto useremo anche $v_p(a)$ per $v(p, 1; a)$ (da non confondere con la valutazione p-adica).

Teorema 6.2 (**Teorema di Sturm**)**.** Sia $p \in \mathbb{R}[X]$, $a, b \in \mathbb{R}$ con $a < b$ che non siano radici di p, allora il numero di radici reali di p in $]a, b[$ è uguale a

$$v_p(a) - v_p(b).$$

Dimostrazione. Ci possiamo ridurre al caso in cui in $]a, b[$ ci sia un sola radice e sfruttare l'additività. Usando quanto detto nell'osservazione, basta osservare che cosa accade alla funzione $x \mapsto v(p, 1; x)$ quando x passa da una radice c di uno dei polinomi della sequenza:

- Se c è una radice di p i segni di p_0 e p_1 cambiano rispettando le seguenti regole[2]

x		c	
p_0	$-$	0	$+$
p_1	$+$	$+$	$+$

oppure

x		c	
p_0	$+$	0	$-$
p_1	$-$	$-$	$-$

  e in entrambi i casi $v(p, 1; x)$ decresce di una unità.

- Se c è una radice di p_i per $i = 1 \ldots k$ allora $p_{i-1}(c) = p_{i+1}(c) \neq 0$ il contributo della terna p_{i-1}, p_i, p_{i+1} a $v(p, 1; x)$ non cambia e rimane uguale a 1.

□

Osservazione 6.1. Dalla dimostrazione viene fuori che se $g = 1$:

1. $p_0 = p$ e la sequenza di Sturm è finita e $p_k = \gcd(p, p') \in \mathbb{R}^*$.
2. se c radice di p_0 allora il prodotto $p_0 p_1$ è negativo su qualche intervallo $(c - \varepsilon, c)$ e positivo su $(c, c + \varepsilon)$.
3. Se c è radice di p_i per $0 < i < k$ allora $p_{i-1}(c)p_{i+1}(c) < 0$: infatti se esiste un radice di p_i tale che $p_{i+1}(c) = 0$ allora anche $p_{i-1}(c) = 0$ e induttivamente sarebbe radice di p e p', il che è assurdo perché il polinomio di partenza non ha radici multiple; infine da $p_{i+1}(c) = p_i(c) - p_{i-1}(c) = p_{i-1}(c)$.

Polinomi con radici multiple

Supponiamo ora che $p(X) \in \mathbb{R}[X]$ sia un polinomio con radici multiple e consideriamo $(p_0, p_1, \ldots, p_k)$ la sequenza di Sturm associata a p p' (ossia $g = 1$). Visto che $p_k = \gcd(p, p')$ in questo caso non è una costante. Consideriamo allora la sequenza $(q_0, q_1, \ldots, q_k)$ data da

$$q_i = \frac{p_i}{p_k}$$

[2] p_1 esprime è la derivata e quindi descrive la pendenza.

Allora si ha che le proprietà dell'osservazione 6.1 valgono per $(q_0, q_1, \ldots, q_{k-1}, 1)$ e visto che q_0 è la parte square free di p si ha che il numero di cambi di segno della sequenza originale è lo stesso di quella nuova, ossia

$$v_p(a) = v_{q_0}(a)$$

per ogni $a \in \mathbb{R}$.
Si ha quindi che il teorema di sturm vale anche in questo caso:

Teorema 6.3 (**Teorema di Sturm**)**.** Sia $p \in \mathbb{R}[X]$, $a, b \in \mathbb{R}$ con $a < b$ che non siano radici di p, allora il numero di radici distinte di p in $]a, b[$ è uguale a

$$v_p(a) - v_p(b).$$

Un bound sull'intervallo

Il teorema di Sturm ci dà il numero di radici in un intervallo. Quanto deve essere grande questo intervallo affinché possiamo avere la certezza di aver contato tutte le radici?

Proposizione 6.4. Sia $p(X) = a_0 X^d + \cdots + a_d$ con $a_0 \neq 0$. Se $c \in \mathbb{C}$ è radice di p, allora

$$|c| \leq M := \max_{i=1,\ldots,d} \left(d \left| \frac{a_i}{a_0} \right| \right)^{1/i}.$$

Dimostrazione. Sia $z \in \mathbb{C}$ di norma maggiore di M. Allora per ogni $i = 1, \ldots, d$

$$|z| > \left(d \left| \frac{a_i}{a_0} \right| \right)^{1/i}$$

ossia

$$|a_0||z|^i / d > |a_i|$$

e quindi

$$|a_1 z^{d-1} + \cdots + a_d| \leq |a_1||z^{d-1}| + \cdots + |a_d| < |a_0 z^d|.$$

□

Corollario 6.5. Sia $p(X) \in \mathbb{R}[X]$. Allora

$$\#Roots(p) = v_p(M) - v_p(-M).$$

Condizioni sul segno

All'inizio del capitolo abbiamo detto che siamo interessati a sistemi di equizioni e disequazioni polinomiali. È interessante quindi prendere in considerazione il problema di contare il numero di radici di un polinomio univariato con delle condizioni espresse da disequazioni.

Iniziamo con la situazione più semplice: contare il numero di radici di $p(X) \in \mathbb{R}[X]$ tali che $q(X) > 0$ per $q(X) \in \mathbb{R}[X]$.

Teorema 6.6 (Teorema di Sylvester)**.** Siano $a, b \in \mathbb{R}$ con $a < b$ che non siano radici di p, allora

$$v(p, q; a) - v(p, q; b)$$

è uguale al numero di radici distinte di p in (a, b) tali che $q(c) > 0$ meno il numero di quelle tali che $q(c) < 0$.

Dimostrazione. Dapprima supponiamo $p_k = \gcd(p, p'g) \in \mathbb{R}^*$. La dimostrazione è analoga al teorema di Sturm. In particolare basta estendere l'osservazione 6.1:

1. $p_0 = p$ e la sequenza di Sturm è finita e $p_k = \gcd(p, p'g) \in \mathbb{R}^*$.

2. se c radice di p_0 allora il prodotto $p_0 p_1 = pp'g$ è negativo su qualche intervallo $(c - \varepsilon, c)$ e positivo su $(c, c + \varepsilon)$. Infatti visto che $g(c) \neq 0$ ci sono due possibilità:
   se $g(c) > 0$

x		c	
p_0	$-$	0	$+$
p_1	$+$	$+$	$+$

   oppure

x		c	
p_0	$+$	0	$-$
p_1	$-$	$-$	$-$

   il numero di cambi di segno decresce di 1; invece se $g(c) < 0$

x		c	
p_0	$-$	0	$+$
p_1	$-$	$-$	$-$

   oppure

x		c	
p_0	$+$	0	$-$
p_1	$+$	$+$	$+$

   il numero di cambi di segno cresce di 1.

3. Se c è radice di p_i per $0 < i < k$ allora $p_{i-1}(c)p_{i+1}(c) < 0$: infatti se esiste un radice di p_i tale che $p_{i+1}(c) = 0$ allora anche $p_{i-1}(c) = 0$ e induttivamente sarebbe radice di p e p', il che è assurdo perché il polinomio di partenza non ha radici multiple; infine da $p_{i+1}(c) = p_i(c) - p_{i-1}(c) = p_{i-1}(c)$.

Se $p_k = \gcd(p, p'g) \neq \mathbb{R}^*$ ci possiamo ridurre ancora un volta al caso precedente usando la serie square free. □

Come possiamo calcolare allora il numero di radici distinte di $p(X)$ tali che $q(X) > 0$? Osserviamo che il numero di radici distinte di p che non sono radici reali di q è

$$v(p, q^2; a) - v(p, q^2; b)$$

allora ci basta calcolare

$$\frac{1}{2}(v(p, q; a) + v(p, q^2; a) - v(p, q; b) - v(p, q^2; b))$$

Occupiamoci del caso generale. Contare il numero di radici di $p(X) \in \mathbb{R}[X]$ tali che $q_1(X), \ldots, q_t(X) > 0$ per $q_i(X) \in \mathbb{R}[X]$.
Supponiamo per ogni i che $\gcd(p, q_i) = 1$ e definiamo

$$\varepsilon = (\varepsilon_1, \ldots, \varepsilon_t) \in \{0, 1\}^t$$

e $Q^\varepsilon(X) = q_1^{\varepsilon_1}(X)\cdots q_t(X)^{\varepsilon_t}$.
Per ogni ε possiamo anche calcolare grazie al teorema 6.6

$$s_\varepsilon = v(p, Q^\varepsilon; -\infty) - v(p, Q^\varepsilon; \infty)$$

e poi per

$$\varphi = (\varphi_1, \ldots, \varphi_t) \in \{0,1\}^t$$

definiamo

$$c_\varphi := \#\{x \in \mathbb{R} \mid p(x) = 0 \wedge \operatorname{sign}(q_i(x)) = (-1)^{\varphi_i}\ \forall i\}$$

Possiamo trovare una relazione tra s_ε e c_φ.

Lemma 6.7. Detti s, c i vettori di dimensione 2^t che contengono tutte i possibili s_ε e c_φ, esiste una matrice A_t $2^t \times 2^t$ invertibile tale che $s = A_t c$.

Dimostrazione. Per induzione su t. Se $t = 0$ è banale $c_\varnothing = s_\varnothing$. Se $t = 1$ il teorema di Sturm e 6.6 ci danno :

- $s_0 = v(p, q^0; -\infty) - v(p, q^0; \infty) = v_p(-\infty) - v_p(\infty)$ il numero totale di radici reali distinte di p
- $s_1 = v(p, q; -\infty) - v(p, q; \infty)$ la differenza tra il numero di radici per cui $g(c) > 0$ meno quelle per cui $g(c) < 0$.
- $c_0 = \#\{x \in \mathbb{C} \mid p(x) = 0 \wedge q(x) > 0\}$
- $c_1 = \#\{x \in \mathbb{C} \mid p(x) = 0 \wedge q(x) < 0\}$

Allora $s_0 = c_0 + c_1$ e $s_1 = c_0 - c_1$:d'd'

$$\begin{bmatrix} s_0 \\ s_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}$$

Se $t > 1$ per induzione otteniamo

$$A_{t+1} = \begin{bmatrix} A_t & A_t \\ A_t & -A_t \end{bmatrix}$$

Osserviamo infine

$${A_{t+1}}^{-1} = \frac{1}{2} \begin{bmatrix} {A_t}^{-1} & {A_t}^{-1} \\ {A_t}^{-1} & -{A_t}^{-1} \end{bmatrix}$$

□

Per calcolare un c_ε, a noi interessa c_0 ma potremmo essere interessati ad altre configurazioni quindi è bene rimanere sul generico, allora possiamo calcolare con il teorema 6.6 s e calcolare $c = {A_t}^{-1} s$ ed estrarre le coordinate di c_ε.

Per concludere supponiamo che p possa avere radici multiple o in comune con qualche q_i. Allora basta sostituire Q^2/Q^ε a Q^ε.

6.2 Il principio di Tarski-Seidenberg

Presentiamo il Principio di Tarski-Seidenberg che è un principio, si può dire logico, valido anche nel contesto dei campi ordinati e che vedremo sarà di fondamentale importanza nello studio degli insiemi semialgebrici.

