Cifrari ed equazioni alle differenze

Roberto La Scala

Università di Bari

ITASEC (Ancona, 4.2.2020)

- Molti cifrari a stream o a blocchi, largamente utilizzati in importanti applicazioni, sono regole ricorsive che determinano l'evoluzione nel tempo (clock) di un vettore (stato, registro) con entrate in un campo finito.
- Tale ricorsione è definita da una funzione vettoriale di transizione (di stato) che generalmente è quella associata ad un sistema esplicito di equazioni alle differenze (ordinarie).
- Diamo subito un paio di esempi che ci guideranno nella formalizzazione della nozione di *cifrario alle differenze*.

- Molti cifrari a stream o a blocchi, largamente utilizzati in importanti applicazioni, sono regole ricorsive che determinano l'evoluzione nel tempo (clock) di un vettore (stato, registro) con entrate in un campo finito.
- Tale ricorsione è definita da una funzione vettoriale di transizione (di stato) che generalmente è quella associata ad un sistema esplicito di equazioni alle differenze (ordinarie).
- Diamo subito un paio di esempi che ci guideranno nella formalizzazione della nozione di *cifrario alle differenze*.

- Molti cifrari a stream o a blocchi, largamente utilizzati in importanti applicazioni, sono regole ricorsive che determinano l'evoluzione nel tempo (clock) di un vettore (stato, registro) con entrate in un campo finito.
- Tale ricorsione è definita da una funzione vettoriale di transizione (di stato) che generalmente è quella associata ad un sistema esplicito di equazioni alle differenze (ordinarie).
- Diamo subito un paio di esempi che ci guideranno nella formalizzazione della nozione di *cifrario alle differenze*.

- **Trivium** è un cifrario sviluppato in Europa (Bart Preneel, Belgio) nell'ambito del progetto eSTREAM.
- È uno dei tre vincitori della call per lo sviluppo di cifrari a flusso efficienti in hardware.
- Trivium è stato pubblicato nel 2003 e nonostante la sua struttura estremamente semplice, nessuno attacco definitivo è stato portato a questo cifrario fino ad oggi.

Equazioni di stato:

$$\begin{cases} x(93) = z(45) + x(24) + z(0) + z(1)z(2), \\ y(84) = x(27) + y(6) + x(0) + x(1)x(2), \\ z(111) = z(24) + y(15) + y(0) + y(1)y(2). \end{cases}$$

- **Trivium** è un cifrario sviluppato in Europa (Bart Preneel, Belgio) nell'ambito del progetto eSTREAM.
- È uno dei tre vincitori della call per lo sviluppo di cifrari a flusso efficienti in hardware.
- Trivium è stato pubblicato nel 2003 e nonostante la sua struttura estremamente semplice, nessuno attacco definitivo è stato portato a questo cifrario fino ad oggi.

Equazioni di stato:

$$\begin{cases} x(93) = z(45) + x(24) + z(0) + z(1)z(2), \\ y(84) = x(27) + y(6) + x(0) + x(1)x(2), \\ z(111) = z(24) + y(15) + y(0) + y(1)y(2). \end{cases}$$

- **Trivium** è un cifrario sviluppato in Europa (Bart Preneel, Belgio) nell'ambito del progetto eSTREAM.
- È uno dei tre vincitori della call per lo sviluppo di cifrari a flusso efficienti in hardware.
- Trivium è stato pubblicato nel 2003 e nonostante la sua struttura estremamente semplice, nessuno attacco definitivo è stato portato a questo cifrario fino ad oggi.

Equazioni di stato:

$$\begin{cases} x(93) &= z(45) + x(24) + z(0) + z(1)z(2), \\ y(84) &= x(27) + y(6) + x(0) + x(1)x(2), \\ z(111) &= z(24) + y(15) + y(0) + y(1)y(2). \end{cases}$$

- **Trivium** è un cifrario sviluppato in Europa (Bart Preneel, Belgio) nell'ambito del progetto eSTREAM.
- È uno dei tre vincitori della call per lo sviluppo di cifrari a flusso efficienti in hardware.
- Trivium è stato pubblicato nel 2003 e nonostante la sua struttura estremamente semplice, nessuno attacco definitivo è stato portato a questo cifrario fino ad oggi.

Equazioni di stato:

$$\left\{\begin{array}{rcl} x(93) & = & z(45)+x(24)+z(0)+z(1)z(2), \\ y(84) & = & x(27)+y(6)+x(0)+x(1)x(2), \\ z(111) & = & z(24)+y(15)+y(0)+y(1)y(2). \end{array}\right.$$

- Le soluzioni di questo sistema di 3 equazioni (quadratiche) alle differenze sono terne di funzioni $x, y, z : \mathbb{N} \to \mathbb{Z}_2$.
- Essendo equazioni alle differenze di tipo esplicito, si ha una corrispondenza 1-1 fra le soluzioni e lo stato iniziale

$$v(0) = (x(0), \ldots, x(92), y(0), \ldots, x(83), z(0), \ldots, z(110))$$

- Ad un qualsiasi clock $t \geq 0$, lo stato (registro) del sistema è quindi definito come un vettore di $\mathbb{Z}_2^{288}$ $(93 + 84 + 111 = 288)$

$$v(t) = (x(t), \ldots, x(92 + t), y(t), \ldots, x(83 + t), z(t), \ldots, z(110 + t))$$

- La funzione di transizione di stato è la mappa vettoriale polinomiale $\mathbb{Z}_2^{288} \to \mathbb{Z}_2^{288}$ tale che $v(t) \mapsto v(t + 1)$, per ogni $t \geq 0$.

- Le soluzioni di questo sistema di 3 equazioni (quadratiche) alle differenze sono terne di funzioni $x, y, z : \mathbb{N} \to \mathbb{Z}_2$.
- Essendo equazioni alle differenze di tipo esplicito, si ha una corrispondenza 1-1 fra le soluzioni e lo stato iniziale

$$v(0) = (x(0), \ldots, x(92), y(0), \ldots, x(83), z(0), \ldots, z(110))$$

- Ad un qualsiasi clock $t \geq 0$, lo stato (registro) del sistema è quindi definito come un vettore di $\mathbb{Z}_2^{288}$ $(93 + 84 + 111 = 288)$

$$v(t) = (x(t), \ldots, x(92+t), y(t), \ldots, x(83+t), z(t), \ldots, z(110+t))$$

- La funzione di transizione di stato è la mappa vettoriale polinomiale $\mathbb{Z}_2^{288} \to \mathbb{Z}_2^{288}$ tale che $v(t) \mapsto v(t+1)$, per ogni $t \geq 0$.

- Le soluzioni di questo sistema di 3 equazioni (quadratiche) alle differenze sono terne di funzioni $x, y, z : \mathbb{N} \to \mathbb{Z}_2$.
- Essendo equazioni alle differenze di tipo esplicito, si ha una corrispondenza 1-1 fra le soluzioni e lo stato iniziale

$$v(0) = (x(0), \ldots, x(92), y(0), \ldots, x(83), z(0), \ldots, z(110))$$

- Ad un qualsiasi clock $t \geq 0$, lo stato (registro) del sistema è quindi definito come un vettore di $\mathbb{Z}_2^{288}$ $(93 + 84 + 111 = 288)$

$$v(t) = (x(t), \ldots, x(92 + t), y(t), \ldots, x(83 + t), z(t), \ldots, z(110 + t))$$

- La funzione di transizione di stato è la mappa vettoriale polinomiale $\mathbb{Z}_2^{288} \to \mathbb{Z}_2^{288}$ tale che $v(t) \mapsto v(t + 1)$, per ogni $t \geq 0$.

- Le soluzioni di questo sistema di 3 equazioni (quadratiche) alle differenze sono terne di funzioni $x, y, z : \mathbb{N} \to \mathbb{Z}_2$.
- Essendo equazioni alle differenze di tipo esplicito, si ha una corrispondenza 1-1 fra le soluzioni e lo stato iniziale

$$v(0) = (x(0), \ldots, x(92), y(0), \ldots, x(83), z(0), \ldots, z(110))$$

- Ad un qualsiasi clock $t \geq 0$, lo stato (registro) del sistema è quindi definito come un vettore di $\mathbb{Z}_2^{288}$ $(93 + 84 + 111 = 288)$

$$v(t) = (x(t), \ldots, x(92+t), y(t), \ldots, x(83+t), z(t), \ldots, z(110+t))$$

- La funzione di transizione di stato è la mappa vettoriale polinomiale $\mathbb{Z}_2^{288} \to \mathbb{Z}_2^{288}$ tale che $v(t) \mapsto v(t+1)$, per ogni $t \geq 0$.

- La chiave ed il IV della cifratura costituiscono $80 + 80 = 160$ bit dello stato iniziale. I restanti 128 bit dello stato iniziale sono prefissati.
- Accanto al sistema alle differenze che determina l'evoluzione dello stato in Trivium, il keystream è ottenuto mediante un polinomio lineare omogeneo

Polinomio di keystream:

$$f = x(27) + x(0) + y(15) + y(0) + z(45) + z(0)$$

- Al clock t, il keystream è definito come il valore del polinomio f calcolato sullo stato $v(t)$.
- Il keystream è fornito per i clock $t \geq T = 4 \cdot 288 = 1152$, in modo da proteggere lo stato iniziale ovvero la chiave della cifratura.

- La chiave ed il IV della cifratura costituiscono $80 + 80 = 160$ bit dello stato iniziale. I restanti 128 bit dello stato iniziale sono prefissati.
- Accanto al sistema alle differenze che determina l'evoluzione dello stato in Trivium, il keystream è ottenuto mediante un polinomio lineare omogeneo

Polinomio di keystream:

$$f \quad = \quad x(27) + x(0) + y(15) + y(0) + z(45) + z(0)$$

- Al clock t, il keystream è definito come il valore del polinomio f calcolato sullo stato $v(t)$.
- Il keystream è fornito per i clock $t \geq T = 4 \cdot 288 = 1152$, in modo da proteggere lo stato iniziale ovvero la chiave della cifratura.

- La chiave ed il IV della cifratura costituiscono $80 + 80 = 160$ bit dello stato iniziale. I restanti 128 bit dello stato iniziale sono prefissati.
- Accanto al sistema alle differenze che determina l'evoluzione dello stato in Trivium, il keystream è ottenuto mediante un polinomio lineare omogeneo

Polinomio di keystream:

$$f = x(27) + x(0) + y(15) + y(0) + z(45) + z(0)$$

- Al clock t, il keystream è definito come il valore del polinomio f calcolato sullo stato $v(t)$.
- Il keystream è fornito per i clock $t \geq T = 4 \cdot 288 = 1152$, in modo da proteggere lo stato iniziale ovvero la chiave della cifratura.

- La chiave ed il IV della cifratura costituiscono $80 + 80 = 160$ bit dello stato iniziale. I restanti 128 bit dello stato iniziale sono prefissati.
- Accanto al sistema alle differenze che determina l'evoluzione dello stato in Trivium, il keystream è ottenuto mediante un polinomio lineare omogeneo

Polinomio di keystream:

$$f = x(27) + x(0) + y(15) + y(0) + z(45) + z(0)$$

- Al clock t, il keystream è definito come il valore del polinomio f calcolato sullo stato $v(t)$.
- Il keystream è fornito per i clock $t \geq T = 4 \cdot 288 = 1152$, in modo da proteggere lo stato iniziale ovvero la chiave della cifratura.

- Una variante di Trivium molto studiata in crittoanalisi è **Bivium**

Equazioni di stato:

$$\begin{cases} x(93) = x(24) + y(15) + y(0) + y(1)y(2), \\ y(84) = x(27) + y(6) + x(0) + x(1)x(2). \end{cases}$$

Polinomio di keystream:

$$f = x(27) + x(0) + y(15) + y(0).$$

- Lo stato è quindi un vettore di $\mathbb{Z}_2^{177}$ $(93 + 84 = 177)$ ed il keystream va in output dopo $T = 4 \cdot 177 = 708$ clock.

- Una variante di Trivium molto studiata in crittoanalisi è **Bivium**

Equazioni di stato:

$$\begin{cases} x(93) = x(24) + y(15) + y(0) + y(1)y(2), \\ y(84) = x(27) + y(6) + x(0) + x(1)x(2). \end{cases}$$

Polinomio di keystream:

$$f = x(27) + x(0) + y(15) + y(0).$$

- Lo stato è quindi un vettore di $\mathbb{Z}_2^{177}$ $(93 + 84 = 177)$ ed il keystream va in output dopo $T = 4 \cdot 177 = 708$ clock.

- Una variante di Trivium molto studiata in crittoanalisi è **Bivium**

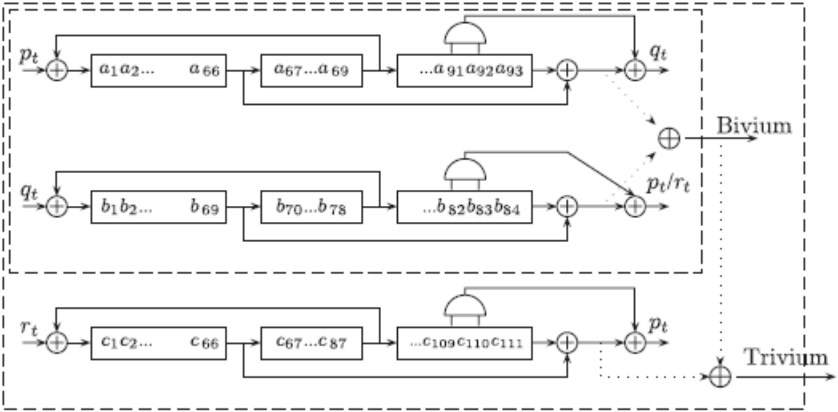
Equazioni di stato:

$$\begin{cases} x(93) & = & x(24) + y(15) + y(0) + y(1)y(2), \\ y(84) & = & x(27) + y(6) + x(0) + x(1)x(2). \end{cases}$$

Polinomio di keystream:

$$f \quad = \quad x(27) + x(0) + y(15) + y(0).$$

- Lo stato è quindi un vettore di $\mathbb{Z}_2^{177}$ $(93 + 84 = 177)$ ed il keystream va in output dopo $T = 4 \cdot 177 = 708$ clock.


p_t
$a_1 a_2 \ldots \quad a_{66}$
$a_{67} \ldots a_{69}$
$\ldots a_{91} a_{92} a_{93}$
q_t
Bivium
q_t
$b_1 b_2 \ldots \quad b_{69}$
$b_{70} \ldots b_{78}$
$\ldots b_{82} b_{83} b_{84}$
p_t/r_t
r_t
$c_1 c_2 \ldots \quad c_{66}$
$c_{67} \ldots c_{87}$
$\ldots c_{109} c_{110} c_{111}$
p_t
Trivium

- Osserviamo che le funzioni di transizione di stato di Bivium e Trivium sono invertibili ovvero è possibile calcolare lo stato iniziale (chiave) a partire da un qualsiasi stato.
- Questa è una potenziale vulnerabilità in quanto è possibile attaccare lo stato al clock T corrispondente all'inizio del keystream.
- D'altra parte, l'invertibilità (e la riducibilità) di un sistema alle differenze può essere considerata una risorsa per costruire un cifrario simmetrico.

- Osserviamo che le funzioni di transizione di stato di Bivium e Trivium sono invertibili ovvero è possibile calcolare lo stato iniziale (chiave) a partire da un qualsiasi stato.
- Questa è una potenziale vulnerabilità in quanto è possibile attaccare lo stato al clock T corrispondente all'inizio del keystream.
- D'altra parte, l'invertibilità (e la riducibilità) di un sistema alle differenze può essere considerata una risorsa per costruire un cifrario simmetrico.

- Osserviamo che le funzioni di transizione di stato di Bivium e Trivium sono invertibili ovvero è possibile calcolare lo stato iniziale (chiave) a partire da un qualsiasi stato.
- Questa è una potenziale vulnerabilità in quanto è possibile attaccare lo stato al clock T corrispondente all'inizio del keystream.
- D'altra parte, l'invertibilità (e la riducibilità) di un sistema alle differenze può essere considerata una risorsa per costruire un cifrario simmetrico.

- Un sistema invertibile contenente un sottosistema definito su un sottoinsieme di variabili definisce un cifrario simmetrico.
- Lo stato iniziale $u(0)$ del sottosistema è la chiave della cifratura. Lo stato iniziale del sistema $(u(0), v(0))$ è costituito dalla chiave e dal plaintext $v(0)$.
- Se $(u(T), v(T))$ è lo stato finale del sistema ad un clock T prefissato, il ciphertext è costituito dal vettore $v(T)$.
- Per decifrare, utilizzando il sottosistema si può calcolare $u(T)$ a partire della chiave $u(0)$. Il sistema inverso è poi capace di calcolare lo stato iniziale $(u(0), v(0))$, e quindi il plaintext $v(0)$, a partire dallo stato finale $(u(T), v(T))$.

- Un sistema invertibile contenente un sottosistema definito su un sottoinsieme di variabili definisce un cifrario simmetrico.
- Lo stato iniziale $u(0)$ del sottosistema è la chiave della cifratura. Lo stato iniziale del sistema $(u(0), v(0))$ è costituito dalla chiave e dal plaintext $v(0)$.
- Se $(u(T), v(T))$ è lo stato finale del sistema ad un clock T prefissato, il ciphertext è costituito dal vettore $v(T)$.
- Per decifrare, utilizzando il sottosistema si può calcolare $u(T)$ a partire della chiave $u(0)$. Il sistema inverso è poi capace di calcolare lo stato iniziale $(u(0), v(0))$, e quindi il plaintext $v(0)$, a partire dallo stato finale $(u(T), v(T))$.

- Un sistema invertibile contenente un sottosistema definito su un sottoinsieme di variabili definisce un cifrario simmetrico.
- Lo stato iniziale $u(0)$ del sottosistema è la chiave della cifratura. Lo stato iniziale del sistema $(u(0), v(0))$ è costituito dalla chiave e dal plaintext $v(0)$.
- Se $(u(T), v(T))$ è lo stato finale del sistema ad un clock T prefissato, il ciphertext è costituito dal vettore $v(T)$.
- Per decifrare, utilizzando il sottosistema si può calcolare $u(T)$ a partire della chiave $u(0)$. Il sistema inverso è poi capace di calcolare lo stato iniziale $(u(0), v(0))$, e quindi il plaintext $v(0)$, a partire dallo stato finale $(u(T), v(T))$.

- Un sistema invertibile contenente un sottosistema definito su un sottoinsieme di variabili definisce un cifrario simmetrico.
- Lo stato iniziale $u(0)$ del sottosistema è la chiave della cifratura. Lo stato iniziale del sistema $(u(0), v(0))$ è costituito dalla chiave e dal plaintext $v(0)$.
- Se $(u(T), v(T))$ è lo stato finale del sistema ad un clock T prefissato, il ciphertext è costituito dal vettore $v(T)$.
- Per decifrare, utilizzando il sottosistema si può calcolare $u(T)$ a partire della chiave $u(0)$. Il sistema inverso è poi capace di calcolare lo stato iniziale $(u(0), v(0))$, e quindi il plaintext $v(0)$, a partire dallo stato finale $(u(T), v(T))$.

- Un sistema invertibile contenente un sottosistema definito su un sottoinsieme di variabili definisce un cifrario simmetrico.
- Lo stato iniziale $u(0)$ del sottosistema è la chiave della cifratura. Lo stato iniziale del sistema $(u(0), v(0))$ è costituito dalla chiave e dal plaintext $v(0)$.
- Se $(u(T), v(T))$ è lo stato finale del sistema ad un clock T prefissato, il ciphertext è costituito dal vettore $v(T)$.
- Per decifrare, utilizzando il sottosistema si può calcolare $u(T)$ a partire della chiave $u(0)$. Il sistema inverso è poi capace di calcolare lo stato iniziale $(u(0), v(0))$, e quindi il plaintext $v(0)$, a partire dallo stato finale $(u(T), v(T))$.

- Un esempio fondamentale di un tale cifrario è **Keeloq**, utilizzato nei transponder RFID per immobilizzatori auto, aperture porte, etc.
- Venduto il brevetto nel 1995 per 10^7 USD, questo cifrario è stato crittoanalizzato in modo critico nel 2008.

Keeloq:

$$\begin{cases} k(64) & = & k(0), \\ x(32) & = & k(0) + x(0) + x(16) + x(9) + x(1) + x(31)x(20) \\ & & + x(31)x(1) + x(26)x(20) + x(26)x(1) + x(20)x(9) \\ & & + x(9)x(1) + x(31)x(9)x(1) + x(31)x(20)x(1) \\ & & + x(31)x(26)x(9) + x(31)x(26)x(20). \end{cases}$$

- Un esempio fondamentale di un tale cifrario è **Keeloq**, utilizzato nei transponder RFID per immobilizzatori auto, aperture porte, etc.
- Venduto il brevetto nel 1995 per 10^7 USD, questo cifrario è stato crittoanalizzato in modo critico nel 2008.

Keeloq:

$$\begin{cases} k(64) &= k(0), \\ x(32) &= k(0) + x(0) + x(16) + x(9) + x(1) + x(31)x(20) \\ & + x(31)x(1) + x(26)x(20) + x(26)x(1) + x(20)x(9) \\ & + x(9)x(1) + x(31)x(9)x(1) + x(31)x(20)x(1) \\ & + x(31)x(26)x(9) + x(31)x(26)x(20). \end{cases}$$

- Un esempio fondamentale di un tale cifrario è **Keeloq**, utilizzato nei transponder RFID per immobilizzatori auto, aperture porte, etc.
- Venduto il brevetto nel 1995 per 10^7 USD, questo cifrario è stato crittoanalizzato in modo critico nel 2008.

Keeloq:

$$\begin{cases} k(64) &= k(0), \\ x(32) &= k(0) + x(0) + x(16) + x(9) + x(1) + x(31)x(20) \\ &\quad + x(31)x(1) + x(26)x(20) + x(26)x(1) + x(20)x(9) \\ &\quad + x(9)x(1) + x(31)x(9)x(1) + x(31)x(20)x(1) \\ &\quad + x(31)x(26)x(9) + x(31)x(26)x(20). \end{cases}$$

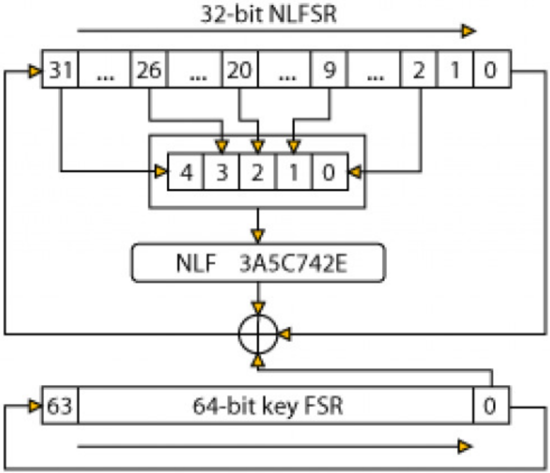
- Il sottosistema di chiave è costituito dalla singola equazione $k(64) = k(0)$.
- Lo stato del sistema $(u(t), v(t))$ è un vettore di $\mathbb{Z}_2^{96}$ $(64 + 32 = 96)$. Lo stato del sottosistema di chiave è il vettore $u(t) \in \mathbb{Z}_2^{64}$.
- Il clock finale del cifrario è $T = 8 \cdot 64 + 16 = 528$.
- Dunque, la chiave è $u(0) = (k(0), \ldots, k(63))$, il plaintext è $v(0) = (x(0), \ldots, x(31))$ ed il ciphertext è $v(528)$.
- Una importante vulnerabilità di Keeloq è costituita dal breve periodo 64 (della funzione di transizione di stato) del sottosistema. Questo permette un efficace attacco di tipo algebrico (Bard).

- Il sottosistema di chiave è costituito dalla singola equazione $k(64) = k(0)$.
- Lo stato del sistema $(u(t), v(t))$ è un vettore di $\mathbb{Z}_2^{96}$ $(64 + 32 = 96)$. Lo stato del sottosistema di chiave è il vettore $u(t) \in \mathbb{Z}_2^{64}$.
- Il clock finale del cifrario è $T = 8 \cdot 64 + 16 = 528$.
- Dunque, la chiave è $u(0) = (k(0), \ldots, k(63))$, il plaintext è $v(0) = (x(0), \ldots, x(31))$ ed il ciphertext è $v(528)$.
- Una importante vulnerabilità di Keeloq è costituita dal breve periodo 64 (della funzione di transizione di stato) del sottosistema. Questo permette un efficace attacco di tipo algebrico (Bard).