Teorema 6.8 (**Principio di Tarski-Seidenberg [prima forma]**)**.** Sia dato un sistema di equazioni polinomiali nelle variabili $T = (T_1, \dots, T_p)$ e X a coefficienti in $\mathbb{R}$, che indichiamo con $\mathscr{S}(T, X)$:

$$\begin{cases} S_1(T,X) \circledast_1 0 \\ S_2(T,X) \circledast_2 0 \\ \dots \\ S_l(T,X) \circledast_l 0 \end{cases}$$

con $\circledast_i \in \{=, \neq, >, \geq\}$ e $S_i(T, X) \in \mathbb{R}[T, X]$ per $i = 1, \dots, l$. Allora esiste un metodo algoritmico per produrre una lista $\mathscr{C}_1(T), \dots, \mathscr{C}_k(T)$ di sistemi di equazioni e disequazioni in $\mathbb{R}[T]$, con k finito, tali che per ogni $t \in \mathbb{R}^p$ i seguenti fatti siano equivalenti:

i. $\mathscr{S}(t, X)$ ha soluzione

ii. almeno uno dei $\mathscr{C}_j(t)$ è soddisfatto.

In altri termini il teorema afferma che l'asserto "$\exists X\, \mathscr{S}(T, X)$" equivale a "$\mathscr{C}_1(T) \vee \dots \vee \mathscr{C}_k(T)$", dunque ci permette di eliminare la variabile X e il quantificatore d'esistenza.
Non entriamo nei dettagli della dimostrazione ma ne riassumiamo i passi principali, che sono quanto ci serve per comprendere quali sono le idee che stanno dietro a questo importante teorema. Iniziamo facendo una osservazione preliminare che ci permette di restringerci ad un caso più semplice da gestire e introduciamo la mappa segno.

Osservazione 6.2. Dato un sistema $\mathscr{S}(T, X)$ è possibile ottenere degli ulteriori sistemi, che chiameremo **sistemi con gradi fissati**, $(\mathscr{S}(T, X), \mathscr{D}(T))$, aggiungendo in $\mathscr{D}(T)$ le condizioni[3] $lc_X S_i \neq 0$. Questi sistemi sono convenienti e più facili da studiare perché permettono di tenere sotto controllo il grado dei polinomi. Visto che ogni sistema può essere scritto come unione disgiunta di sistemi con gradi fissati (induttivamente sul grado si separa il caso in cui il leading coefficient sia uguale o diverso da zero) supporremo sempre di essere in tale ipotesi.

Dimostrazione. (TSI) Grazie all'osservazione senza perdita di generalità intenderemo sempre, parlando di sistema, un sistema con gradi fissati.

Passo 1 Sistemi con una solo equazione.

Sia $(P, Q_1, \dots, Q_l)$ una l-upla di polinomi in $\mathbb{R}[T, X]$, algoritmicamente si trovano

$$R_1(T), \dots, R_k(T) \in \mathbb{R}[T]$$

[3] Con $lc_X f$ intendiamo il coefficiente direttivo, o leading coefficient, del polinomio f considerato come univariato in X.

e una funzione $c\colon \{-1,0,1\}^k \to \mathbb{N}$ tali che per ogni $t \in R^p$ e per ogni $\varepsilon = (\varepsilon_1,\dots,\varepsilon_k) \in \{-1,0,1\}^k$, se vale

$$"\mathscr{D}(t) \wedge \text{sign}(R_1(t)) = \varepsilon_1 \wedge \cdots \wedge sign(R_k(t)) = \varepsilon_k"$$

allora il sistema

$$P = 0 \wedge Q_1 > 0 \wedge \cdots \wedge Q_l > 0$$

ha $c(\varepsilon_1,\dots,\varepsilon_k)$ soluzioni.
Questo fatto si dimostra usando la sequenza di Sturm e quando sviluppato a partire dal teorema 6.6. L'idea è quella di costruire le varie sequenze rispetto alla variabile X e creare un albero di decisione in base alle varie condizioni che si trovano sui coefficienti, i quali ricordiamo sono funzioni in T. In ogni foglia abbiamo il sistema determinato dalle condizioni e i segni dei leading coefficient (che sono polinomi in $\mathbb{R}[T]$) della successione di Sturm così ottenuta determinano la differenza del numero di cambi di segno. C'è una cosa importante da notare: questi leading coefficient sono funzioni razionali del tipo $A(T)/B(T)$, dove B non può annullarsi nel ramo in cui compare; ma il segno di $A(t)/B(t)$ è lo stesso di $A(t)B(t)$ perciò ci basta prendere come $R_1,\dots,R_m$ tutti i polinomi di questo tipo presenti in tutti i rami di tutti gli alberi di calcolo delle successioni di Sturm associate al sistema. ci basta per concludere: detto $\mathscr{C}_\varepsilon(T)$ il sistema polinomiale

$$\begin{cases} \text{sign}(R_1(t)) = \varepsilon_1 \\ \dots \\ \text{sign}(R_k(t)) = \varepsilon_k \\ c(\varepsilon) > 0 \end{cases}$$

allora se esiste $\tilde{\varepsilon}$ tale che in $(t,\tilde{\varepsilon})$ $\mathscr{C}_\varepsilon(T)$ ha soluzione la ha anche $\mathscr{S}(T,X)$; viceversa se esiste una soluzione per $\mathscr{S}(t,X)$ supponendo che nessuno dei $\mathscr{C}_{\tilde{\varepsilon}}(t)$ abbia soluzione avremmo sempre che $c(\varepsilon) = 0$, quindi che $\mathscr{S}(t,X)$ avrebbe zero soluzioni, assurdo.

Passo 2 Caso generale.
Se ci sono più equazioni di grado positivo rispetto ad X $P_1 = \cdots = P_b = 0$ si possono sostituire con $P_1^2 + \cdots + P_b^2 = 0$ e ricondurci al passo 1. Altrimenti siamo nel caso in cui il sistema è

$$Q_1 > 0 \wedge \cdots \wedge Q_l > 0$$

con almeno un polinomio di grado positivo in X. In tal caso il sistema ha soluzione in un intervallo aperto e illimitato se e solo se i coefficienti direttori dei polinomi hanno tutti lo stesso segno. Altrimenti il sistema ha soluzione in un in intervallo le cui soluzioni i cui estremi sono radici di $Q = \prod Q_j$ se e solo se il sistema

$$\frac{\partial Q}{\partial X} = 0 \wedge Q_1 > 0 \wedge \cdots \wedge Q_l > 0$$

ha soluzioni reali (il che ci riconduce ancora una volta al passo 1).

Per concludere riflettiamo ancora un attimo su quanto detto sin ora. Il fulcro della dimostrazione è che sui campi reali siamo in grado di dire in un intervallo quante radici reali ci sono e quindi caratterizzare il segno ha un polinomio (contando i cambiamenti di segno della valutazione di altri polinomi nei soli estremi). Questo procedimento è fattibile algoritmicamente in un numero finito di passi, cosicché andando a ritroso si trovano un numero finito di polinomi univariati, quindi un sistema, il cui studio delle soluzioni è realizzabile in molteplici modalità, anch'esse algoritmiche. □

6.3 Il metodo di Hermite per contare le radici

Presentiamo, senza tutti i dettagli, un altro metodo per contare le radici di un polinomio univariato. La motivazione che ci spinge a studiarlo è che ci permetterà di introdurre alcuni concetti che ci serviranno più avanti e che è una buona alternativa al metodo di Sturm.

Consideriamo $p \in \mathbb{R}[X]$ polinomio monico

$$p(X) = X^d + a_1 X^{d-1} + \cdots + a_d$$

Sia $i \in \mathbb{N}$ la **i-esima somma di Newton** si p è

$$N_i := \sum_{j=1}^{d} \alpha_j^i$$

dove $\alpha_1, \ldots, \alpha_d \in \mathbb{C}$ sono le radici di p contate con molteplicità.

Osservazione 6.3. Se il polinomio non è monico le somme di Newton possono essere viste come polinomi in $\mathbb{Z}[a_1/a_0, \ldots, a_d/a_0]$.

Le somme di Newton si possono ricavare dai coefficienti di p, anche senza conoscerne le radici. Con un semplice conto si ottiene:

Lemma 6.9. Vale che:

$$\frac{p'}{p} = \sum_{i \geq 0} \frac{N_i}{X^{i+1}}$$

Da cui si ha che

$$p'(x) = \sum_{k=1}^{d} \sum_{i=0}^{d-k} a_{d-k-i} N_i x^{k-1} \tag{6.1}$$

e eguagliando alla scrittura che si ottiene dalle formule per la derivazione classiche di un polinomio:

Corollario 6.10.

$$\begin{aligned}
N_0 &= d \\
N_1 &= -a_1 \\
N_2 &= -(N_1 a_1 - 2a_2) \\
N_i &= -(N_{i-1} a_1 + N_{i-2} a_2 + \cdots + i a_i) && i \leq d \\
N_i &= -(N_{i-1} a_1 + N_{i-2} a_2 + \cdots + N_{i-d} a_d) && i > d
\end{aligned}$$

Consideriamo la seguente matrice

$$H(p) = \begin{bmatrix} N_0 & N_1 & \dots & N_{d-1} \\ N_1 & N_2 & \dots & N_d \\ \vdots & \vdots & \vdots & \vdots \\ N_{d-1} & N_d & \dots & N_{2d-2} \end{bmatrix}$$

essendo simmetrica possiamo interpretarla come una forma quadratico e quindi è ben definita la segnatura[4].

Teorema 6.11. La segnatura $\sigma(H(p))$ è uguale al numero di radici reali distinte di p e $\mathrm{rk}(H(p))$ è uguale al numero totale di radici distinte di p.

Osservazione 6.4.

- Se $V = V(\alpha_1, \dots, \alpha_d)$ è la matrice di Vandermonde associata alle radici di p si ha che $H(p) = VV^T$.
- Il teorema di Sylvester ci dice che $H(p)$ è simile ad una matrice a blocchi

$$S = \begin{bmatrix} I_{i_+} & & \\ & -I_{i_-} & \\ & & 0_{i_0} \end{bmatrix}$$

tramite una matrice ortogonale M. In particolare la forma quadratica bilineare associata $x \mapsto x^T H(p)x$ diventa

$$x^T H(p)x = x^T M^T SMx = (Mx)^T S(Mx) \tag{6.2}$$

$$= \sum_{j=1}^{i_+} (M_j x)^2 - \sum_{j=i_++1}^{i_++i_-} (M_j x)^2. \tag{6.3}$$

Dimostrazione. Dall'osservazione discende che $\mathrm{rk}(H(p)) = \mathrm{rk}(VV^T) = \mathrm{rk}(V)$ che è proprio il numero di radici distinte di $p(X)$.
Osserviamo che

$$x^T H(p)x = x^T VV^T x =$$

$$= \begin{bmatrix} \sum_{j=1}^d \alpha_1^{j-1} x_j & \sum_{j=1}^d \alpha_2^{j-1} x_j & \dots & \sum_{j=1}^d \alpha_i^{j-1} x_j & \dots & \sum_{j=1}^d \alpha_d^{j-1} x_j \end{bmatrix} \cdot \begin{bmatrix} \sum_{j=1}^d \alpha_1^{j-1} x_j \\ \sum_{j=1}^d \alpha_2^{j-1} x_j \\ \vdots \\ \sum_{j=1}^d \alpha_i^{j-1} x_j \\ \vdots \\ \sum_{j=1}^d \alpha_d^{j-1} x_j \end{bmatrix}$$

$$= \sum_{i=1}^d (\sum_{j=1}^d \alpha_i^{j-1} x_j)^2$$

[4]La **segnatura** qui è intesa non come terna fatta dagli indici di positività i_+, negatività i_- e nullità i_0, ma come differenza dei primi due.