- Il sottosistema di chiave è costituito dalla singola equazione $k(64) = k(0)$.
- Lo stato del sistema $(u(t), v(t))$ è un vettore di $\mathbb{Z}_2^{96}$ $(64 + 32 = 96)$. Lo stato del sottosistema di chiave è il vettore $u(t) \in \mathbb{Z}_2^{64}$.
- Il clock finale del cifrario è $T = 8 \cdot 64 + 16 = 528$.
- Dunque, la chiave è $u(0) = (k(0), \ldots, k(63))$, il plaintext è $v(0) = (x(0), \ldots, x(31))$ ed il ciphertext è $v(528)$.
- Una importante vulnerabilità di Keeloq è costituita dal breve periodo 64 (della funzione di transizione di stato) del sottosistema. Questo permette un efficace attacco di tipo algebrico (Bard).

- Il sottosistema di chiave è costituito dalla singola equazione $k(64) = k(0)$.
- Lo stato del sistema $(u(t), v(t))$ è un vettore di $\mathbb{Z}_2^{96}$ $(64 + 32 = 96)$. Lo stato del sottosistema di chiave è il vettore $u(t) \in \mathbb{Z}_2^{64}$.
- Il clock finale del cifrario è $T = 8 \cdot 64 + 16 = 528$.
- Dunque, la chiave è $u(0) = (k(0), \dots, k(63))$, il plaintext è $v(0) = (x(0), \dots, x(31))$ ed il ciphertext è $v(528)$.
- Una importante vulnerabilità di Keeloq è costituita dal breve periodo 64 (della funzione di transizione di stato) del sottosistema. Questo permette un efficace attacco di tipo algebrico (Bard).

- Il sottosistema di chiave è costituito dalla singola equazione $k(64) = k(0)$.
- Lo stato del sistema $(u(t), v(t))$ è un vettore di $\mathbb{Z}_2^{96}$ $(64 + 32 = 96)$. Lo stato del sottosistema di chiave è il vettore $u(t) \in \mathbb{Z}_2^{64}$.
- Il clock finale del cifrario è $T = 8 \cdot 64 + 16 = 528$.
- Dunque, la chiave è $u(0) = (k(0), \ldots, k(63))$, il plaintext è $v(0) = (x(0), \ldots, x(31))$ ed il ciphertext è $v(528)$.
- Una importante vulnerabilità di Keeloq è costituita dal breve periodo 64 (della funzione di transizione di stato) del sottosistema. Questo permette un efficace attacco di tipo algebrico (Bard).


32-bit NLFSR
31
...
26
...
20
...
9
...
2
1
0
4
3
2
1
0
NLF 3A5C742E
63
64-bit key FSR
0

Formalizzazione

- Sia $\mathbb{K}$ un campo qualsiasi. Sia $X(t) = \{x_1(t), \ldots, x_n(t)\}$ un insieme di variabili, per ogni intero (clock) $t \geq 0$. Consideriamo l'algebra dei polinomi $R = \mathbb{K}[X]$ sull'insieme infinito di variabili $X = \bigcup_{t \geq 0} X(t)$.
- Consideriamo l'endomorfismo di algebre $\sigma : R \to R$ tale che $x_i(t) \mapsto x_i(t+1)$. Chiamiamo σ *l'operatore di shift di* R.
- L'algebra R, dotata dell'endomorfismo σ, si chiama *l'algebra dei polinomi alle differenze (ordinarie) nelle variabili* $x_1, \ldots, x_n$.
- Consideriamo ora gli interi $r_1, \ldots, r_n \geq 0$. Definiamo il sottinsieme

$$X_{r_1,\ldots,r_n} = \{x_1(0), \ldots, x_1(r_1 - 1), \ldots, x_n(0), \ldots, x_n(r_n - 1)\} \subset X$$

e la sottoalgebra $R_{r_1,\ldots,r_n} = \mathbb{K}[X_{r_1,\ldots,r_n}] \subset R$.

Formalizzazione

- Sia $\mathbb{K}$ un campo qualsiasi. Sia $X(t) = \{x_1(t), \ldots, x_n(t)\}$ un insieme di variabili, per ogni intero (clock) $t \geq 0$. Consideriamo l'algebra dei polinomi $R = \mathbb{K}[X]$ sull'insieme infinito di variabili $X = \bigcup_{t \geq 0} X(t)$.
- Consideriamo l'endomorfismo di algebre $\sigma : R \to R$ tale che $x_i(t) \mapsto x_i(t+1)$. Chiamiamo σ *l'operatore di shift di* R.
- L'algebra R, dotata dell'endomorfismo σ, si chiama *l'algebra dei polinomi alle differenze (ordinarie) nelle variabili* $x_1, \ldots, x_n$.
- Consideriamo ora gli interi $r_1, \ldots, r_n \geq 0$. Definiamo il sottinsieme

$$X_{r_1,\ldots,r_n} = \{x_1(0), \ldots, x_1(r_1 - 1), \ldots, x_n(0), \ldots, x_n(r_n - 1)\} \subset X$$

e la sottoalgebra $R_{r_1,\ldots,r_n} = \mathbb{K}[X_{r_1,\ldots,r_n}] \subset R$.

Formalizzazione

- Sia $\mathbb{K}$ un campo qualsiasi. Sia $X(t) = \{x_1(t), \ldots, x_n(t)\}$ un insieme di variabili, per ogni intero (clock) $t \geq 0$. Consideriamo l'algebra dei polinomi $R = \mathbb{K}[X]$ sull'insieme infinito di variabili $X = \bigcup_{t \geq 0} X(t)$.
- Consideriamo l'endomorfismo di algebre $\sigma : R \to R$ tale che $x_i(t) \mapsto x_i(t+1)$. Chiamiamo σ *l'operatore di shift di R*.
- L'algebra R, dotata dell'endomorfismo σ, si chiama *l'algebra dei polinomi alle differenze (ordinarie) nelle variabili $x_1, \ldots, x_n$*.
- Consideriamo ora gli interi $r_1, \ldots, r_n \geq 0$. Definiamo il sottinsieme

$$X_{r_1,\ldots,r_n} = \{x_1(0), \ldots, x_1(r_1 - 1), \ldots, x_n(0), \ldots, x_n(r_n - 1)\} \subset X$$

e la sottoalgebra $R_{r_1,\ldots,r_n} = \mathbb{K}[X_{r_1,\ldots,r_n}] \subset R$.

Formalizzazione

- Sia $\mathbb{K}$ un campo qualsiasi. Sia $X(t) = \{x_1(t), \ldots, x_n(t)\}$ un insieme di variabili, per ogni intero (clock) $t \geq 0$. Consideriamo l'algebra dei polinomi $R = \mathbb{K}[X]$ sull'insieme infinito di variabili $X = \bigcup_{t\geq 0} X(t)$.
- Consideriamo l'endomorfismo di algebre $\sigma : R \to R$ tale che $x_i(t) \mapsto x_i(t+1)$. Chiamiamo σ *l'operatore di shift di* R.
- L'algebra R, dotata dell'endomorfismo σ, si chiama *l'algebra dei polinomi alle differenze (ordinarie) nelle variabili* $x_1, \ldots, x_n$.
- Consideriamo ora gli interi $r_1, \ldots, r_n \geq 0$. Definiamo il sottinsieme

$$X_{r_1,\ldots,r_n} = \{x_1(0), \ldots, x_1(r_1 - 1), \ldots, x_n(0), \ldots, x_n(r_n - 1)\} \subset X$$

e la sottoalgebra $R_{r_1,\ldots,r_n} = \mathbb{K}[X_{r_1,\ldots,r_n}] \subset R$.

Definizione

Siano $f_1, \ldots, f_n$ polinomi di $R_{r_1,\ldots,r_n}$. Un sistema di equazioni esplicite alle differenze (ordinarie) *è un sistema infinito di equazioni algebriche del tipo*

$$\left\{\begin{array}{rcl} x_1(r_1+t) & = & \sigma^t(f_1), \\ & \vdots & \\ x_n(r_n+t) & = & \sigma^t(f_n). \end{array}\right. \qquad (t \geq 0)$$

Tale sistema si denota brevemente come

$$\left\{\begin{array}{rcl} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n. \end{array}\right. \tag{1}$$

Una $\mathbb{K}$-soluzione del sistema (1) *è quindi una n-upla* $(a_1, \ldots, a_n)$ *di funzioni (successioni)* $a_i : \mathbb{N} \to \mathbb{K}$. *In particolare, se i polinomi f_i sono tutti lineari omogenei abbiamo un sistema di LFSR.*

Definizione

Siano $f_1, \dots, f_n$ *polinomi di* $R_{r_1,\dots,r_n}$. *Un* sistema di equazioni esplicite alle differenze (ordinarie) *è un sistema infinito di equazioni algebriche del tipo*

$$\begin{cases} x_1(r_1 + t) & = & \sigma^t(f_1), \\ & \vdots & \\ x_n(r_n + t) & = & \sigma^t(f_n). \end{cases} \qquad (t \geq 0)$$

Tale sistema si denota brevemente come

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n. \end{cases} \tag{1}$$

Una $\mathbb{K}$-soluzione del sistema (1) *è quindi una n-upla* $(a_1, \dots, a_n)$ *di funzioni (successioni)* $a_i : \mathbb{N} \to \mathbb{K}$. *In particolare, se i polinomi* f_i *sono tutti lineari omogenei abbiamo un sistema di LFSR.*

Definizione

Siano $f_1, \ldots, f_n$ *polinomi di* $R_{r_1,\ldots,r_n}$. *Un* sistema di equazioni esplicite alle differenze (ordinarie) *è un sistema infinito di equazioni algebriche del tipo*

$$\begin{cases} x_1(r_1 + t) & = & \sigma^t(f_1), \\ & \vdots & \\ x_n(r_n + t) & = & \sigma^t(f_n). \end{cases} \qquad (t \geq 0)$$

Tale sistema si denota brevemente come

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n. \end{cases} \tag{1}$$

Una $\mathbb{K}$-soluzione del sistema (1) *è quindi una n-upla* $(a_1, \ldots, a_n)$ *di funzioni (successioni)* $a_i : \mathbb{N} \to \mathbb{K}$. *In particolare, se i polinomi* f_i *sono tutti lineari omogenei abbiamo un sistema di LFSR.*

Definizione

Siano $f_1, \ldots, f_n$ *polinomi di* $R_{r_1,\ldots,r_n}$. *Un* sistema di equazioni esplicite alle differenze (ordinarie) *è un sistema infinito di equazioni algebriche del tipo*

$$\begin{cases} x_1(r_1 + t) & = & \sigma^t(f_1), \\ & \vdots & \\ x_n(r_n + t) & = & \sigma^t(f_n). \end{cases} \qquad (t \geq 0)$$

Tale sistema si denota brevemente come

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n. \end{cases} \tag{1}$$

Una $\mathbb{K}$-soluzione del sistema (1) *è quindi una n-upla* $(a_1, \ldots, a_n)$ *di funzioni (successioni)* $a_i : \mathbb{N} \to \mathbb{K}$. *In particolare, se i polinomi* f_i *sono tutti lineari omogenei abbiamo un sistema di LFSR.*

Definizione

Per ogni sistema (1)*, definiamo il corrispondente endomorfismo di algebre* $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *tale che* $(1 \leq i \leq n)$

$$x_i(0) \mapsto x_i(1), \ldots, x_i(r_i - 2) \mapsto x_i(r_i - 1), x_i(r_i - 1) \mapsto f_i.$$

Se $r = r_1 + \ldots + r_n$*, denotiamo* $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ *la funzione vettoriale polinomiale corrispondente a* T*. Allora, per ogni* $t \geq 0$ *abbiamo* $\hat{\mathrm{T}} : v(t) \mapsto v(t+1)$ *dove*

$$v(t) = (a_1(t), \ldots, a_1(t + r_1 - 1), \ldots, a_n(t), \ldots, a_n(t + r_n - 1)) \in \mathbb{K}^r$$

è lo stato al clock *t di una qualsiasi* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$*. Chiamiamo* T l'endomorfismo *e* $\hat{\mathrm{T}}$ *la* funzione di transizione di stato del sistema (1).

Definizione

Per ogni sistema (1)*, definiamo il corrispondente endomorfismo di algebre* $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *tale che* $(1 \leq i \leq n)$

$$x_i(0) \mapsto x_i(1), \ldots, x_i(r_i - 2) \mapsto x_i(r_i - 1), x_i(r_i - 1) \mapsto f_i.$$

Se $r = r_1 + \ldots + r_n$*, denotiamo* $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ *la funzione vettoriale polinomiale corrispondente a* T*. Allora, per ogni* $t \geq 0$ *abbiamo* $\hat{\mathrm{T}} : v(t) \mapsto v(t+1)$ *dove*

$$v(t) = (a_1(t), \ldots, a_1(t + r_1 - 1), \ldots, a_n(t), \ldots, a_n(t + r_n - 1)) \in \mathbb{K}^r$$

è lo stato al clock *t di una qualsiasi* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$*. Chiamiamo* T l'endomorfismo *e* $\hat{\mathrm{T}}$ *la* funzione di transizione di stato del sistema (1).

Definizione

Per ogni sistema (1)*, definiamo il corrispondente endomorfismo di algebre* $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *tale che* $(1 \leq i \leq n)$

$$x_i(0) \mapsto x_i(1), \ldots, x_i(r_i - 2) \mapsto x_i(r_i - 1), x_i(r_i - 1) \mapsto f_i.$$

Se $r = r_1 + \ldots + r_n$*, denotiamo* $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ *la funzione vettoriale polinomiale corrispondente a* T*. Allora, per ogni* $t \geq 0$ *abbiamo* $\hat{\mathrm{T}} : v(t) \mapsto v(t+1)$ *dove*

$$v(t) = (a_1(t), \ldots, a_1(t + r_1 - 1), \ldots, a_n(t), \ldots, a_n(t + r_n - 1)) \in \mathbb{K}^r$$

è lo stato al clock t *di una qualsiasi* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$*.*
Chiamiamo T l'endomorfismo *e* $\hat{\mathrm{T}}$ *la* funzione di transizione di stato del sistema (1).

Definizione

Per ogni sistema (1), *definiamo il corrispondente endomorfismo di algebre* $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *tale che* $(1 \leq i \leq n)$

$$x_i(0) \mapsto x_i(1), \ldots, x_i(r_i - 2) \mapsto x_i(r_i - 1), x_i(r_i - 1) \mapsto f_i.$$

Se $r = r_1 + \ldots + r_n$, *denotiamo* $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ *la funzione vettoriale polinomiale corrispondente a* T. *Allora, per ogni* $t \geq 0$ *abbiamo* $\hat{\mathrm{T}} : v(t) \mapsto v(t+1)$ *dove*

$$v(t) = (a_1(t), \ldots, a_1(t + r_1 - 1), \ldots, a_n(t), \ldots, a_n(t + r_n - 1)) \in \mathbb{K}^r$$

è lo stato al clock t *di una qualsiasi* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$. *Chiamiamo* T l'endomorfismo *e* $\hat{\mathrm{T}}$ *la* funzione di transizione di stato del sistema (1).

Proposizione

Denotiamo con $V_{\mathbb{K}}$ l'insieme di tutte le $\mathbb{K}$-soluzioni del sistema (1). *È data una mappa bigettiva $\iota : V_{\mathbb{K}} \to \mathbb{K}^r$ tale che*

$$(a_1, \ldots, a_n) \mapsto (a_1(0), \ldots, a_1(r_1 - 1), \ldots, a_n(0), \ldots, a_n(r_n - 1)).$$

In altri termini, il sistema (1) *ammette un unica soluzione un volta fissato il suo stato iniziale. Inoltre, le funzioni ι, ι^{-1} sono entrambe polinomiali.*

Definizione

Consideriamo l'endomorfismo $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ e la corrispondente funzione di transizione di stato $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$. Diremo che il sistema (1) *è* invertibile *se T è un automorfismo. In tal caso, anche $\hat{\mathrm{T}}$ è una funzione bigettiva.*

Proposizione

Denotiamo con $V_{\mathbb{K}}$ l'insieme di tutte le $\mathbb{K}$-soluzioni del sistema (1). È data una mappa bigettiva $\iota : V_{\mathbb{K}} \to \mathbb{K}^r$ tale che

$$(a_1, \ldots, a_n) \mapsto (a_1(0), \ldots, a_1(r_1 - 1), \ldots, a_n(0), \ldots, a_n(r_n - 1)).$$

In altri termini, il sistema (1) ammette un unica soluzione un volta fissato il suo stato iniziale. Inoltre, le funzioni ι, ι^{-1} sono entrambe polinomiali.

Definizione

Consideriamo l'endomorfismo $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ e la corrispondente funzione di transizione di stato $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$. Diremo che il sistema (1) è invertibile *se T è un automorfismo. In tal caso, anche $\hat{\mathrm{T}}$ è una funzione bigettiva.*

Proposizione

Denotiamo con $V_{\mathbb{K}}$ *l'insieme di tutte le* $\mathbb{K}$*-soluzioni del sistema* (1). *È data una mappa bigettiva* $\iota : V_{\mathbb{K}} \to \mathbb{K}^r$ *tale che*

$$(a_1, \ldots, a_n) \mapsto (a_1(0), \ldots, a_1(r_1 - 1), \ldots, a_n(0), \ldots, a_n(r_n - 1)).$$

In altri termini, il sistema (1) *ammette un unica soluzione un volta fissato il suo stato iniziale. Inoltre, le funzioni* ι, ι^{-1} *sono entrambe polinomiali.*

Definizione

Consideriamo l'endomorfismo $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *e la corrispondente funzione di transizione di stato* $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$. *Diremo che il sistema* (1) *è* invertibile *se* T *è un automorfismo. In tal caso, anche* $\hat{\mathrm{T}}$ *è una funzione bigettiva.*

Proposizione

Denotiamo con $V_{\mathbb{K}}$ l'insieme di tutte le $\mathbb{K}$-soluzioni del sistema (1). *È data una mappa bigettiva $\iota : V_{\mathbb{K}} \to \mathbb{K}^r$ tale che*

$$(a_1, \ldots, a_n) \mapsto (a_1(0), \ldots, a_1(r_1 - 1), \ldots, a_n(0), \ldots, a_n(r_n - 1)).$$

In altri termini, il sistema (1) *ammette un unica soluzione un volta fissato il suo stato iniziale. Inoltre, le funzioni ι, ι^{-1} sono entrambe polinomiali.*

Definizione

Consideriamo l'endomorfismo $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ e la corrispondente funzione di transizione di stato $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$. Diremo che il sistema (1) *è* invertibile *se T è un automorfismo. In tal caso, anche $\hat{\mathrm{T}}$ è una funzione bigettiva.*

Proposizione

Denotiamo con $V_{\mathbb{K}}$ *l'insieme di tutte le* $\mathbb{K}$*-soluzioni del sistema* (1)*. È data una mappa bigettiva* $\iota : V_{\mathbb{K}} \to \mathbb{K}^r$ *tale che*

$$(a_1, \ldots, a_n) \mapsto (a_1(0), \ldots, a_1(r_1 - 1), \ldots, a_n(0), \ldots, a_n(r_n - 1)).$$

In altri termini, il sistema (1) *ammette un unica soluzione un volta fissato il suo stato iniziale. Inoltre, le funzioni* ι, ι^{-1} *sono entrambe polinomiali.*

Definizione

Consideriamo l'endomorfismo $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *e la corrispondente funzione di transizione di stato* $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$*. Diremo che il sistema* (1) *è* invertibile *se* T *è un automorfismo. In tal caso, anche* $\hat{\mathrm{T}}$ *è una funzione bigettiva.*

Le basi di Gröbner forniscono un metodo algoritmico per stabilire se un endomorfismo è invertibile e per calcolarne l'inverso.

Teorema

Siano $X = \{x_1, \ldots, x_r\}, X' = \{x'_1, \ldots, x'_r\}$ due insiemi di variabili e definiamo le algebra polinomiali $P = \mathbb{K}[X], P' = \mathbb{K}[X']$ e $Q = \mathbb{K}[X \cup X'] = P \otimes P'$. Consideriamo un endomorfismo $\varphi : P \to P$ tale che $x_1 \mapsto g_1, \ldots, x_r \mapsto g_r$ $(g_i \in P)$ ed il corrispondente ideale $J \subset Q$ generato dall'insieme $\{x'_1 - g_1, \ldots, x'_r - g_r\}$. Dotiamo inoltre Q di un ordinamento monomiale prodotto tale che $X \succ X'$. Allora, φ è un automorfismo di P se e solo se la base di Gröbner ridotta di J è del tipo $\{x_1 - g'_1, \ldots, x_r - g'_r\}$ dove $g'_i \in P'$. In questo caso, a meno di identificare (isomorfismo) P, P', abbiamo che $\varphi^{-1} : P \to P$ è l'endomorfismo tale che $x_1 \mapsto g'_1, \ldots, x_r \mapsto g'_r$.

Le basi di Gröbner forniscono un metodo algoritmico per stabilire se un endomorfismo è invertibile e per calcolarne l'inverso.

Teorema

Siano $X = \{x_1, \ldots, x_r\}, X' = \{x'_1, \ldots, x'_r\}$ *due insiemi di variabili e definiamo le algebra polinomiali* $P = \mathbb{K}[X], P' = \mathbb{K}[X']$ *e* $Q = \mathbb{K}[X \cup X'] = P \otimes P'$*. Consideriamo un endomorfismo* $\varphi : P \to P$ *tale che* $x_1 \mapsto g_1, \ldots, x_r \mapsto g_r$ $(g_i \in P)$ *ed il corrispondente ideale* $J \subset Q$ *generato dall'insieme* $\{x'_1 - g_1, \ldots, x'_r - g_r\}$*. Dotiamo inoltre* Q *di un ordinamento monomiale prodotto tale che* $X \succ X'$*. Allora,* φ *è un automorfismo di* P *se e solo se la base di Gröbner ridotta di* J *è del tipo* $\{x_1 - g'_1, \ldots, x_r - g'_r\}$ *dove* $g'_i \in P'$*. In questo caso, a meno di identificare (isomorfismo)* P, P'*, abbiamo che* $\varphi^{-1} : P \to P$ *è l'endomorfismo tale che* $x_1 \mapsto g'_1, \ldots, x_r \mapsto g'_r$*.*

Le basi di Gröbner forniscono un metodo algoritmico per stabilire se un endomorfismo è invertibile e per calcolarne l'inverso.

Teorema

Siano $X = \{x_1, \ldots, x_r\}, X' = \{x'_1, \ldots, x'_r\}$ *due insiemi di variabili e definiamo le algebra polinomiali* $P = \mathbb{K}[X], P' = \mathbb{K}[X']$ *e* $Q = \mathbb{K}[X \cup X'] = P \otimes P'$. *Consideriamo un endomorfismo* $\varphi : P \to P$ *tale che* $x_1 \mapsto g_1, \ldots, x_r \mapsto g_r$ $(g_i \in P)$ *ed il corrispondente ideale* $J \subset Q$ *generato dall'insieme* $\{x'_1 - g_1, \ldots, x'_r - g_r\}$. *Dotiamo inoltre* Q *di un ordinamento monomiale prodotto tale che* $X \succ X'$. *Allora,* φ *è un automorfismo di* P *se e solo se la base di Gröbner ridotta di* J *è del tipo* $\{x_1 - g'_1, \ldots, x_r - g'_r\}$ *dove* $g'_i \in P'$. *In questo caso, a meno di identificare (isomorfismo)* P, P', *abbiamo che* $\varphi^{-1} : P \to P$ *è l'endomorfismo tale che* $x_1 \mapsto g'_1, \ldots, x_r \mapsto g'_r$.

In particolare, per i sistemi espliciti alle differenze abbiamo

Proposizione

Sia $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *l'automorfismo di transizione di stato correspondente al sistema invertibile (1), precisamente*

$$x_i(0) \mapsto x_i(1), \ldots, x_i(r_i - 2) \mapsto x_i(r_i - 1), x_i(r_i - 1) \mapsto f_i.$$

Poniamo $R'_{r_1,\ldots,r_n} = \mathbb{K}[X'_{r_1,\ldots,r_n}]$ *e sia* $Q_{r_1,\ldots,r_n} = R_{r_1,\ldots,r_n} \otimes R'_{r_1,\ldots,r_n}$. *Consideriamo l'ideale* $J \subset Q_{r_1,\ldots,r_n}$ *generato dai polinomi*

$$x'_i(0) - x_i(1), \ldots, x'_i(r_i - 2) - x_i(r_i - 1), x'_i(r_i - 1) - f_i.$$

Allora, la base di Gröbner ridotta di J è del tipo

$$x_i(1) - x'_i(0), \ldots, x_i(r_i - 1) - x'_i(r_i - 2), x_i(0) - f'_i.$$

dove $f'_i \in R'_{r_1,\ldots,r_n}$ $(1 \leq i \leq n)$.