Consideriamo adesso una radice $\beta \in \mathbb{C}\setminus\mathbb{R}$, visto che il polinomio ha coefficienti reali avremo che anche la sua coniugata $\bar{\beta}$ compare nella lista con la stessa molteplicità. Osserviamo che

$$\begin{aligned}
&(\sum_{j=1}^{d} \beta^{j-1} x_j)^2 + (\sum_{j=1}^{d} \bar{\beta}^{j-1} x_j)^2 = \\
=&(\sum_{j=1}^{d} Re(\beta^{j-1}) x_j + i \sum_{j=1}^{d} Im(\beta^{j-1}) x_j)^2 + (\sum_{j=1}^{d} Re(\beta^{j-1}) x_j - i \sum_{j=1}^{d} Im(\beta^{j-1}) x_j)^2 = \\
=&(\sum_{j=1}^{d} Re(\beta^{j-1}) x_j)^2 - (\sum_{j=1}^{d} Im(\beta^{j-1}) x_j)^2
\end{aligned}$$

dunque nella scrittura come in formula 6.3 il contributo alla segnatura si elide.
Rimane da osservare un cosa, nella scrittura dalla quale possiamo leggere la segnatura, c'è bisogno che tutti i pezzi siano linearmente indipendenti, dette perciò $c_1, \dots, c_k$ le radici reali distinte di p con $e_1, \dots, e_k$ le rispettive molteplicità e $\beta_1, \dots, \beta_{(r-k)/2}, \bar{\beta}_1, \dots, \bar{\beta}_{(r-k)/2}$ le restanti si ha

$$x^T H(p) x = \sum_{i=1}^{k} e_i^2 \left(\sum_{j=1}^{d} c_i^{j-1} x_j \right)^2 + \sum_{i=1}^{(r-k)/2} \left((\sum_{j=1}^{d} Re(\beta_i^{j-1}) x_j)^2 - (\sum_{j=1}^{d} Im(\beta_i^{j-1}) x_j)^2 \right)$$

Da cui $\sigma(H(p)) = k$, come volevasi dimostrare. □

Come nella sezione 1 siamo interessati anche al caso un cui vi sia una condizione di segno espressa da un certo $q(X) \in \mathbb{R}[X]$.

Definizione 6.12. Sia $i \in \mathbb{N}$ la **i-esima somma di Newton generalizzata** di p è

$$\hat{N}_i := \sum_{j=1}^{d} q(\alpha_j) \alpha_j^i$$

dove $\alpha_1, \dots, \alpha_d \in \mathbb{C}$ sono le radici di p contate con molteplicità.

$$H(p,q) = \begin{bmatrix} \hat{N}_0 & \hat{N}_1 & \dots & \hat{N}_{d-1} \\ \hat{N}_1 & \hat{N}_2 & \dots & \hat{N}_d \\ \vdots & \vdots & \vdots & \vdots \\ \hat{N}_{d-1} & \hat{N}_d & \dots & \hat{N}_{2d-2} \end{bmatrix}$$

Teorema 6.13. La segnatura $\sigma(H(p,q))$ è uguale alla differenza tra il numero di radici reali distinte c di p tali che $q(c) > 0$ e quelle per cui$q(c) < 0$; il $\mathrm{rk}(H(p,q))$ è uguale al numero totale di radici distinte di p che non annullano q.

Corollario 6.14. Il rango $\mathrm{rk}(H(p,p'))$ fornisce il numero di radici complesse semplici mentre la segnatura di $H(p,q^2)$ fornisce il numero di radici reali di p che non annullano q.

6.3.1 Calcolo della segnatura

Richiamiamo un importante fatto di algebra lineare:

Teorema 6.15 (Jacobi)**.** Sia V un $\mathbb{R}$-spazio vettoriale di dimensione d e sia

$$b\colon V \times V \longrightarrow \mathbb{R}$$

una forma bilineare con matrice associata A rispetto a una qualche base $\mathscr{B}$. Siano $\delta_1,\dots,\delta_d$ i determinanti dei minori principali di testa di A. E supponiamo che $\delta_i \neq 0$ per ogni i. Allora esiste una matrice ortogonale M tale che $MAM^T = \operatorname{diag}(\delta_1, \delta_2/\delta_1, \dots, \delta_d/\delta_{d-1})$. Inoltre, detto v il numero di cambiamenti di segno della sequenza $(1, \delta_1, \dots, \delta_d)$, si ha

$$\sigma(A) = d - 2v.$$

Per poter utilizzare questo risultato c'è bisogno che valga un'ipotesi molto forte, ossia che i determinanti dei minori principali di testa siano tutti nonnulli. Il nostro scopo è calcolare la segnatura della matrice $H(p)$, che sappiamo essere invertibile se e solo se le radici di p tutte distinte. Possiamo aggirare questo ostacolo sfruttando una particolare proprietà di queste matrici.

Definizione 6.16. Una matrice $A = (a_{ij})$ è detta **matrice di Henkel** se per ogni k i coefficienti a_{ij} tale che $i + j = k$ sono uguali.

Esempio 3.

$$\begin{bmatrix} a & b & c & d \\ b & c & d & e \\ c & d & e & f \\ d & e & f & g \end{bmatrix}$$

è una matrice di Henkel.

La matrice $H(p)$ è di Henkel. Grazie a Frobenius abbiamo un metodo per calcolare la segnatura usando solo i minori principali:

Proposizione 6.17. Sia $p(X) \in \mathbb{R}[X]$ un polinomio di grado d e siano $\delta_1,\dots,\delta_d$ i minori principali della matrice $H(p)$. Sia r tale che $\delta_r \neq 0$ e $\delta_{r+1} = \cdots = \delta_d = 0$ (nota che $r \geq 1$ visto che $\delta_1 = d$). Per $1 \leq i \leq r$ definiamo i **segni convenzionali** $\widetilde{\operatorname{sign}}(\delta_i)$ come segue:

1. se $\delta_i \neq 0$ allora $\widetilde{\operatorname{sign}}(\delta_i) = \operatorname{sign}(\delta_i)$
2. se $\delta_i = \delta_{i-1} \cdots = \delta_{i-j+1} = 0$ e $\delta_{i-j} \neq 0$

$$\widetilde{\operatorname{sign}}(\delta_i) = (-1)^{j(j-1)/2} \operatorname{sign}(\delta_{i-j})$$

Allora il numero di radici reali distinte di p è r meno due volte il numero di cambi di segno nella sequenza

$$(1, \widetilde{\operatorname{sign}}(\delta_1), \dots, \widetilde{\operatorname{sign}}(\delta_r)).$$

Quello che fa funzionare la proposizione 6.17 è il fatto che da un certo $r+1$ in poi i minori principali si annullano e che tale r è proprio il rango della matrice $H(p)$.
Non daremo la dimostrazione completa del criterio, ma introdurremo quanto basta per dimostrare che

$$\operatorname{rk}(H(p)) = r \iff \delta_r \neq 0 \text{ e } \delta_{r+1} = \cdots = \delta_d = 0.$$

Coefficienti sottorisultanti principali

Un classico strumento algebrico è il risultante di due polinomi che tra le altre cose può essere usato per dire se due polinomi hanno o meno fattori in comune, qui estenderemo questo concetto creando una successione di *coefficienti sottorisultanti principali* tramite la quale ottenere qualche informazione in più.

Definizione 6.18. Consideriamo $A(y) = a_0 y^d + \cdots + a_d$ e $B(y) = b_0 y^e + \cdots + b_e$ due polinomi tali che $a_0.b_0 \neq 0$ e la matrice

$$S = \begin{bmatrix} a_0 & \dots & a_d & & \\ & \ddots & & \ddots & \\ & a_0 & & \dots & a_d \\ & & b_0 & \dots & b_e \\ b_0 & \dots & b_e & & \end{bmatrix}$$

che è una permutazione della matrice di Sylvester di A e B. Definiamo $j-$esimo **coefficiente sottorisultante principale PSRC**$_j(A,B)$ per $j = 0,\dots,\min(e,d)$ come il determinate della sottomatrice quadrata di taglia $d+e-2j$ ottenuta da S eliminando le prime e le ultime j righe e j colonne.

Teorema 6.19. Sia s un intero $0 \leq s < \min\{d,e\}$. Allora $\deg(\gcd(A,B)) > s$ se e solo se

$$\mathrm{PSRC}_0(A,B) = \cdots = \mathrm{PSRC}_s(A,B) = 0.$$

Dimostrazione. Per induzione su s. Se $s = 0$ è il caso del risultante.
Se $s > 0$ si ha che $\deg(\gcd(A,B)) > s$ se e solo se esistono due polinomi U,V di grado rispettivamente minore si $d-s$ e $e-s$ tali che

$$\deg(AU+BV) > s \tag{6.4}$$

Supponiamo che valga questa disequazione, per ipotesi induttiva

$$\mathrm{PSRC}_0(A,B) = \cdots = \mathrm{PSRC}_{s-1}(A,B) = 0$$

Osserviamo che l'equazione 6.4 equivale ad un sistema lineare omogeneo di $d+e-2s$ incognite (i coefficienti di U,V) che ha come determinante proprio $\pm\mathrm{PSRC}_s(A,B)$. Dunque tale sistema ha soluzione se e solo se $\mathrm{PSRC}_s(A,B) = 0$.
Viceversa visto che i coefficienti sottorisultanti principali si annullano fino all's-esimo per ipotesi induttiva $\deg(AU+BV) \geq s$ e per quanto osservato vale la disuguaglianza stretta se e solo se se e solo se $\mathrm{PSRC}_s(A,B) = 0$. □

Torniamo al nostro scopo originale e facciamo vedere come si relazionano i coefficienti sottorisultanti principali con la matrice di Henkel:

Proposizione 6.20. Consideriamo $\{\delta_i\}_{0<j\leq d}$ i minori principali di $H(p)$ per

$$p(X) = X^d + a_1 X^{d-1} + \cdots + a_d \in \mathbb{R}$$

. Allora

$$a_0^{2j-1}\delta_j = \mathrm{PSRC}_{d-j}(P,P').$$

Dimostrazione. Grazie all'equazione 6.1 il prodotto tra le matrici[5] di taglia $2j-1$

$$A_1 = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & N_0 & \dots & N_{j-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & N_0 & \dots & N_{j-3} & N_{j-2} & \dots & N_{2j-3} \\ N_0 & N_1 & \dots & N_{j-2} & N_{j-1} & \dots & N_{2j-2} \end{bmatrix}$$

$$A_1 = \begin{bmatrix} a_0 & a_1 & \dots & a_{2j-2} \\ 0 & a_0 & \dots & a_{2j-1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & a_0 \end{bmatrix}$$

è proprio la matrice tramite cui si calcola $\mathrm{PSRC}_{d-j}(P,P')$ e per Binet

$$\mathrm{PSRC}_{d-j}(P,P') = \det(A_1 A_2) = \det A_1 \det A_2 = a_0^{2j-1}\delta_j.$$

□

Abbiamo introdotto questi coefficienti perché vale una proprietà molto utile:

Corollario 6.21. Dato $p \in \mathbb{R}[X]$ di grado d, sono fatti equivalenti:

i. p ha r radici complesse distinte.

ii. $\mathrm{rk}(H(p)) = r$.

iii. $\delta_r \neq 0$ e $\delta_j = 0$ per $r < j \leq d$.

iv. $\mathrm{PSRC}_{d-r}(p,p') \neq 0$ e $\mathrm{PSRC}_l(p,p') = 0$ per ogni $0 \leq l < d-r$.