In particolare, per i sistemi espliciti alle differenze abbiamo

Proposizione

Sia $\mathrm{T} : R_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ *l'automorfismo di transizione di stato correspondente al sistema invertibile (1), precisamente*

$$x_i(0) \mapsto x_i(1), \ldots, x_i(r_i - 2) \mapsto x_i(r_i - 1), x_i(r_i - 1) \mapsto f_i.$$

Poniamo $R'_{r_1,\ldots,r_n} = \mathbb{K}[X'_{r_1,\ldots,r_n}]$ *e sia* $Q_{r_1,\ldots,r_n} = R_{r_1,\ldots,r_n} \otimes R'_{r_1,\ldots,r_n}$. *Consideriamo l'ideale* $J \subset Q_{r_1,\ldots,r_n}$ *generato dai polinomi*

$$x'_i(0) - x_i(1), \ldots, x'_i(r_i - 2) - x_i(r_i - 1), x'_i(r_i - 1) - f_i.$$

Allora, la base di Gröbner ridotta di J è del tipo

$$x_i(1) - x'_i(0), \ldots, x_i(r_i - 1) - x'_i(r_i - 2), x_i(0) - f'_i.$$

dove $f'_i \in R'_{r_1,\ldots,r_n}$ $(1 \leq i \leq n)$.

In particolare, per i sistemi espliciti alle differenze abbiamo

Proposizione

Sia $\mathrm{T} : R_{r_1,\dots,r_n} \to R_{r_1,\dots,r_n}$ *l'automorfismo di transizione di stato correspondente al sistema invertibile (1), precisamente*

$$x_i(0) \mapsto x_i(1), \dots, x_i(r_i - 2) \mapsto x_i(r_i - 1), x_i(r_i - 1) \mapsto f_i.$$

Poniamo $R'_{r_1,\dots,r_n} = \mathbb{K}[X'_{r_1,\dots,r_n}]$ *e sia* $Q_{r_1,\dots,r_n} = R_{r_1,\dots,r_n} \otimes R'_{r_1,\dots,r_n}$. *Consideriamo l'ideale* $J \subset Q_{r_1,\dots,r_n}$ *generato dai polinomi*

$$x'_i(0) - x_i(1), \dots, x'_i(r_i - 2) - x_i(r_i - 1), x'_i(r_i - 1) - f_i.$$

Allora, la base di Gröbner ridotta di J *è del tipo*

$$x_i(1) - x'_i(0), \dots, x_i(r_i - 1) - x'_i(r_i - 2), x_i(0) - f'_i.$$

dove $f'_i \in R'_{r_1,\dots,r_n}$ $(1 \leq i \leq n)$.

Definizione

Con le notazioni del precedente risultato, sia

$$G = \bigcup_i \{x_i(1) - x_i'(0), \ldots, x_i(r_i - 1) - x_i'(r_i - 2), x_i(0) - f_i'\}$$

la base di Gröbner ridotta dell'ideale J. Denotiamo con g_i l'immagine del polinomio f_i' rispetto all'isomorfismo $R'_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ tale che

$$x_i'(0) \mapsto x_i(r_i - 1), x_i'(1) \mapsto x_i(r_i - 2), \ldots, x_i'(r_i - 1) \mapsto x_i(0).$$

Il sistema inverso del sistema invertibile (1) è *per definizione*

$$\left\{ \begin{array}{rcl} x_1(r_1) & = & g_1, \\ & \vdots & \\ x_n(r_n) & = & g_n. \end{array} \right. \quad (2)$$

Definizione

Con le notazioni del precedente risultato, sia

$$G = \bigcup_i \{x_i(1) - x_i'(0), \ldots, x_i(r_i - 1) - x_i'(r_i - 2), x_i(0) - f_i'\}$$

la base di Gröbner ridotta dell'ideale J. Denotiamo con g_i l'immagine del polinomio f_i' rispetto all'isomorfismo $R'_{r_1,\ldots,r_n} \to R_{r_1,\ldots,r_n}$ tale che

$$x_i'(0) \mapsto x_i(r_i - 1), x_i'(1) \mapsto x_i(r_i - 2), \ldots, x_i'(r_i - 1) \mapsto x_i(0).$$

Il sistema inverso del sistema invertibile (1) *è per definizione*

$$\left\{\begin{array}{rcl} x_1(r_1) & = & g_1, \\ & \vdots & \\ x_n(r_n) & = & g_n. \end{array}\right. \qquad (2)$$

- Gli endomorfismi (e quindi le funzioni) di transizione di stato di un sistema invertibile (1) e del suo sistema inverso (2) sono l'uno l'inverso dell'altro a meno della permutazione di variabili (isomorfismo)

$$x_i(0) \mapsto x_i(r_i - 1), x_i(1) \mapsto x_i(r_i - 2), \ldots, x_i(r_i - 1) \mapsto x_i(0).$$

- Questo permette di calcolare lo stato iniziale $v(0)$ da uno stato qualsiasi $v(t)$ di (1) considerando $v(t)$ come stato iniziale di (2) (a meno di riordinare le coordinate).
- Ad esempio, l'inverso del sistema delle equazioni di stato di Trivium è

$$\left\{ \begin{array}{rcl} x(93) & = & y(0) + x(66) + y(78) + x(91)x(92), \\ y(84) & = & z(0) + y(69) + z(87) + y(82)y(83), \\ z(111) & = & x(0) + z(66) + x(69) + z(109)z(110). \end{array} \right.$$

- Gli endomorfismi (e quindi le funzioni) di transizione di stato di un sistema invertibile (1) e del suo sistema inverso (2) sono l'uno l'inverso dell'altro a meno della permutazione di variabili (isomorfismo)

$$x_i(0) \mapsto x_i(r_i - 1), x_i(1) \mapsto x_i(r_i - 2), \ldots, x_i(r_i - 1) \mapsto x_i(0).$$

- Questo permette di calcolare lo stato iniziale $v(0)$ da uno stato qualsiasi $v(t)$ di (1) considerando $v(t)$ come stato iniziale di (2) (a meno di riordinare le coordinate).
- Ad esempio, l'inverso del sistema delle equazioni di stato di Trivium è

$$\left\{ \begin{array}{rcl} x(93) & = & y(0) + x(66) + y(78) + x(91)x(92), \\ y(84) & = & z(0) + y(69) + z(87) + y(82)y(83), \\ z(111) & = & x(0) + z(66) + x(69) + z(109)z(110). \end{array} \right.$$

- Gli endomorfismi (e quindi le funzioni) di transizione di stato di un sistema invertibile (1) e del suo sistema inverso (2) sono l'uno l'inverso dell'altro a meno della permutazione di variabili (isomorfismo)

$$x_i(0) \mapsto x_i(r_i - 1), x_i(1) \mapsto x_i(r_i - 2), \ldots, x_i(r_i - 1) \mapsto x_i(0).$$

- Questo permette di calcolare lo stato iniziale $v(0)$ da uno stato qualsiasi $v(t)$ di (1) considerando $v(t)$ come stato iniziale di (2) (a meno di riordinare le coordinate).
- Ad esempio, l'inverso del sistema delle equazioni di stato di Trivium è

$$\left\{ \begin{array}{rcl} x(93) & = & y(0) + x(66) + y(78) + x(91)x(92), \\ y(84) & = & z(0) + y(69) + z(87) + y(82)y(83), \\ z(111) & = & x(0) + z(66) + x(69) + z(109)z(110). \end{array} \right.$$

Definizione

Un sistema (1) *si dice* riducibile *se per qualche $m < n$ ammette un sottosistema*

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_m(r_m) & = & f_m. \end{cases} \qquad (f_i \in R_{r_1,\ldots,r_m}) \tag{3}$$

In questo caso, l'endomorfismo e la funzione di transizione di stato di (3) *sono le restrizioni delle corrispondenti mappe di* (1).

Definizione

Sia $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ la funzione di transizione di stato di un sistema invertibile. Chiamiamo il sistema periodico *se esiste un intero $d > 0$ tale che $\hat{\mathrm{T}}^d = \mathrm{id}$. In tal caso, tutte le $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ sono costituite da funzioni periodiche, cioè $a_i(t) = a_i(t + d)$ per ogni $t \geq 0$. Poichè $\hat{\mathrm{T}} \in \mathbb{S}(\mathbb{K})$, se $\mathbb{K}$ è un campo finito allora tutti i sistemi invertibili sono periodici.*

Definizione

Un sistema (1) *si dice* riducibile *se per qualche $m < n$ ammette un sottosistema*

$$\begin{cases} x_1(r_1) &= f_1, \\ & \vdots \qquad\qquad (f_i \in R_{r_1,\ldots,r_m}) \\ x_m(r_m) &= f_m. \end{cases} \tag{3}$$

In questo caso, l'endomorfismo e la funzione di transizione di stato di (3) *sono le restrizioni delle corrispondenti mappe di* (1).

Definizione

Sia $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ la funzione di transizione di stato di un sistema invertibile. Chiamiamo il sistema periodico *se esiste un intero $d > 0$ tale che $\hat{\mathrm{T}}^d = \mathrm{id}$. In tal caso, tutte le $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ sono costituite da funzioni periodiche, cioè $a_i(t) = a_i(t + d)$ per ogni $t \geq 0$. Poichè $\hat{\mathrm{T}} \in \mathbb{S}(\mathbb{K})$, se $\mathbb{K}$ è un campo finito allora tutti i sistemi invertibili sono periodici.*

Definizione

Un sistema (1) *si dice* riducibile *se per qualche* $m < n$ *ammette un sottosistema*

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_m(r_m) & = & f_m. \end{cases} \qquad (f_i \in R_{r_1,\dots,r_m}) \tag{3}$$

In questo caso, l'endomorfismo e la funzione di transizione di stato di (3) *sono le restrizioni delle corrispondenti mappe di* (1).

Definizione

Sia $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ *la funzione di transizione di stato di un sistema invertibile. Chiamiamo il sistema* periodico *se esiste un intero* $d > 0$ *tale che* $\hat{\mathrm{T}}^d = \mathrm{id}$. *In tal caso, tutte le* $\mathbb{K}$*-soluzioni* $(a_1, \dots, a_n)$ *sono costituite da funzioni periodiche, cioè* $a_i(t) = a_i(t + d)$ *per ogni* $t \geq 0$. *Poichè* $\hat{\mathrm{T}} \in \mathbb{S}(\mathbb{K})$, *se* $\mathbb{K}$ *è un campo finito allora tutti i sistemi invertibili sono periodici.*

Definizione

Un sistema (1) *si dice* riducibile *se per qualche $m < n$ ammette un sottosistema*

$$\left\{ \begin{array}{rcl} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_m(r_m) & = & f_m. \end{array} \right. \quad (f_i \in R_{r_1,\ldots,r_m}) \tag{3}$$

In questo caso, l'endomorfismo e la funzione di transizione di stato di (3) *sono le restrizioni delle corrispondenti mappe di* (1).

Definizione

Sia $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ la funzione di transizione di stato di un sistema invertibile. Chiamiamo il sistema periodico *se esiste un intero $d > 0$ tale che $\hat{\mathrm{T}}^d = \mathrm{id}$. In tal caso, tutte le $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ sono costituite da funzioni periodiche, cioè $a_i(t) = a_i(t + d)$ per ogni $t \geq 0$.*

Poichè $\hat{\mathrm{T}} \in \mathbb{S}(\mathbb{K})$, se $\mathbb{K}$ è un campo finito allora tutti i sistemi invertibili sono periodici.

Definizione

Un sistema (1) *si dice* riducibile *se per qualche $m < n$ ammette un sottosistema*

$$\left\{ \begin{array}{rcl} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_m(r_m) & = & f_m. \end{array} \right. \qquad (f_i \in R_{r_1,\dots,r_m}) \tag{3}$$

In questo caso, l'endomorfismo e la funzione di transizione di stato di (3) *sono le restrizioni delle corrispondenti mappe di* (1).

Definizione

Sia $\hat{\mathrm{T}} : \mathbb{K}^r \to \mathbb{K}^r$ *la funzione di transizione di stato di un sistema invertibile. Chiamiamo il sistema* periodico *se esiste un intero $d > 0$ tale che $\hat{\mathrm{T}}^d = \mathrm{id}$. In tal caso, tutte le $\mathbb{K}$-soluzioni $(a_1, \dots, a_n)$ sono costituite da funzioni periodiche, cioè $a_i(t) = a_i(t + d)$ per ogni $t \geq 0$. Poichè $\hat{\mathrm{T}} \in \mathbb{S}(\mathbb{K})$, se $\mathbb{K}$ è un campo finito allora tutti i sistemi invertibili sono periodici.*

Nel seguito assumiamo $\mathbb{K} = \mathrm{GF}(q)$ un campo finito.

Definizione

Un cifrario a flusso alle differenze $\mathcal{C}$ *è per definizione un sistema esplicito alle differenze* (1) *insieme ad un polinomio* $f \in R_{r_1,\ldots,r_n}$. *Se* $(a_1, \ldots, a_n)$ *è una* $\mathbb{K}$*-soluzione, il suo stato iniziale è chiamato la* chiave di $(a_1, \ldots, a_n)$. *Inoltre, se il vettore* $v(t) \in \mathbb{K}^r$ *denota uno stato di* $(a_1, \ldots, a_n)$, *la funzione* $b : \mathbb{N} \to \mathbb{K}$ *tale che* $b(t) = f(v(t))$ *per ogni* $t \geq 0$, *si chiama il* keystream di $(a_1, \ldots, a_n)$. *Infine, chiamiamo* f *il* polinomio di keystream del cifrario $\mathcal{C}$.

Essendo $R_{r_1,\ldots,r_n,0} = R_{r_1,\ldots,r_n}$, un tale cifrario si può definire pure come uno speciale sistema

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n, \\ y(0) & = & f. \end{cases}$$

Nel seguito assumiamo $\mathbb{K} = \mathrm{GF}(q)$ un campo finito.

Definizione

Un cifrario a flusso alle differenze $\mathcal{C}$ *è per definizione un sistema esplicito alle differenze* (1) *insieme ad un polinomio* $f \in R_{r_1,\ldots,r_n}$. *Se* $(a_1, \ldots, a_n)$ *è una* $\mathbb{K}$*-soluzione, il suo stato iniziale è chiamato la* chiave di $(a_1, \ldots, a_n)$. *Inoltre, se il vettore* $v(t) \in \mathbb{K}^r$ *denota uno stato di* $(a_1, \ldots, a_n)$, *la funzione* $b : \mathbb{N} \to \mathbb{K}$ *tale che* $b(t) = f(v(t))$ *per ogni* $t \geq 0$, *si chiama il* keystream di $(a_1, \ldots, a_n)$. *Infine, chiamiamo* f *il* polinomio di keystream del cifrario $\mathcal{C}$.

Essendo $R_{r_1,\ldots,r_n,0} = R_{r_1,\ldots,r_n}$, un tale cifrario si può definire pure come uno speciale sistema

$$\begin{cases} x_1(r_1) &= f_1, \\ & \vdots \\ x_n(r_n) &= f_n, \\ y(0) &= f. \end{cases}$$

Nel seguito assumiamo $\mathbb{K} = \mathrm{GF}(q)$ un campo finito.

Definizione

Un cifrario a flusso alle differenze $\mathcal{C}$ *è per definizione un sistema esplicito alle differenze* (1) *insieme ad un polinomio* $f \in R_{r_1,\ldots,r_n}$. *Se* $(a_1,\ldots,a_n)$ *è una* $\mathbb{K}$*-soluzione, il suo stato iniziale è chiamato la* chiave di $(a_1,\ldots,a_n)$. *Inoltre, se il vettore* $v(t) \in \mathbb{K}^r$ *denota uno stato di* $(a_1,\ldots,a_n)$, *la funzione* $b : \mathbb{N} \to \mathbb{K}$ *tale che* $b(t) = f(v(t))$ *per ogni* $t \geq 0$, *si chiama il* keystream di $(a_1,\ldots,a_n)$. *Infine, chiamiamo* f *il* polinomio di keystream del cifrario $\mathcal{C}$.

Essendo $R_{r_1,\ldots,r_n,0} = R_{r_1,\ldots,r_n}$, un tale cifrario si può definire pure come uno speciale sistema

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n, \\ y(0) & = & f. \end{cases}$$

Nel seguito assumiamo $\mathbb{K} = \mathrm{GF}(q)$ un campo finito.

Definizione

Un cifrario a flusso alle differenze $\mathcal{C}$ *è per definizione un sistema esplicito alle differenze* (1) *insieme ad un polinomio* $f \in R_{r_1,\ldots,r_n}$. *Se* $(a_1,\ldots,a_n)$ *è una* $\mathbb{K}$*-soluzione, il suo stato iniziale è chiamato la* chiave di $(a_1,\ldots,a_n)$. *Inoltre, se il vettore* $v(t) \in \mathbb{K}^r$ *denota uno stato di* $(a_1,\ldots,a_n)$*, la funzione* $b : \mathbb{N} \to \mathbb{K}$ *tale che* $b(t) = f(v(t))$ *per ogni* $t \geq 0$*, si chiama il* keystream di $(a_1,\ldots,a_n)$. *Infine, chiamiamo f il* polinomio di keystream del cifrario $\mathcal{C}$.

Essendo $R_{r_1,\ldots,r_n,0} = R_{r_1,\ldots,r_n}$, un tale cifrario si può definire pure come uno speciale sistema

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n, \\ y(0) & = & f. \end{cases}$$

Nel seguito assumiamo $\mathbb{K} = \mathrm{GF}(q)$ un campo finito.

Definizione

Un cifrario a flusso alle differenze $\mathcal{C}$ *è per definizione un sistema esplicito alle differenze* (1) *insieme ad un polinomio* $f \in R_{r_1,\ldots,r_n}$*. Se* $(a_1, \ldots, a_n)$ *è una* $\mathbb{K}$*-soluzione, il suo stato iniziale è chiamato la* chiave di $(a_1, \ldots, a_n)$*. Inoltre, se il vettore* $v(t) \in \mathbb{K}^r$ *denota uno stato di* $(a_1, \ldots, a_n)$*, la funzione* $b : \mathbb{N} \to \mathbb{K}$ *tale che* $b(t) = f(v(t))$ *per ogni* $t \geq 0$*, si chiama il* keystream di $(a_1, \ldots, a_n)$*. Infine, chiamiamo* f *il* polinomio di keystream del cifrario $\mathcal{C}$.

Essendo $R_{r_1,\ldots,r_n,0} = R_{r_1,\ldots,r_n}$, un tale cifrario si può definire pure come uno speciale sistema

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n, \\ y(0) & = & f. \end{cases}$$

Nel seguito assumiamo $\mathbb{K} = \mathrm{GF}(q)$ un campo finito.

Definizione

Un cifrario a flusso alle differenze $\mathcal{C}$ *è per definizione un sistema esplicito alle differenze* (1) *insieme ad un polinomio* $f \in R_{r_1,\ldots,r_n}$. *Se* $(a_1, \ldots, a_n)$ *è una* $\mathbb{K}$*-soluzione, il suo stato iniziale è chiamato la* chiave di $(a_1, \ldots, a_n)$. *Inoltre, se il vettore* $v(t) \in \mathbb{K}^r$ *denota uno stato di* $(a_1, \ldots, a_n)$, *la funzione* $b : \mathbb{N} \to \mathbb{K}$ *tale che* $b(t) = f(v(t))$ *per ogni* $t \geq 0$, *si chiama il* keystream di $(a_1, \ldots, a_n)$. *Infine, chiamiamo* f *il* polinomio di keystream del cifrario $\mathcal{C}$.

Essendo $R_{r_1,\ldots,r_n,0} = R_{r_1,\ldots,r_n}$, un tale cifrario si può definire pure come uno speciale sistema

$$\begin{cases} x_1(r_1) & = & f_1, \\ & \vdots & \\ x_n(r_n) & = & f_n, \\ y(0) & = & f. \end{cases}$$

Definizione

Sia $\mathcal{C}$ un cifrario a flusso definito dal sistema (1) *e dal polinomio f. Sia $b : \mathbb{N} \to \mathbb{K}$ il keystream di una $\mathbb{K}$-soluzione di* (1) *e fissiamo un clock $T \geq 0$. Consideriamo l'ideale*

$$J = \sum_{t \geq T} \langle \sigma^t(f) - b(t) \rangle \subset R$$

e denotiamo con $V_{\mathbb{K}}(J)$ l'insieme delle $\mathbb{K}$-soluzioni dei polinomi di J, equivalentemente, dei suoi generatori. Un attacco algebrico a $\mathcal{C}$ mediante il keystream b a partire dal clock T *consiste nel calcolo delle $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ del sistema* (1) *tali che $(a_1, \ldots, a_n) \in V_{\mathbb{K}}(J)$. In altri termini, se consideriamo l'ideale (alle differenze) di R associato ad* (1)

$$I = \langle x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n \rangle_\sigma =$$

$$\langle x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0 \rangle,$$

si vuole calcolare $V_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(I) \cap V_{\mathbb{K}}(J) = V_{\mathbb{K}} \cap V_{\mathbb{K}}(J)$.

Definizione

Sia $\mathcal{C}$ un cifrario a flusso definito dal sistema (1) *e dal polinomio f. Sia $b : \mathbb{N} \to \mathbb{K}$ il keystream di una $\mathbb{K}$-soluzione di* (1) *e fissiamo un clock $T \geq 0$. Consideriamo l'ideale*

$$J = \sum_{t \geq T} \langle \sigma^t(f) - b(t) \rangle \subset R$$

e denotiamo con $V_{\mathbb{K}}(J)$ l'insieme delle $\mathbb{K}$-soluzioni dei polinomi di J, equivalentemente, dei suoi generatori. *Un* attacco algebrico a $\mathcal{C}$ mediante il keystream b a partire dal clock T *consiste nel calcolo delle $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ del sistema* (1) *tali che $(a_1, \ldots, a_n) \in V_{\mathbb{K}}(J)$. In altri termini, se consideriamo l'ideale (alle differenze) di R associato ad* (1)

$$I = \langle x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n \rangle_\sigma =$$

$$\langle x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0 \rangle,$$

si vuole calcolare $V_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(I) \cap V_{\mathbb{K}}(J) = V_{\mathbb{K}} \cap V_{\mathbb{K}}(J)$.

Definizione

Sia $\mathcal{C}$ un cifrario a flusso definito dal sistema (1) *e dal polinomio f. Sia $b : \mathbb{N} \to \mathbb{K}$ il keystream di una $\mathbb{K}$-soluzione di* (1) *e fissiamo un clock $T \geq 0$. Consideriamo l'ideale*

$$J = \sum_{t \geq T} \langle \sigma^t(f) - b(t) \rangle \subset R$$

e denotiamo con $V_{\mathbb{K}}(J)$ l'insieme delle $\mathbb{K}$-soluzioni dei polinomi di J, equivalentemente, dei suoi generatori. Un attacco algebrico a $\mathcal{C}$ mediante il keystream b a partire dal clock T *consiste nel calcolo delle $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ del sistema* (1) *tali che $(a_1, \ldots, a_n) \in V_{\mathbb{K}}(J)$.*

In altri termini, se consideriamo l'ideale (alle differenze) di R associato ad (1)

$$I = \langle x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n \rangle_\sigma =$$
$$\langle x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0 \rangle,$$

si vuole calcolare $V_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(I) \cap V_{\mathbb{K}}(J) = V_{\mathbb{K}} \cap V_{\mathbb{K}}(J)$.