Dimostrazione. i.⇒ ii. Teorema 6.11.
ii.⇒ iii. Il rango di una matrice è pari al massimo ordine di un suo minore invertibile.
iii.⇒ iv. Dato che p ha grado d, per la Proposizione 6.20 si ha

$$\mathrm{PSRC}_{d-r}(p,p') = a_0^{2r-1}\delta_r \neq 0$$

e per ogni $0 \leq l < d-r$

$$\mathrm{PSRC}_l(p,p') = a_0^{2(r+l)-1}\delta_{r+l} = 0.$$

iv.⇒ i. Dal Teorema 6.19 sappiamo che $d-r-1 < \deg(\gcd(p,p')) \leq d-r$ e quindi il numero delle radici distinte è $d-(d-r) = r$. □

Possiamo perciò applicare la Problema 6.17 a $H(p)$:

Corollario 6.22. Il numero di radici reali distinte di p dipende solo dai segni ($<, >$ o $=$) dei coefficienti sottorisultanti principali relativi a p e p'.

[5] Con la convenzione $a_s = 0$ se $s > d$

6.4 Insiemi algebrici

Prima di definire che cosa è un semialgebrico, è bene richiamare precisamente la nozione di insieme algebrico e cercare di capire la differenza tra gli insiemi algebrici reali e quelli complessi:

Definizione 6.23. Sia $\mathbb{K}$ un campo e $B \subseteq \mathbb{K}[X_1, \ldots, X_n]$. L'insieme degli zeri di B è

$$\mathcal{V}(B) := \{x \in \mathbb{K}^n \mid \forall f \in B \ f(x) = 0\}$$

Un **insieme algebrico** V è un sottoinsieme di $\mathbb{K}^n$ tale per cui esiste B e $V = \mathcal{V}(B)$.

Inoltre fissiamo la seguente notazione.

Definizione 6.24. Sia $\mathbb{K}$ un campo e $S \subseteq \mathbb{K}^n$. Indichiamo con

$$\mathcal{I}(S) := \{f \in \mathbb{K}[X_1, \ldots, X_n] \mid \forall x \in S \ f(x) = 0\}$$

l'ideale dei polinomi che si annullano su S.

Richiamiamo, senza troppi dettagli, alcuni risultati base di geometria algebrica (per una trattazione più completa e le dimostrazioni rimandiamo a [Shape]) sia perché ci serviranno più avanti per trattare alcune nozioni, sia per capire le peculiarità del caso reale.

Osserviamo come prima cosa che un insieme $A \subseteq \mathbb{K}^n$ è algebrico se e solo se $A = \mathcal{V}(\mathcal{I}(A))$. Definiamo per $S \subset \mathbb{K}^n$ **anello delle funzioni polinomiali su** S:

$$\mathcal{P}(S) = {}^{\mathbb{K}[X_1, \ldots, X_n]}\!/_{\mathcal{I}(S)}$$

Ovvero possiamo identificare $\mathcal{P}(S)$ con l'anello delle funzioni da S su $\mathbb{K}$ che sono restrizioni di polinomi; il lemma di normalizzazione di Noether ci dice anche che questo anello è intero su un anello di polinomi su $\mathbb{K}$.
Si dice che un insieme algebrico non vuoto $A \subseteq \mathbb{K}^n$ è **irriducibile** se non può essere scritto come unione di due suoi sottoinsiemi algebrici propri. Vale che se A è irriducibile allora $\mathcal{I}(A)$ è un ideale primo e quindi $\mathcal{P}(A)$ è un dominio, in particolare è ben definito il campo delle frazioni $k(A)$, il **campo delle funzioni razionali** su A.
La **dimensione** di un insieme algebrico A è definita come la dimensione di Krull di $\mathcal{P}(A)$, ossia la massima lunghezza di una catena di ideali primi di tale anello. Questa definizione prettamente algebrica permette di caratterizzare, nel caso l'insieme sia irriducibile, la dimensione come grado di trascendenza di $k(A)$ su $\mathbb{K}$. Dato un insieme algebrico A esiste unica **decomposizione in componenti irriducibili** non ridondante, ossia $A_1, \ldots, A_s$ algebrici irriducibili tali che $A_j \nsubseteq \cup_{k \neq j} A_k$ e $A = A_1 \cup \cdots \cup A_s$. Vale che:

Teorema 6.25.

i. Sia A un insieme algebrico e A_i con $i = 1 \ldots s$ le sue componenti irriducibili, allora $\dim(A) = \max_i\{\dim(A_i)\}$.

ii. $V \subseteq \mathbb{K}^n$ e $W \subseteq \mathbb{K}^m$ algebrici allora $V \times W$ è algebrico in $\mathbb{K}^n \times \mathbb{K}^m$ e vale che $\mathcal{P}(V \times W) = \mathcal{P}(V) \otimes_{\mathbb{K}} \mathcal{P}(W)$. Inoltre prodotto di irriducibili è irriducibile e $\dim(V \times W) = \dim V + \dim W$.

Se $S \subseteq \mathbb{K}^n$ è un sottoinsieme qualsiasi,

$$\mathrm{Clos}_Z(S) := \mathcal{V}(\mathcal{I}((S)))$$

è il più piccolo insieme algebrico che lo contiene ed è chiamato **chiusura di Zariski** di S. Sostanzialmente stiamo prendendo la chiusura rispetto alla topologia di Zariski su $S \subseteq \mathbb{K}^n$, i cui chiusi sono tutti e soli gli algebrici.

Concentriamoci adesso sul campo reale e quello complesso. Ogni insieme algebrico complesso $A \subseteq \mathbb{C}^n$ può essere visto come sottoinsieme di $\mathbb{R}^{2n}$ e gli insiemi algebrici reali che possono essere visti come realizzazione di un algebrico complesso godono di particolari proprietà, infatti nel capitolo 7 di [Shape] viene mostrato che:

Proposizione 6.26. Sia $A \subset \mathbb{C}^n$ un algebrico irriducibile, con dimensione complessa d, considerato come sottoinsieme algebrico di $\mathbb{R}^{2n}$. Allora:

1. A è connesso,

2. A è illimitato (eccetto se è un punto),

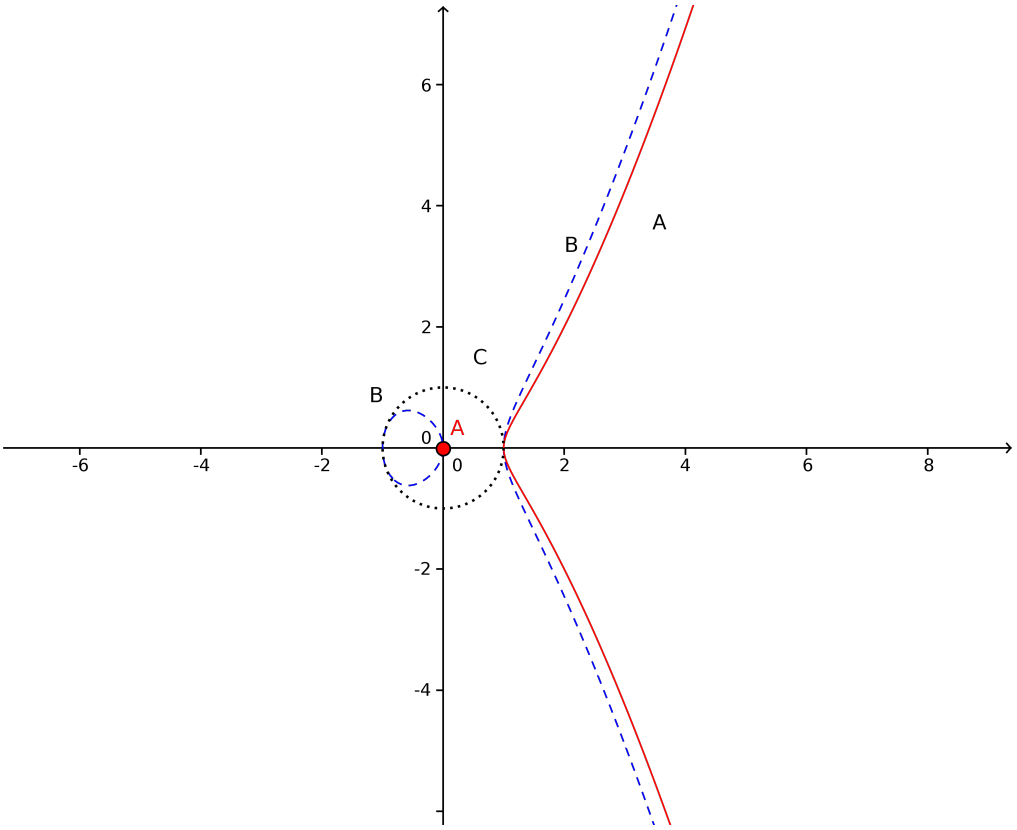
3. $\dim V_x$, ossia la dimensione locale in $x \in A$, è $2d$.

Questo significa che gli insiemi algebrici che possono essere scritti come algebrici complessi si comportano molto bene, ma le condizioni che li caratterizzano sono abbastanza restrittive. Già a partire da un solo polinomio si ottengono insiemi che non le soddisfano. Riportiamo qualche esempio che illustri diverse patologie:

Esempio 4.

1. I luoghi di zeri in $\mathbb{R}^2$ delle cubiche $y^2 - x^3 + x^2$ e $y^2 - x^3 + x$ hanno entrambi due componenti connesse, la prima ha un punto isolato, ma sono irriducibili.

2. $S^1 \subset \mathbb{R}^2$ è un insieme algebrico limitato.



3. L'ombrello di Cartan è la superficie $z(x^2+y^2)=x^3$ di $\mathbb{R}^3$, questa è irriducibile e connessa ma ha un "manico" in corrispondenza dell'asse z i cui punti hanno tutti dimensione 1.

6.5 Insiemi semialgebrici e stabilità per proiezione

Definizione 6.27. Un **insieme semialgebrico** di $\mathbb{R}^n$ è un insieme di punti

$$x=(x_1,\dots,x_n)\in\mathbb{R}^n$$

che soddisfano una combinazione booleana di equazioni e disequazioni polinomiali a coefficienti reali. In particolare, chiameremo semialgebrici la più piccola classe di sottoinsiemi di $\mathbb{R}^n$, che indicheremo con $\mathscr{SA}_n$, tale che

(i) Se $p\in\mathbb{R}[X_1,\dots,X_n]$, allora

$$\{x\in\mathbb{R}^n \mid p(x)=0\}\in\mathscr{SA}_n$$

$$\{x\in\mathbb{R}^n \mid p(x)>0\}\in\mathscr{SA}_n$$

(ii) È chiusa per unione finita, intersezione finita, passaggio al complementare.