Definizione

Sia $\mathcal{C}$ un cifrario a flusso definito dal sistema (1) *e dal polinomio f. Sia $b : \mathbb{N} \to \mathbb{K}$ il keystream di una $\mathbb{K}$-soluzione di* (1) *e fissiamo un clock $T \geq 0$. Consideriamo l'ideale*

$$J = \sum_{t \geq T} \langle \sigma^t(f) - b(t) \rangle \subset R$$

e denotiamo con $V_{\mathbb{K}}(J)$ l'insieme delle $\mathbb{K}$-soluzioni dei polinomi di J, equivalentemente, dei suoi generatori. Un attacco algebrico a $\mathcal{C}$ mediante il keystream b a partire dal clock T *consiste nel calcolo delle $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ del sistema* (1) *tali che $(a_1, \ldots, a_n) \in V_{\mathbb{K}}(J)$. In altri termini, se consideriamo l'ideale (alle differenze) di R associato ad* (1)

$$I = \langle x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n \rangle_\sigma =$$

$$\langle x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0 \rangle,$$

si vuole calcolare $V_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(I) \cap V_{\mathbb{K}}(J) = V_{\mathbb{K}} \cap V_{\mathbb{K}}(J)$.

- Per definizione, $V_{\mathbb{K}}(I + J)$ è la varietà delle $\mathbb{K}$-soluzioni del sistema (1) che sono compatibili con il keystream osservato $b(t)$ ($t \geq T$). Gli stati iniziali di queste soluzioni sono le chiavi compatibili con il keystream.

- Potrebbe essere conveniente calcolare direttamente le equazioni della varietà $\bar{V}_{\mathbb{K}}(I + J) \subset \mathbb{K}^r$ delle chiavi compatibili con il keystream. Questa è l'immagine della varietà $V_{\mathbb{K}}(I + J)$ rispetto alla bigezione polinomiale $V_{\mathbb{K}} \to \mathbb{K}^r$ che mappa le $\mathbb{K}$-soluzioni del sistema (1) nei loro stati iniziali.

- Per definizione, $V_{\mathbb{K}}(I + J)$ è la varietà delle $\mathbb{K}$-soluzioni del sistema (1) che sono compatibili con il keystream osservato $b(t)$ $(t \geq T)$. Gli stati iniziali di queste soluzioni sono le chiavi compatibili con il keystream.

- Potrebbe essere conveniente calcolare direttamente le equazioni della varietà $\bar{V}_{\mathbb{K}}(I + J) \subset \mathbb{K}^r$ delle chiavi compatibili con il keystream. Questa è l'immagine della varietà $V_{\mathbb{K}}(I + J)$ rispetto alla bigezione polinomiale $V_{\mathbb{K}} \to \mathbb{K}^r$ che mappa le $\mathbb{K}$-soluzioni del sistema (1) nei loro stati iniziali.

Assumiamo $x_i(r_i) \succ \mathrm{lm}(f_i)$ ($1 \leq i \leq n$) per un ordinamento monomiale di R compatibile con l'operatore di shift σ ($m \prec n$ implica $\sigma(m) \prec \sigma(n)$).

Allora, l'insieme $\{x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n\}$ è una *base di Gröbner alle differenze di I*, cioè $\{x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0\}$ è una base di Gröbner di I.

Proposizione

Denotiamo con $f'_t \in R_{r_1,\ldots,r_n}$ la forma normale del polinomio $\sigma^t(f)$ modulo I e definiamo l'ideale

$$J' = \sum_{t \geq T} \langle f'_t - b(t) \rangle \subset R_{r_1,\ldots,r_n}.$$

Abbiamo allora che $\bar{V}_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(J')$.

Assumiamo $x_i(r_i) \succ \mathrm{lm}(f_i)$ ($1 \leq i \leq n$) per un ordinamento monomiale di R compatibile con l'operatore di shift σ ($m \prec n$ implica $\sigma(m) \prec \sigma(n)$). Allora, l'insieme $\{x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n\}$ è una *base di Gröbner alle differenze di I*, cioè $\{x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0\}$ è una base di Gröbner di I.

Proposizione

Denotiamo con $f'_t \in R_{r_1,\ldots,r_n}$ la forma normale del polinomio $\sigma^t(f)$ modulo I e definiamo l'ideale

$$J' = \sum_{t \geq T} \langle f'_t - b(t) \rangle \subset R_{r_1,\ldots,r_n}.$$

Abbiamo allora che $\bar{V}_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(J')$.

Assumiamo $x_i(r_i) \succ \mathrm{lm}(f_i)$ ($1 \leq i \leq n$) per un ordinamento monomiale di R compatibile con l'operatore di shift σ ($m \prec n$ implica $\sigma(m) \prec \sigma(n)$). Allora, l'insieme $\{x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n\}$ è una *base di Gröbner alle differenze di* I, cioè $\{x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0\}$ è una base di Gröbner di I.

Proposizione

Denotiamo con $f'_t \in R_{r_1,\ldots,r_n}$ *la forma normale del polinomio* $\sigma^t(f)$ *modulo* I *e definiamo l'ideale*

$$J' = \sum_{t \geq T} \langle f'_t - b(t) \rangle \subset R_{r_1,\ldots,r_n}.$$

Abbiamo allora che $\bar{V}_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(J')$.

Assumiamo $x_i(r_i) \succ \mathrm{lm}(f_i)$ $(1 \leq i \leq n)$ per un ordinamento monomiale di R compatibile con l'operatore di shift σ ($m \prec n$ implica $\sigma(m) \prec \sigma(n)$). Allora, l'insieme $\{x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n\}$ è una *base di Gröbner alle differenze di I*, cioè $\{x_1(r_1 + t) - \sigma^t(f_1), \ldots, x_n(r_n + t) - \sigma^t(f_n) \mid t \geq 0\}$ è una base di Gröbner di I.

Proposizione

Denotiamo con $f'_t \in R_{r_1,\ldots,r_n}$ la forma normale del polinomio $\sigma^t(f)$ modulo I e definiamo l'ideale

$$J' = \sum_{t \geq T} \langle f'_t - b(t) \rangle \subset R_{r_1,\ldots,r_n}.$$

Abbiamo allora che $\bar{V}_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(J')$.

- In un vero attacco algebrico abbiamo solo un numero finito di valori del keystream, ovvero dato un clock $B \geq T$ possiamo effettivamente costruire l'ideale

$$J'_B = \sum_{T \leq t \leq B} \langle f'_t - b(t) \rangle \subset R_{r_1,\dots,r_n}.$$

- Chiaramente $J' = \bigcup_{B \geq T} J'_B$ con $J'_B \subset J'_{B+1}$. Poichè l'algebra $R_{r_1,\dots,r_n}$ è finitamente generata e quindi Noetheriana, abbiamo che $J'_B = J'$ per un opportuno clock $B \geq T$.
- In altri termini, con un numero sufficiente di valori del keystream non perdiamo alcuna equazione soddisfatta dalle chiavi.

- In un vero attacco algebrico abbiamo solo un numero finito di valori del keystream, ovvero dato un clock $B \geq T$ possiamo effettivamente costruire l'ideale

$$J'_B = \sum_{T \leq t \leq B} \langle f'_t - b(t) \rangle \subset R_{r_1,\ldots,r_n}.$$

- Chiaramente $J' = \bigcup_{B \geq T} J'_B$ con $J'_B \subset J'_{B+1}$. Poichè l'algebra $R_{r_1,\ldots,r_n}$ è finitamente generata e quindi Noetheriana, abbiamo che $J'_B = J'$ per un opportuno clock $B \geq T$.
- In altri termini, con un numero sufficiente di valori del keystream non perdiamo alcuna equazione soddisfatta dalle chiavi.

- In un vero attacco algebrico abbiamo solo un numero finito di valori del keystream, ovvero dato un clock $B \geq T$ possiamo effettivamente costruire l'ideale

$$J'_B = \sum_{T \leq t \leq B} \langle f'_t - b(t) \rangle \subset R_{r_1,\ldots,r_n}.$$

- Chiaramente $J' = \bigcup_{B \geq T} J'_B$ con $J'_B \subset J'_{B+1}$. Poichè l'algebra $R_{r_1,\ldots,r_n}$ è finitamente generata e quindi Noetheriana, abbiamo che $J'_B = J'$ per un opportuno clock $B \geq T$.
- In altri termini, con un numero sufficiente di valori del keystream non perdiamo alcuna equazione soddisfatta dalle chiavi.

Per calcolare l'insieme delle chiavi $V_{\mathbb{K}}(J'_B) \subset \mathbb{K}^r$ possiamo usare basi di Gröbner oppure SAT solvers se $\mathbb{K} = \mathrm{GF}(2)$. Per i cifrari reali, abbiamo generalmente che $V_{\mathbb{K}}(J'_B)$ contiene una singola chiave. Possiamo applicare quindi il seguente risultato che è conseguenza diretta dello Shape Lemma.

Proposizione

Consideriamo l'algebra $P = \mathbb{K}[x_1, \ldots, x_r]$ ($\mathbb{K} = \mathrm{GF}(q)$) e l'ideale $L = \langle x_1^q - x_1, \ldots, x_r^q - x_r \rangle \subset P$. Sia $J \subset P$ un ideale e denotiamo con $V(J)$ l'insieme delle $\bar{\mathbb{K}}$-soluzioni dei polinomi $f \in J$ dove il campo $\bar{\mathbb{K}}$ è la chiusura algebrica di $\mathbb{K}$. Abbiamo allora che $V(L) = \mathbb{K}^r$ e $V_{\mathbb{K}}(J) = V(J) \cap \mathbb{K}^r = V(J + L)$ dove $J + L \subset P$ è un ideale radicale. Inoltre, se $V_{\mathbb{K}}(J) = \{(\alpha_1, \ldots, \alpha_n)\}$ allora $G = \{x_1 - \alpha_1, \ldots, x_n - \alpha_n\}$ è la base di Gröbner universale (ridotta) di $J + L$, cioè, G è la sua base di Gröbner rispetto ad ogni ordinamento monomiale di P.

Per calcolare l'insieme delle chiavi $V_{\mathbb{K}}(J'_B) \subset \mathbb{K}^r$ possiamo usare basi di Gröbner oppure SAT solvers se $\mathbb{K} = \mathrm{GF}(2)$. Per i cifrari reali, abbiamo generalmente che $V_{\mathbb{K}}(J'_B)$ contiene una singola chiave.
Possiamo applicare quindi il seguente risultato che è conseguenza diretta dello Shape Lemma.

Proposizione

Consideriamo l'algebra $P = \mathbb{K}[x_1, \ldots, x_r]$ ($\mathbb{K} = \mathrm{GF}(q)$) e l'ideale $L = \langle x_1^q - x_1, \ldots, x_r^q - x_r \rangle \subset P$. Sia $J \subset P$ un ideale e denotiamo con $V(J)$ l'insieme delle $\bar{\mathbb{K}}$-soluzioni dei polinomi $f \in J$ dove il campo $\bar{\mathbb{K}}$ è la chiusura algebrica di $\mathbb{K}$. Abbiamo allora che $V(L) = \mathbb{K}^r$ e $V_{\mathbb{K}}(J) = V(J) \cap \mathbb{K}^r = V(J + L)$ dove $J + L \subset P$ è un ideale radicale. Inoltre, se $V_{\mathbb{K}}(J) = \{(\alpha_1, \ldots, \alpha_n)\}$ allora $G = \{x_1 - \alpha_1, \ldots, x_n - \alpha_n\}$ è la base di Gröbner universale (ridotta) di $J + L$, cioè, G è la sua base di Gröbner rispetto ad ogni ordinamento monomiale di P.

Per calcolare l'insieme delle chiavi $V_{\mathbb{K}}(J'_B) \subset \mathbb{K}^r$ possiamo usare basi di Gröbner oppure SAT solvers se $\mathbb{K} = \mathrm{GF}(2)$. Per i cifrari reali, abbiamo generalmente che $V_{\mathbb{K}}(J'_B)$ contiene una singola chiave. Possiamo applicare quindi il seguente risultato che è conseguenza diretta dello Shape Lemma.

Proposizione

Consideriamo l'algebra $P = \mathbb{K}[x_1, \ldots, x_r]$ ($\mathbb{K} = \mathrm{GF}(q)$) e l'ideale $L = \langle x_1^q - x_1, \ldots, x_r^q - x_r \rangle \subset P$. Sia $J \subset P$ un ideale e denotiamo con $V(J)$ l'insieme delle $\bar{\mathbb{K}}$-soluzioni dei polinomi $f \in J$ dove il campo $\bar{\mathbb{K}}$ è la chiusura algebrica di $\mathbb{K}$. Abbiamo allora che $V(L) = \mathbb{K}^r$ e $V_{\mathbb{K}}(J) = V(J) \cap \mathbb{K}^r = V(J + L)$ dove $J + L \subset P$ è un ideale radicale. Inoltre, se $V_{\mathbb{K}}(J) = \{(\alpha_1, \ldots, \alpha_n)\}$ allora $G = \{x_1 - \alpha_1, \ldots, x_n - \alpha_n\}$ è la base di Gröbner universale (ridotta) di $J + L$, cioè, G è la sua base di Gröbner rispetto ad ogni ordinamento monomiale di P.