Proposizione 6.28. Ogni $A\in\mathscr{SA}_n$ è unione finita di insiemi della forma

$$\{x\in\mathbb{R}^n \mid P(x)=0\wedge Q_1(x)>0\wedge\cdots\wedge Q_s(x)>0\}$$

con $P,Q_1,\dots,Q_s\in\mathbb{R}[X_1,\dots,X_n]$

Esempio 5.

1) In $\mathbb{R}$ i semialgebrici sono unioni finite di punti e intervalli aperti.

2) Gli algebrici sono semialgebrici.

3) Siano $F\colon \mathbb{R}^n \to \mathbb{R}^m$ mappa polinomiale e $A \in \mathscr{SA}_m$, allora $F^{-1}(A) \in \mathscr{SA}_n$.

4) $A \in \mathscr{SA}_m$ e $B \in \mathscr{SA}_n$ allora $A \times B \in \mathscr{SA}_{m+n}$.

Abbiamo enunciato il Teorema di Tarski-Seidenberg (6.8), facciamo vedere adesso che la chiusura rispetto alla proiezione della classe dei semialgebrici ne è un corollario:

Teorema 6.29 (**Tarski-Seidenberg [seconda forma]**)**.** Sia $A \in \mathscr{SA}_{n+1}$ e

$$\pi\colon \mathbb{R}^{n+1} \to \mathbb{R}^n$$

la proiezione sulle prime coordinate. Allora $\pi(A) \in \mathscr{SA}_n$.

Dimostrazione. A è unione finita di insiemi della forma $\{x \in \mathbb{R}^{n+1} \mid P(x) = 0 \wedge Q_1(x) > 0 \wedge \cdots \wedge Q_s(x) > 0\}$ per il teorema 6.8 esiste una combinazione booleana $\mathscr{C}(X_1, \ldots, X_n)$ di equazioni e disequazioni polinomiali tali che

$$\pi(A) = \{y \in \mathbb{R}^n \mid \exists x_{n+1} \in \mathbb{R}\ (y, x_{n+1}) \in A\}$$

sia l'insieme degli $y = (x_1, \ldots, x_n)$ tale che $\mathscr{C}(x_1, \ldots, x_n)$ sia vero, ossia $\pi(A)$ è semialgebrico. □

Corollario 6.30.

1. Se $A \in \mathscr{SA}_{n+k}$ e $\pi\colon \mathbb{R}^{n+k} \to \mathbb{R}^n$ allora $\pi(A) \in \mathscr{SA}_n$.
2. Se $A \in \mathscr{SA}_m$ e $F\colon \mathbb{R}^n \to \mathbb{R}^m$ mappa polinomiale allora $F(A) \in \mathscr{SA}_n$.

Dimostrazione. Il primo punto si fa per induzione. Per il secondo basta notare $B = \{(x, y) \in \mathbb{R}^{m+n} \mid x \in A \text{ e } y = F(x)\} \in \mathscr{SA}_{m+n}$ e $F(A) = \pi_m(B)$. □

Corollario 6.31. Se $A \in \mathscr{SA}_n$ la chiusura in $\mathbb{R}^n$

$$\mathrm{Clos}(A) = \{x \in \mathbb{R}^n \mid \forall \varepsilon \in \mathbb{R}\colon \varepsilon > 0 \Rightarrow \exists y \in \mathbb{R}^n,\ y \in A \wedge \|x - y\| < \varepsilon\}$$

è semialgebrica.

Dimostrazione. Per sfruttare le chiusura per proiezione dei semialgebrici cerchiamo una scrittura alternativa di questo insieme. Le variabili in questione formano una tripla

$$(x, \varepsilon, y) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}^n$$

e la condizione sulla norma può essere espressa con la disuguaglianza polinomiale

$$\sum_{i=1}^{n} (x_i - y_i)^2 < \varepsilon^2.$$

Consideriamo ora il semialgebrico

$$B = (\mathbb{R}^n \times \mathbb{R} \times A) \cap \{(x, \varepsilon, y) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}^n \mid \|x - y\| < \varepsilon\} \subseteq \mathbb{R}^{2n+1}$$

e le proiezioni

$$\begin{array}{rccc} \pi_1\colon & \mathbb{R}^{n+1} & \longrightarrow & \mathbb{R}^n \\ & (x,\varepsilon) & \longmapsto & x \\ \pi_2\colon & \mathbb{R}^{2n+1} & \longrightarrow & \mathbb{R}^{n+1} \\ & (x,\varepsilon,y) & \longmapsto & (x,\varepsilon) \end{array}$$

allora

$$\mathrm{Clos}(A) = \mathbb{R}^n \setminus (\pi_1(\{(x,\varepsilon) \in \mathbb{R}^n \times \mathbb{R} \mid \varepsilon > 0\} \setminus \pi_2(B)))$$

e quindi è algebrico. □

È utile anche un terza forma del teorema in termini logici, più facile da usare. Dobbiamo però richiamare qualche definizione per enunciarla.
Per noi una **formula al primo ordine** è una proposizione ottenuta con le seguenti regole:

1. Dato $p \in \mathbb{R}[X_1, \ldots, X_n]$, $\Lambda \leftarrow p = 0, p > 0$
2. $\Lambda \leftarrow (\Phi \vee \Psi), (\Phi \wedge \Psi), (\neg\Psi)$
3. $\Lambda \leftarrow (\exists x \Psi), (\forall x \Psi)$

dove con le lettere greche abbiamo indicato formule al prim'ordine e con x un elemento di $\mathbb{R}$; in pratica stiamo dicendo che ad esempio $\Lambda = (\forall x \neg(\Phi \vee \Psi))$ è una formula al prim'ordine. Di solito se la formula dipende da uno o più parametri lo indichiamo tra parentesi, nell'esempio allo avremmo potuto scrivere $\Lambda(x)$. Inoltre, se una formula è ottenuta usando solo i primi due punti è detta *libera da quantificatori*.

Osservazione 6.5. Un sottoinsieme $A \subseteq \mathbb{R}^n$ è semialgebrico se e solo se esiste una formula libera da quantificatori Φ tale che

$$(x_1, \ldots, x_n) \in A \iff \Phi(x_1, \ldots, x_n)$$

Teorema 6.32 (**Tarski-Seidenberg [terza forma]**)**.** Se $\Phi(X_1, \ldots, X_n)$ è una formula al prim'ordine, l'insieme dei $(x_1, \ldots, x_n) \in \mathbb{R}^n$ che soddisfano tale formula formano un semialgebrico.

Dimostrazione. Per induzione sulla costruzione della formula. La prima regola produce solo sottoinsiemi semialgebrici; l'applicazione della seconda, visto che è un operazione finita, produce un semialgebrico. Se l'insieme

$$\{(x_1, \ldots, x_{n+1}) \in \mathbb{R}^{n+1} \mid \Psi(x_1, \ldots, x_{n+1})\}$$

è semialgebrico allora

$$\{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid \exists y \in \mathbb{R}\colon \Psi(x_1, \ldots, x_n, y)\}$$

è la sua proiezione sulle prime n coordinate e dunque è semialgebrico. Per concludere riguardo le formule ottenute con la terza regola, osserviamo che

$$\forall x \Psi \iff \neg\exists x \neg\Psi.$$

□

Osservazione 6.6. Stiamo dicendo in poche parole che ogni formula al prim'ordine è equivalente ad una formula libera da quantificatori.

6.6 Mappe semialgebriche

Introduciamo adesso i morfismi che meglio si adattano a alla classe dei semialgebrici:

Definizione 6.33. Siano $A \in \mathscr{SA}_m$ e $B \in \mathscr{SA}_n$, una $f\colon A \to B$ è detta **mappa semialgebrica** se il suo grafico

$$\Gamma_f = \{(x,y) \in \mathbb{R}^{m+n} \mid y = f(x)\}$$

è semialgebrico.

Esempio 6.

- Le mappe polinomiali e le le funzioni regolari sono semialgebriche.
- Se $f\colon A \to B$ è semialgebrica allora $|f|$ è semialgebrica.
- Se $f\colon A \to B$ è semialgebrica e positiva su tutto il dominio allora $\sqrt{f}$ è semialgebrica.

Usando le proprietà sopra enunciate, in particolare la chiusura rispetto alla proiezione, otteniamo che:

Proposizione 6.34.

1. Se $f\colon A \to B$ è semialgebrica allora $f^{-1}(B)$ e $f(A)$ sono semialgebrici.
2. $f\colon A \to B$ e $g\colon B \to C$ mappe semialgebriche, allora $g \circ f$ è semialgebrica.
3. L'insieme $\mathscr{S}_A F := \{f\colon A \to \mathbb{R} \mid \text{mappa semialgebrica}\}$ è un anello.

A partire dalla proposizione sopra enunciata e notando che l'identità è una mappa semialgebrica e le mappe semialgebriche sono associative, si può dire che i semialgebrici con queste frecce sono una categoria.

Nei contesti dove dir ciò ha senso, le mappe semialgebriche si comportano bene definitivamente.

Proposizione 6.35. Sia $f\colon (a,+\infty) \to \mathbb{R}$ una mappa semialgebrica non necessariamente continua. Allora esistono $b \geq a$ e un intero positivo n tale che $|f| \leq x^n$ per ogni $x \in (b,+\infty)$.

Dimostrazione. Per definizione il grafico di f è un semialgebrico e dunque si scrive come una unione finita $\Gamma = G_1 \cup \cdots \cup G_l$ con

$$G_i = \{(x,y) \in \mathbb{R}^2 \mid P^i(x,y) = 0 \wedge Q_1^i(x,y) > 0 \wedge \cdots \wedge Q_{s_i}^i(x,y) > 0\}.$$

Osserviamo se se esistesse un j tale che $\deg_y P^j = 0$, allora se un punto $(x_0,y_0) \in G_j$ avremmo che tutto un intervallo di $\{x_0\} \times \mathbb{R}$ sta G_j, ma questo è assurdo per la definizione di funzione (avrei più immagini per un punto). Possiamo quindi assumere $\deg_y P^i > 0$ per ogni i e ha senso la scrittura

$$\Lambda(x,y) = \prod_{i=1}^{l} P^j(x,y) = a_0(x)\,y^d + \cdots + a_d(x)$$

con $d > 0$ e $a_0 \neq 0$. Possiamo quindi scegliere $c \geq a$ tale che $a_0(x) \neq 0$ per $x \in (c, +\infty)$.
Gli elementi che si scrivono come $f(x)$, ossia che stanno nell'immagine della mappa, sono radici di Λ e quindi

$$|f(x)| \leq \max_i \left(d \left| \frac{a_i(x)}{a_0(x)} \right| \right)^{\frac{1}{i}}$$

facendo il limite per x che va a più infinito si vede che questo massimo si comporta come x^α con $\alpha \in \mathbb{Q}$. Basta scegliere n come la parte intera superiore di $\mathbb{Q}$ e scegliere $b > c$. □

6.7 Decomposizione di insiemi semialgebrici

Ogni $A \in \mathscr{SA}_1$ si scrive come unione finita di punti e intervalli aperti. Il nostro scopo per il resto di questo sottocapitolo sarà mostrare che è possibile decomporre, per un n qualsiasi, ogni semialgebrico in $\mathscr{SA}_n$ come unione disgiunta di pezzi che sono semialgebricamente omeomorfi a ipercubi aperti di dimensione d, ossia $(0,1)^d$. Notiamo che assumendo che per $d = 0$ l'ipercubo sia un punto, nel caso di $\mathbb{R}$ ritroviamo quanto sapevamo.
È bene precisare la seguente nozione:

Definizione 6.36. Un omeomorfismo semialgebrico $h\colon S \to T$ è una mappa continua biunivoca e semialgebrica con inversa continua.

Osservazione 6.7. La condizione che l'inversa sia semialgebrica è necessaria visto che $\Gamma_h = \Gamma_{h^{-1}}$.

Introduciamo ora la *Decomposizione algebrica cilindrica* (CAD) di $\mathbb{R}^n$ che, vedremo, sarà risolutiva rispetto lo scopo che ci siamo prefissi, la quale è anche particolarmente interessante per due motivi: il primo è che ne esiste un "raffinamento" tale per cui è possibile che una famiglia di polinomi abbia segno costante su ciascun pezzo della decomposizione, il secondo è che questa procedura è intrinsecamente algoritmica.

Definizione 6.37. Una **decomposizione algebrica cilindrica** di $\mathbb{R}^n$ è un lista $\mathscr{C}_1, \ldots, \mathscr{C}_n$, dove per $1 \leq k \leq n$ l'insieme $\mathscr{C}_k$ è una partizione di $\mathbb{R}^k$ in sottoinsiemi semialgebrici, detti **celle**, che soddisfano le seguenti proprietà:

(a) Ogni $C \in \mathscr{C}_1$ è un punto o un intervallo aperto.

(b) Per ogni $k = 1, \ldots, n-1$ e per ogni cella $C \in \mathscr{C}_k$ esistono delle funzioni semialgebriche continue

$$\xi_{C,1} < \cdots < \xi_{C,l_C}\colon C \to \mathbb{R}$$

tali per cui il cilindro $C \times \mathbb{R} \subseteq \mathbb{R}^{k+1}$ sia unione disgiunta di celle di $\mathscr{C}_{k+1}$ che sono di esattamente uno dei seguenti tipi:

(i) il **grafico** di una $\xi_{C,j}$ per $j = 1, \ldots, l_C$

$$A_{C,j} = \{(x', x_{k+1}) \in C \times \mathbb{R} \mid x_{k+1} = \xi_{C,j}(x')\}$$

(ii) una **banda** del cilindro limitata superiormente e inferiormente rispettivamente dai due grafici delle $\xi_{C,j}$ e $\xi_{C,j+1}$ con $j = 0, \ldots, l_C + 1$ (dove poniamo $\xi_{C,0} = -\infty$ e $\xi_{C,l_C+1} = +\infty$)

$$B_{C,j} = \{(x', x_{k+1}) \in C \times \mathbb{R} \mid \xi_{C,j}(x') < x_{k+1} < \xi_{C,j+1}(x')\}$$

Lemma 6.38. Ogni cella di una CAD $\mathscr{C}_1, \ldots, \mathscr{C}_n$ è semialgebricamente omeomorfa a $(0,1)^d$ per d opportuno.

Dimostrazione. Per induzione su k; $k = 1$ è vero per definizione. Se $k > 1$, data $C \in \mathscr{C}_k$ per costruzione abbiamo che ogni $A_{C,j}$ è omeomorfo semialgebricamente a C tramite

$$\begin{array}{rccc} \varphi\colon & C & \to & A_{C,j} \\ & x' & \mapsto & (x', \xi_{C,j}(x')) \end{array}$$

mentre ogni $B_{C,j}$ è omeomorfa a $C \times (0,1)$ tramite la mappa

$$\lambda(x',t) = \begin{cases} (x', (1-t)\xi_{C,j}(x') + t\xi_{C,j+1}(x')) & 0 < j < l_C \\ (x', -\frac{1}{t} + \frac{1}{1-t}) & j = l_C = 0 \\ (x', -\frac{1-t}{t} + \xi_{C,1}(x')) & j = 0, l_C \neq 0 \\ (x', \frac{t}{t-1} + \xi_{C,l_C}(x')) & j = l_C \neq 0 \end{cases}$$

Applicando l'ipotesi induttiva abbiamo la tesi. □

Abbiamo anticipato che è possibile costruire la decomposizione in modo che sui pezzi della CAD, che ora sappiamo chiamarsi celle, una famiglia di polinomi abbia segno costante:

Definizione 6.39. Sia $(P_1, \ldots, P_r)$ una r-upla di polinomi in $\mathbb{R}[X_1, \ldots, X_n]$. Diremo che un sottoinsieme $C \subseteq \mathbb{R}^n$ è $(P_1, \ldots, P_r)$-**invariante** se ogni P_j ha segno segno costante su C per $j = 1, \ldots, r$.

Definizione 6.40. Siano $(P_1, \ldots, P_r)$ una r-upla di polinomi in $\mathbb{R}[X_1, \ldots, X_n]$ e $\mathscr{C}_1, \ldots, \mathscr{C}_n$ una CAD di $\mathbb{R}^n$; diremo che è **adatta** a $(P_1, \ldots, P_r)$ se vale anche

(c) Ogni cella $C \in \mathscr{C}_n$ è $(P_1, \ldots, P_r)-$invariante.

Soffermiamoci un attimo sul perché è sensato aggiungere questa ipotesi. La nostra attenzione, ricordiamo, è rivolta allo studio degli insiemi semialgebrici, i quali possono essere espressi tramite combinazioni booleane di equazioni e disequazioni polinomiali; questo vuol dire che se prendiamo una decomposizione algebrica cilindrica adatta alla famiglia dei polinomi che descrivono un fissato $S \in \mathscr{SA}_n$ questo si scriverà come unione finita di alcune delle celle di grado n. Dunque dimostrando che a partire dalle equazioni che descrivono un semialgebrico siamo in grado di costruire una CAD, avremo anche provato che esso è semialgebricamente omeomorfo ad un'unione finita di ipercubi aperti di dimensione al più n.

Rimane da dimostrare che una decomposizione algebrica cilindrica esiste; riportiamo qui una dimostrazione costruttiva, la quale ci fornisce anche un algoritmo esplicito.
Enunciamo un paio di fatti che ci serviranno per far ciò (cerchiamo di mantenere una notazione coerente con la costruzione anche se i fatti valgono più in generale).

Lemma 6.41. Sia $P(X_1,\dots,X_n) \in \mathbb{R}[X_1,\dots,X_n]$ e $C \subseteq \mathbb{R}^{n-1}$ un sottoinsieme semialgebrico connesso e $0 < k \leq d$ interi tale che per ogni $\tilde{x} \in C$ il polinomio $P(\tilde{x}, X_n)$ ha grado d ed esattamente k radici distinte in $\mathbb{C}$. Allora esistono $l \leq k$ funzioni semialgebriche continue

$$\xi_1 < \cdots < \xi_l \colon C \to \mathbb{R}$$

tali che per ogni $\tilde{x} \in C$ l'insieme delle radici reali di $P(\tilde{x}, X_n)$ è $\{\xi_1(\tilde{x}),\dots,\xi_l(\tilde{x})\}$. Inoltre per ogni $i = 1,\dots,l$, la molteplicità di $\xi_i(\tilde{x})$ è costante al variare del punto in C.

Prima di dimostrare il lemma enunciamo (la dimostrazione si fa come nel caso generale) il principio di continuità delle radici in queste particolari ipotesi:

Proposizione 6.42. Sia $P(X_1,\dots,X_n) \in \mathbb{R}[X_1,\dots,X_n]$ e $\tilde{a} \in C \subseteq \mathbb{R}^{n-1}$ e siano $z_1,\dots,z_k$ radici distinte di $P(\tilde{a}, X_n)$ con $z_1,\dots,z_k$ le rispettivamente molteplicità. Dato $\varepsilon > 0$ tale che i dischi D_j di centro z_j e raggio ε, contenuti in $\mathbb{C}$ per ogni j, siano tutti disgiunti ; se $\tilde{b}$ è sufficientemente vicino ad $\tilde{a}$ allora $P(\tilde{b}, X_n)$ ha esattamente m_j radici in D_j per ogni j.

Dimostriamo il Lemma 6.41:

Dimostrazione. Consideriamo $P(X_1,\dots,X_n) \in \mathbb{R}[X_1,\dots,X_n]$. Per la proposizione 6.42, con la stessa notazione, preso $\tilde{b} \in C$ il polinomio $P(\tilde{b}, X_n)$ in ogni disco D_j di centro z_j e raggio ε ha esattamente una radice di molteplicità m_j per le ipotesi fatte su C (il numero di radici distinte è fissato): chiamiamo questa radice ζ_j. Osserviamo che se z_j è reale lo è anche ζ_j,. infatti agendo per coniugazione dovremmo poter rimanere nella stesso disco.
Se $\tilde{a} \in C$ e $\tilde{b} \in C$ sono abbastanza vicini allora i polinomi $P(\tilde{a}, X_n)$ e $P(\tilde{b}, X_n)$ hanno lo stesso numero di radici reali; ma visto che C è connesso in realtà questo vale per tutti i punti e dunque il numero di radici reali è costante su tutto C e verrà indicato con l. Definiamo allora per ogni $1 \leq i \leq l$

$$\begin{array}{rccc} \xi_i \colon & C & \to & \mathbb{R} \\ & \tilde{x} & \mapsto & \zeta_{\tilde{x},i} \end{array}$$

dove $\zeta_{\tilde{x},1} < \cdots < \zeta_{\tilde{x},l}$ sono le radici reali di $P(\tilde{x}, X_n)$. Ovviamente le ξ_i sono continue e $\xi_i(\tilde{x})$ ha molteplicità costante come radice. Inoltre visto che C è semialgebrico esiste una formula $\Theta(\tilde{x})$ che lo descrive e i grafici delle ξ si possono scrivere nel seguente modo:

$$\begin{aligned} \Gamma_{\xi_i} = \{(\tilde{x}, x_n) \in \mathbb{R}^n \mid \Theta(\tilde{x}) \wedge (\exists y_1 < y_2 \cdots < y_l \wedge \\ P(\tilde{x}, y_1) = 0 \wedge \cdots \wedge P(\tilde{x}, y_l) = 0 \wedge x_n = y_i)\} \end{aligned}$$

E dunque le ξ_i sono anche semialgebriche. □

Se la famiglia di polinomi constasse solo di un elemento, questo lemma sarebbe sufficiente, tuttavia questo è un caso molto speciale. In generale è opportuno capire come interagiscono tra di loro queste funzioni se i polinomi in gioco sono almeno due, anzi senza perdita di generalità possiamo ridurci a studiare proprio questo caso. Faremo vedere infatti che se due funzioni α e β trovate come nel lemma precedente (rispettivamente a due diversi polinomi) coincidono in un punto allora coincidono su tutto il dominio C; nell'ottica di costruire una CAD questo ci dice che quelle che avevamo indicato con $\xi_{C,1} < \cdots < \xi_{C,s_C}$ possono essere scelte come l'unione delle mappe semialgebriche che troviamo grazie al Lemma 6.41 per ognuno dei due polinomi.

Lemma 6.43. Siano $P, Q \in \mathbb{R}[X_1, \ldots, X_n]$ e $C \in \mathbb{R}^{n-1}$ un semialgebrico connesso tale che per ogni $\tilde{x} \in C$ siano costanti

- il grado e il numero di radici distinte di $P(\tilde{x}, X_n)$ (risp. $Q(\tilde{x}, X_n)$)
- il grado in X_n del $\gcd(P(\tilde{x}, X_n), Q(\tilde{x}, X_n))$.

e siano $\alpha, \beta \colon C \to \mathbb{R}$ due mappe semialgebriche continue tali che per ogni $\tilde{x} \in C$ $P(\tilde{x}, \alpha(\tilde{x})) = 0$ e $Q(\tilde{x}, \beta(\tilde{x})) = 0$. Allora se esiste $\tilde{a} \in C$ per cui $\alpha(\tilde{a}) = \beta(\tilde{a})$ necessariamente $\alpha \equiv \beta$.

Dimostrazione. Indichiamo con $z_1 = \alpha(\tilde{a}) = \beta(\tilde{a}), z_2, \ldots, z_k$ le radici distinte in $\mathbb{C}$ del polinomio prodotto $P(\tilde{a}, X_n)Q(\tilde{a}, X_n)$ e con m_i e p_j per $i, j = 1, \ldots, k$ le rispettive molteplicità[6] come radici di $P(\tilde{a}, X_n)$ e $Q(\tilde{a}, X_n)$. Allora il grado di $\gcd(P(\tilde{a}, X_n), Q(\tilde{a}, X_n))$ è esattamente $\sum_i^k \min\{m_i, p_i\}$.
Scegliamo $\varepsilon > 0$ in modo che $D_j = D(\varepsilon, z_j)$ siano aperti disgiunti in $\mathbb{C}$, allora per la Proposizione 6.42 per ogni $\tilde{x} \in C$ vicino ad $\tilde{a}$ si ha che ogni disco contiene una radice di molteplicità m_j di $P(\tilde{x}, X_n)$ e una di molteplicità p_j di $Q(\tilde{x}, X_n)$. Il grado del gcd è costante su C per ipotesi si ha anche che questo ha una radice in ogni disco di molteplicità $\min\{m_j, p_j\}$ (dove questa quantità è non nulla). Per connessione allora deve valere che per ogni $\tilde{x} \in C$ $\alpha(\tilde{x}) = \beta(\tilde{x})$. □

Abbiamo fin qui gettato le basi per quello che sarà il passo induttivo dell'algoritmo. Vengono in nostro aiuto ora i *coefficienti sottorisultanti principali*. Sia $(P_1, \ldots, P_r)$ una $r-$upla di polinomi in $\mathbb{R}[X_1, \ldots, X_n]$. Possiamo vedere ogni P_j come polinomio in X_n e definiamo **PROJ**$(P_1, \ldots, P_r)$ la più piccola famiglia di polinomi in $\mathbb{R}[X_1, \ldots, X_n]$ tali che per $i, j, k = 1, \ldots, r$

1. se $\deg_{X_n} P_j = d > 2$ allora $\mathrm{PROJ}(P_1, \ldots, P_r)$ contiene tutti i polinomi non costanti tra i $\mathrm{PSRC}_s(P_j, \frac{\partial P_j}{\partial_{X_n}})$ per $s = 0 \ldots d-1$;
2. se $1 \leq d = \min\{\deg_{X_n} P_i, \deg_{X_n} P_k\}$ allora $\mathrm{PROJ}(P_1, \ldots, P_r)$ contiene tutti i polinomi non costanti tra i $\mathrm{PSRC}_s(P_k, P_i)$ per $s = 0 \ldots d$;
3. se $\deg_{X_n} P_j \geq 2$ e $\mathrm{lc}_{x_n}(P_j) \notin \mathbb{R}$ allora $\mathrm{PROJ}(P_1, \ldots, P_r)$ contiene $\mathrm{lc}_{X_n}(P_j)$ e $\mathrm{PROJ}(P_1, \ldots, Q, \ldots, P_r)$ dove Q è la *coda* di P_j, ossia[7] $Q = P_j - \mathrm{lt}_{X_n}(P_j)$;
4. se $\deg_{X_n} P_j = 0$ e P_j non costante allora $P_j \in \mathrm{PROJ}(P_1, \ldots, P_r)$.

Abbiamo dato tutti gli ingredienti che ci permettono di enunciare il teorema giustifica quello che sarà il passo induttivo:

Teorema 6.44. Sia $(P_1, \ldots, P_r)$ una $r-$upla di polinomi in $\mathbb{R}[X_1, \ldots, X_n]$ e $C \subseteq \mathbb{R}^{n-1}$ un sottoinsieme connesso semialgebrico e $\mathrm{PROJ}(P_1, \ldots, P_r)$-invariante. Allora esistono

$$\xi_1 < \cdots < \xi_l \colon C \to \mathbb{R}$$

[6] Ricordiamo che per convenzione dicendo che $z \in \mathbb{C}$ ha molteplicità zero come radice di un certo polinomio p si intende che $p(z) \neq 0$.
[7] Con $\mathrm{lt}(F)$ si intende il termine di testa del polinomio F.

funzioni semialgebriche continue tali che $\{\xi_1(\tilde{x}),\dots,\xi_l(\tilde{x})\}$ sia l'insieme delle radici reali di $P_1(\tilde{x},X_n),\dots,P_r(\tilde{x},X_n)$, non nulli, per ogni $\tilde{x}\in C$. Inoltre i grafici di ogni ξ_i e le bande del cilindro $C\times\mathbb{R}$ delimitate da tali grafici sono

- insiemi semialgebrici connessi,
- semialgebricamente omeomorfi rispettivamente a C e $C\times(0,1)$,
- $(P_1,\dots,P_r)$-invarianti.

Dimostrazione. Per dimostrare la tesi ci basta far vedere che le ipotesi dei Lemmi 6.41 e 6.43 sono soddisfatti. In particolare capiamo cosa vuol dire che C è PROJ$(P_1,\dots,P_r)$-invariante: le condizioni 3. e 4. ci danno che il grado rispetto X_n è costante su C e quindi lo è anche il numero di radici come polinomi nell'ultima variabile; la condizione 1. invece dà che il gcd tra ogni polinomio e la sua derivata abbia grado costante e unito a quelle precedenti dunque che il numero di zeri distinti sia costante; infine la condizione 2. ci dà che il numero di zeri comuni di ogni coppia di polinomi sia costante. □

Possiamo costruire quindi una CAD adatta iterativamente. Ripetendo PROJ $n-1$ volte arriviamo a una famiglia finita di polinomi nella variabile X_1, per i quali è facile costruire una CAD adattata di $\mathbb{R}$ (i loro zeri reali tagliano la retta reale in un numero finito di punti e intervalli aperti); a questo punto grazie al Teorema 6.44 siamo in grado di "rimontare" i pezzi. Otteniamo dunque che:

Corollario 6.45. Per ogni $(P_1,\dots,P_r)$ una r-upla di polinomi in $\mathbb{R}[X_1,\dots,X_n]$ esiste una CAD di $\mathbb{R}^n$ $(P_1,\dots,P_r)$-adatta.

Vediamo prima di parlare di algoritmo, un esempio di questa costruzione:

Esempio 7. Costruiamo un CAD di $\mathbb{R}^3$ adatta a $p(x,y,z)=x^2+y^2+z^2-1$.

Step 1 Calcoliamo iterativamente PROJ.

- $\partial_z p=2z$, $\mathrm{PSRC}_0=-4(x^2+y^2+1)$ e $\mathrm{PSRC}_1=2$ allora PROJ $=\{-(x^2+y^2-1)\}$;
- Sia $q=-(x^2+y^2-1)$ allora $\partial_y(q)=-2y$, $\mathrm{PSRC}_0=x^2-1$ e $\mathrm{PSRC}_1=-1$ allora $\mathrm{PROJ}^2(p)=\{x^2-1\}$;

Indichiamo con $r(x)=x^2-1=(x-1)(x+1)$, questo polinomio ha due radici in $\mathbb{R}$ 1 e -1, dunque

$$\mathscr{C}_1=\{(-\infty,-1),\{-1\},(-1,1),\{1\},(1,\infty)\}.$$

Step 2 Calcoliamo $\mathscr{C}_2$

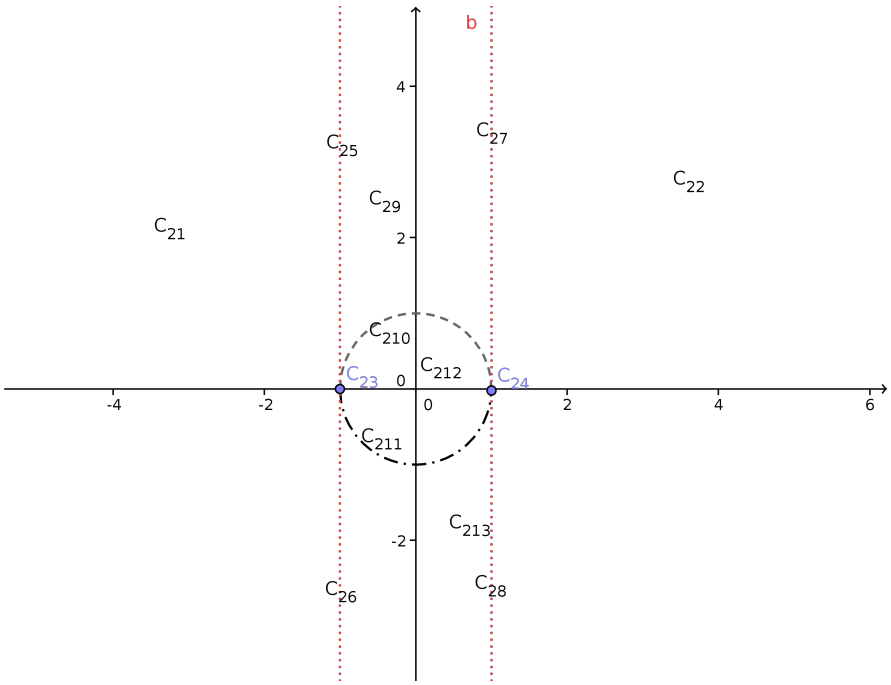
- Fissiamo $a\in(-\infty,-1)=C_{1,1}$, allora $q(a,y)=-(y^2+a^2-1)$ non ha radici reali. Allora otteniamo una banda $C_{2,1}=(-\infty,-1)\times\mathbb{R}$.
- Analogamente al caso precedente da $(1,\infty)$ otteniamo $C_{2,2}=(1,\infty)\times\mathbb{R}$.
- Fissiamo il punto $\{-1\}$, $q(-1,y)=-y^2$ che si annulla solo per $y=0$ e quindi otteniamo che $C_{2,3}$ è il grafico di $\xi_{-1}\equiv 0$ più due bande.

- Analogamente al caso precedente da $\{1\}$ otteniamo di nuovo $C_{2,4}$ più due bande.
- Fissiamo $a \in (-1,1)$ ci sono due casi casi

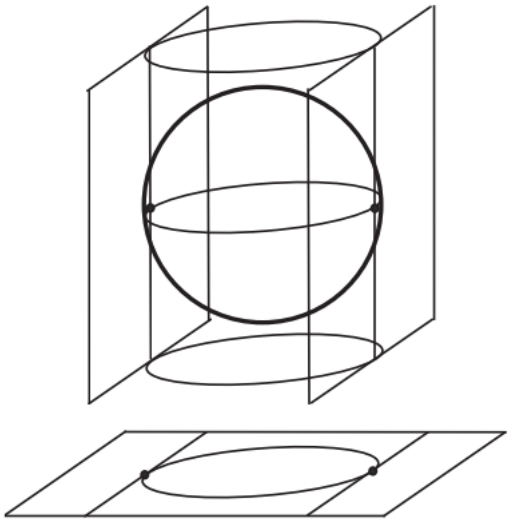
$$\begin{cases} \text{se } y>0 \; y=\sqrt{1-a^2} \\ \text{se } y>0 \; y=-\sqrt{1-a^2} \end{cases}$$

Otteniamo quindi due grafici e tre bande.

Graficamente $\mathscr{C}_2$ risulta:



Step 3 In breve per quanto riguarda $\mathscr{C}_3$: su $C_{2,1}$,$C_{2,2}$,$C_{2,5}$,$C_{2,6}$,$C_{2,7}$, $C_{2,8}$ il polinomio p non ha radici, su $C_{2,3}$, $C_{2,4}$, $C_{2,11}$ e $C_{2,10}$ una sola radice e su $C_{2,12}$ abbiamo otto possibili grafici. La decomposizione è schematizzabile come segue:



Algoritmo CAD

L'algoritmo prende in input una lista di polinomi $P_1,\dots,P_r \in \mathbb{Q}[X_1,\dots,X_n]$ e produce in output le celle di una CAD adatta a $P_1,\dots,P_r$ (con qualche informazione sull'arrangiamento dei cilindri) e un **punto test** per ogni cella a coordinate razionali o reali (algebriche).
Vediamo a grandi linee come funziona:

Passo Base Data un lista di polinomi univariati in X_1 calcola e isola in intervalli con estremi razionali le radici reali di tali polinomi (questo si può fare usando il metodo di Sturm).
Le celle $\mathscr{C}_1$ sono le radici e gli intervalli tra esse. visto che stiamo lavorando con numeri di macchina, le radici sono caratterizzate dalle equazioni polinomiali che soddisfano e gli intervalli ad estremi razionali che le isolano. In particolare questi punti razionali possono essere presi ad esempio come punti test.

Passo Induttivo Data una lista di polinomi $P_1,\dots,P_r \in \mathbb{Q}[X_1,\dots,X_n]$ con $n>1$ calcola

$$\mathrm{PROJ}(P_1,\dots,P_r) \subseteq \mathbb{Q}[X_1,\dots,X_{n-1}]$$

e richiama l'algoritmo su questi polinomi.
Una volta ottenuto $\mathscr{C}_{n-1}$ che è una partizione di $\mathbb{R}^{n-1}$ $\mathrm{PROJ}(P_1,\dots,P_r)$-invariante in celle e un punto test $\tilde{a}_C$ per ogni cella, usando il Teorema 6.44 per ogni $C \in \mathscr{C}_{n-1}$ taglia il cilindro $C\times\mathbb{R}$ in celle $(P_1,\dots,P_r)$-invarianti.
Per sapere quante celle ci sono e calcolare i punti test si possono calcolare le radici reali di $P_1(\tilde{a}_C,X_n),\dots,P_r(\tilde{a}_C,X_n)$.
Attenzione! per calcolare le radici reali bisogna tenere presente che tali polinomi possono essere a coefficienti reali algebrici.

Osservazione 6.8. L'algoritmo della CAD può essere usato anche per decidere se un formula senza variabili libere è vera o meno (*problemi di decisione*. Più generale data una formula al prim'ordine la CAD ci dice se il semialgebrico che definisce è vuoto o meno e in tal caso ne fornisce un punto razionale per ogni componente connessa.

Complessità. La complessità è doppiamente esponenziale nel numeratoro delle variabili: il risultante rispetto ad X_n di grado totale d ha in generale grado d^2, iterando la costruzione di PROJ $n-1$ volte arriviamo a polinomi univariati di massimo grado $d^{2^{n-1}}$.

6.7.1 Componenti connesse di un insieme semialgebrico

L'esistenza di una CAD ha molte conseguenze e corollari, ad esempio come vedremo nel prossimo capitolo permette di definire la nozione di dimensione. Qui la useremo per dimostrare che un semialgebrico si scrive come unione di una quantità finita di componenti connesse.

Teorema 6.46. Ogni insieme semialgebrico ha una quantità finita di componenti connesse semialgebriche. Inoltre ogni insieme semialgebrico è localmente connesso.

Dimostrazione. Sia $S \in \mathscr{SA}_n$, il Corollario 6.45 e la teoria sviluppata riguardo la CAD implicano che S può essere scritto come unione disgiunta di $C_1, \ldots, C_p$ semialgebrici che sono semialgebricamente omeomorfi a $(0,1)^{d_i}$ per $i = 1, \ldots, p$ (vedi Lemma 6.38); ovviamente questi insiemi sono connessi.
Vogliamo quindi trovare le componenti connesse di S a partire dalle celle. Diremo che C_i e C_j sono adiacenti se $C_i \cap \mathrm{Clos}(C_j) \neq 0$ e indicheremo con $\sim$ la relazione d'equivalenza sulle celle indotta dall'adiacenza. Dunque $C_i \sim C_j$ se esiste una sequenza ordinata $C_i, \ldots, C_k, \ldots, C_j$ di celle adiacenti. Allora abbiamo che $S_1, \ldots, S_r$, dove ogni S_i è unione massimale di celle adiacenti, è una partizione di S in semialgebrici disgiunti. Osserviamo che ogni S_i è chiuso in S, infatti se $C_j \cap \mathrm{Clos}(S_i) \neq 0$ allora C_j è adiacenti ad almeno una cella di S_i e dunque è $C_j \subseteq S_i$. Tuttavia visto che il complementare è unione finita di chiusi, ogni S_i è anche aperto.
Rimane da far vedere che sono connessi. Supponiamo di avere $S_i = F_1 \sqcup F_2$ unione disgiunta di due chiusi, allora si deve avere sia che ogni cella appartiene è contenuta (per connessione) in uno solo dei due chiusi sia che due celle adiacenti devono stare nello stesso F_s; perciò $S_i = F_1$ oppure $S_i = F_2$.
Dire che $S \in \mathscr{SA}_n$ è localmente connesso significa che per ogni $x \in S$ ogni palla di centro x contiene un intorno connesso di x in S. Una palla B è un semialgebrico e dunque lo è anche $S \cap B$, è per quanto dimostrato sopra è unione finita di componenti connesse, in particolare quella che contiene x è l'intorno che cercavamo. □

Appendice A

Note

Consideriamo un endomorfismo α di E: $y^2 = x^3 + Ax + B$ definita su un campo K

$$\alpha(x,y) = (R_1(x,y), R_2(x,y))$$

con $R_1(x,y), R_2(x,y) \in \bar{K}(E)$ perciò per $i = 1,2$

$$\begin{aligned} R_i &= \frac{d_i(x) + yc_i(x)}{a_i(x) + yb_i(x)} = \frac{d_i(x) + yc_i(x)}{a_i(x) + yb_i(x)} \frac{a_i(x) - yb_i(x)}{a_i(x) - yb_i(x)} \\ &= \frac{(d_i(x) + yc_i(x))(a_i(x) - yb_i(x))}{a_i^2(x) + y^2 b_i^2(x)} = \frac{p_i(x) + yq_i(x)}{s_i(x)} \end{aligned}$$

Visto che α è un omomorfismo $\alpha(-P) = -\alpha(P)$ ossia

$$(R_1(x,-y), R_2(x,-y)) = (R_1(x,y), -R_2(x,y))$$

quindi R_1 deve essere pari e R_2 dispari:

$$\begin{cases} R_1(x,y) = \frac{f_1}{g_1}(x) & f_1, g_1 \in \bar{K}[x] \\ R_2(x,y) = y\frac{f_2}{g_2}(x) & f_2, g_2 \in \bar{K}[x] \end{cases}$$

Inoltre se $(f_i, g_i) = 1$ supponiamo che, fissato $P = (x,y)$, sia $g_1(x) = 0$ si ha che $\alpha(P) = O$ mentre se $g_1(x) \neq 0$ anche $g_2(x) \neq 0$: consideriamo le relazioni

$$R_2^2 = y^2 \frac{f_2^2}{g_2^2}(x) = (x^3 + Ax + B)\frac{f_2^2}{g_2^2}(x) \tag{A.1}$$

$$R_2^2 = R_1^3 + AR_1 + B = \frac{u(x)}{g_1^3(x)} \tag{A.2}$$

e visto che $(u, g_1) = (f_1, g_1) = 1$ l'ultima frazione è ridotta ai minimi termini; supponiamo che $g_2(x) = 0$ allora eguagliando le equazioni sopra

$$0 = g_2(x)^2 u(x) = g_1^3(x) f_2^2(x)(x^3 + Ax + B)$$

Per ipotesi $(f_2, g_2) = 1$ quindi o $g_1^3(x) = 0$ o $x^3 + Ax + B = 0$, sempre per ipotesi la prima possibilità si esclude e non può valere la seconda perché le radici di $x^3 + Ax + B$ sono semplici (il discriminate deve essere nullo perché E sia una curva ellittica) mentre $g_2(x)^2$ ha tutte radici almeno doppie, assurdo.

Bibliografia

[AT11] M. Abate and F. Tovena. *Geometria Differenziale*. UNITEXT. Springer Milan, 2011.

[Bec81] Becker. Valuation and real places in the theory of formally real fields. *Géométrie Algébrique Réelle et Formes Quadratiques, Lecture Notes in Mathematics*, 959:1–40, 1981.

[BF03] Boneh and Franklin. Identity-based encryption from the weil pairing. *SIAM J. of Computing*, 32:586–615, 2003.

[BN93] Becker and Neuhaus. On the computation of the real radical. *Progress in Mathematics*, 109:1–20, 1993.

[BR98] Coste Bochnak and Roy. *Real Algebraic Geometry*. Springer, 1998.

[Cos00] Coste. *An introduction to semialgebraic geometry*. Dip. Mat. Univ.Pisa, Dottorato di Ricerca in Matematica, 2000.

[HP53] W. V. D. Hodge and D. Pedoe. *Methods of algebraic geometry*. Cambridge, 1953.

[Pre84] Prestel. *Lectures on Formally Real Fields*. Springer, Berlin, 1984.

[Shape] Shafarevich. *Basic algebraic geometry*. Springer, 1977pe.

[Sil08] J.H. Silverman. *The Arithmetic of Elliptic Curves*. 2008.

[Was08] L.C. Washington. *Elliptic Curves: Number Theory and Cryptography, Second Edition*. Discrete Mathematics and Its Applications. CRC Press, 2008.

[ZCS75] O. Zariski, I.S. Cohen, and P. Samuel. *Commutative Algebra I*. Graduate Texts in Mathematics. Springer New York, 1975.

[ZS76] O. Zariski and P. Samuel. *Commutative Algebra II*. Graduate Texts in Mathematics. Springer New York, 1976.