Per calcolare l'insieme delle chiavi $V_{\mathbb{K}}(J'_B) \subset \mathbb{K}^r$ possiamo usare basi di Gröbner oppure SAT solvers se $\mathbb{K} = \mathrm{GF}(2)$. Per i cifrari reali, abbiamo generalmente che $V_{\mathbb{K}}(J'_B)$ contiene una singola chiave. Possiamo applicare quindi il seguente risultato che è conseguenza diretta dello Shape Lemma.

Proposizione

Consideriamo l'algebra $P = \mathbb{K}[x_1, \ldots, x_r]$ ($\mathbb{K} = \mathrm{GF}(q)$) e l'ideale $L = \langle x_1^q - x_1, \ldots, x_r^q - x_r \rangle \subset P$. Sia $J \subset P$ un ideale e denotiamo con $V(J)$ l'insieme delle $\bar{\mathbb{K}}$-soluzioni dei polinomi $f \in J$ dove il campo $\bar{\mathbb{K}}$ è la chiusura algebrica di $\mathbb{K}$. Abbiamo allora che $V(L) = \mathbb{K}^r$ e $V_{\mathbb{K}}(J) = V(J) \cap \mathbb{K}^r = V(J + L)$ dove $J + L \subset P$ è un ideale radicale. Inoltre, se $V_{\mathbb{K}}(J) = \{(\alpha_1, \ldots, \alpha_n)\}$ allora $G = \{x_1 - \alpha_1, \ldots, x_n - \alpha_n\}$ è la base di Gröbner universale (ridotta) di $J + L$, cioè, G è la sua base di Gröbner rispetto ad ogni ordinamento monomiale di P.

Per calcolare l'insieme delle chiavi $V_{\mathbb{K}}(J'_B) \subset \mathbb{K}^r$ possiamo usare basi di Gröbner oppure SAT solvers se $\mathbb{K} = \mathrm{GF}(2)$. Per i cifrari reali, abbiamo generalmente che $V_{\mathbb{K}}(J'_B)$ contiene una singola chiave. Possiamo applicare quindi il seguente risultato che è conseguenza diretta dello Shape Lemma.

Proposizione

Consideriamo l'algebra $P = \mathbb{K}[x_1, \ldots, x_r]$ ($\mathbb{K} = \mathrm{GF}(q)$) e l'ideale $L = \langle x_1^q - x_1, \ldots, x_r^q - x_r \rangle \subset P$. Sia $J \subset P$ un ideale e denotiamo con $V(J)$ l'insieme delle $\bar{\mathbb{K}}$-soluzioni dei polinomi $f \in J$ dove il campo $\bar{\mathbb{K}}$ è la chiusura algebrica di $\mathbb{K}$. Abbiamo allora che $V(L) = \mathbb{K}^r$ e $V_{\mathbb{K}}(J) = V(J) \cap \mathbb{K}^r = V(J + L)$ dove $J + L \subset P$ è un ideale radicale. Inoltre, se $V_{\mathbb{K}}(J) = \{(\alpha_1, \ldots, \alpha_n)\}$ allora $G = \{x_1 - \alpha_1, \ldots, x_n - \alpha_n\}$ è la base di Gröbner universale (ridotta) di $J + L$, cioè, G è la sua base di Gröbner rispetto ad ogni ordinamento monomiale di P.

Per calcolare l'insieme delle chiavi $V_{\mathbb{K}}(J'_B) \subset \mathbb{K}^r$ possiamo usare basi di Gröbner oppure SAT solvers se $\mathbb{K} = \mathrm{GF}(2)$. Per i cifrari reali, abbiamo generalmente che $V_{\mathbb{K}}(J'_B)$ contiene una singola chiave. Possiamo applicare quindi il seguente risultato che è conseguenza diretta dello Shape Lemma.

Proposizione

Consideriamo l'algebra $P = \mathbb{K}[x_1, \ldots, x_r]$ $(\mathbb{K} = \mathrm{GF}(q))$ *e l'ideale* $L = \langle x_1^q - x_1, \ldots, x_r^q - x_r \rangle \subset P$. *Sia* $J \subset P$ *un ideale e denotiamo con* $V(J)$ *l'insieme delle* $\bar{\mathbb{K}}$*-soluzioni dei polinomi* $f \in J$ *dove il campo* $\bar{\mathbb{K}}$ *è la chiusura algebrica di* $\mathbb{K}$. *Abbiamo allora che* $V(L) = \mathbb{K}^r$ *e* $V_{\mathbb{K}}(J) = V(J) \cap \mathbb{K}^r = V(J + L)$ *dove* $J + L \subset P$ *è un ideale radicale. Inoltre, se* $V_{\mathbb{K}}(J) = \{(\alpha_1, \ldots, \alpha_n)\}$ *allora* $G = \{x_1 - \alpha_1, \ldots, x_n - \alpha_n\}$ *è la base di Gröbner universale (ridotta) di* $J + L$*, cioè,* G *è la sua base di Gröbner rispetto ad ogni ordinamento monomiale di* P.

- Questo risultato è molto utile per implementare attacchi algebrici perchè il calcolo delle basi di Gröbner è molto sensibile agli ordinamenti monomiali scelti. Nel caso di soluzione unica siamo liberi di scegliere gli ordinamenti più efficienti come DegRevLex.
- Un'altra possibile ottimizzazione per questo caso consiste nello stoppare l'algoritmo di Buchberger quando tutte le variabili x_i si sono ottenute come leading monomial di un elemento nella base corrente.
- Osserviamo che la forma normale f_t' del polinomio $\sigma^t(f)$ $(T \leq t \leq B)$ ha generalmente grado alto se il clock iniziale T del keystream è grande, come accade in molti cifrari reali.

- Questo risultato è molto utile per implementare attacchi algebrici perchè il calcolo delle basi di Gröbner è molto sensibile agli ordinamenti monomiali scelti. Nel caso di soluzione unica siamo liberi di scegliere gli ordinamenti più efficienti come DegRevLex.
- Un'altra possibile ottimizzazione per questo caso consiste nello stoppare l'algoritmo di Buchberger quando tutte le variabili x_i si sono ottenute come leading monomial di un elemento nella base corrente.
- Osserviamo che la forma normale f_t' del polinomio $\sigma^t(f)$ $(T \leq t \leq B)$ ha generalmente grado alto se il clock iniziale T del keystream è grande, come accade in molti cifrari reali.

- Questo risultato è molto utile per implementare attacchi algebrici perchè il calcolo delle basi di Gröbner è molto sensibile agli ordinamenti monomiali scelti. Nel caso di soluzione unica siamo liberi di scegliere gli ordinamenti più efficienti come DegRevLex.
- Un'altra possibile ottimizzazione per questo caso consiste nello stoppare l'algoritmo di Buchberger quando tutte le variabili x_i si sono ottenute come leading monomial di un elemento nella base corrente.
- Osserviamo che la forma normale f'_t del polinomio $\sigma^t(f)$ $(T \leq t \leq B)$ ha generalmente grado alto se il clock iniziale T del keystream è grande, come accade in molti cifrari reali.

- Se il sistema (1) è invertibile, possiamo assumere di fatto $T = 0$. Infatti, grazie al sistema inverso possiamo attaccare lo stato al clock T piuttosto che lo stato iniziale, cioè, la chiave. Questo è una ottimizzazione molto importante perchè riduce drasticamente i gradi dei generatori dell'ideale $J'_B = \sum_{T \leq t \leq B} \langle f'_t - b(t) \rangle$ a quelli dei generatori dell'ideale

$$J''_B = \sum_{0 \leq t \leq B-T} \langle f'_t - b(T+t) \rangle.$$

Abbiamo usato questa strategia per attaccare Bivium.

- Se le forme normali f'_t continuano ad avere gradi troppi alti, una strategia alternativa consiste nel calcolare direttamente le $\mathbb{K}$-soluzioni del sistema (1) che sono pure soluzioni dei polinomi $\sigma^t(f) - b(t)$ che hanno tutti lo stesso grado. Questa strategia ha lo svantaggio di richiedere il calcolo di una base di Gröbner su un numero elevato di variabili.

- Se il sistema (1) è invertibile, possiamo assumere di fatto $T = 0$. Infatti, grazie al sistema inverso possiamo attaccare lo stato al clock T piuttosto che lo stato iniziale, cioè, la chiave. Questo è una ottimizzazione molto importante perchè riduce drasticamente i gradi dei generatori dell'ideale $J'_B = \sum_{T\leq t\leq B}\langle f'_t - b(t)\rangle$ a quelli dei generatori dell'ideale

$$J''_B = \sum_{0\leq t\leq B-T} \langle f'_t - b(T+t)\rangle.$$

Abbiamo usato questa strategia per attaccare Bivium.

- Se le forme normali f'_t continuano ad avere gradi troppi alti, una strategia alternativa consiste nel calcolare direttamente le $\mathbb{K}$-soluzioni del sistema (1) che sono pure soluzioni dei polinomi $\sigma^t(f) - b(t)$ che hanno tutti lo stesso grado. Questa strategia ha lo svantaggio di richiedere il calcolo di una base di Gröbner su un numero elevato di variabili.

- Se il sistema (1) è invertibile, possiamo assumere di fatto $T = 0$. Infatti, grazie al sistema inverso possiamo attaccare lo stato al clock T piuttosto che lo stato iniziale, cioè, la chiave. Questo è una ottimizzazione molto importante perchè riduce drasticamente i gradi dei generatori dell'ideale $J'_B = \sum_{T \leq t \leq B} \langle f'_t - b(t) \rangle$ a quelli dei generatori dell'ideale

$$J''_B = \sum_{0 \leq t \leq B-T} \langle f'_t - b(T+t) \rangle.$$

  Abbiamo usato questa strategia per attaccare Bivium.

- Se le forme normali f'_t continuano ad avere gradi troppi alti, una strategia alternativa consiste nel calcolare direttamente le $\mathbb{K}$-soluzioni del sistema (1) che sono pure soluzioni dei polinomi $\sigma^t(f) - b(t)$ che hanno tutti lo stesso grado. Questa strategia ha lo svantaggio di richiedere il calcolo di una base di Gröbner su un numero elevato di variabili.

- Se il sistema (1) è invertibile, possiamo assumere di fatto $T = 0$. Infatti, grazie al sistema inverso possiamo attaccare lo stato al clock T piuttosto che lo stato iniziale, cioè, la chiave. Questo è una ottimizzazione molto importante perchè riduce drasticamente i gradi dei generatori dell'ideale $J'_B = \sum_{T \leq t \leq B} \langle f'_t - b(t) \rangle$ a quelli dei generatori dell'ideale

$$J''_B = \sum_{0 \leq t \leq B-T} \langle f'_t - b(T+t) \rangle.$$

  Abbiamo usato questa strategia per attaccare Bivium.
- Se le forme normali f'_t continuano ad avere gradi troppi alti, una strategia alternativa consiste nel calcolare direttamente le $\mathbb{K}$-soluzioni del sistema (1) che sono pure soluzioni dei polinomi $\sigma^t(f) - b(t)$ che hanno tutti lo stesso grado. Questa strategia ha lo svantaggio di richiedere il calcolo di una base di Gröbner su un numero elevato di variabili.

- Se il sistema (1) è invertibile, possiamo assumere di fatto $T = 0$. Infatti, grazie al sistema inverso possiamo attaccare lo stato al clock T piuttosto che lo stato iniziale, cioè, la chiave. Questo è una ottimizzazione molto importante perchè riduce drasticamente i gradi dei generatori dell'ideale $J'_B = \sum_{T \leq t \leq B} \langle f'_t - b(t) \rangle$ a quelli dei generatori dell'ideale

$$J''_B = \sum_{0 \leq t \leq B-T} \langle f'_t - b(T+t) \rangle.$$

  Abbiamo usato questa strategia per attaccare Bivium.
- Se le forme normali f'_t continuano ad avere gradi troppi alti, una strategia alternativa consiste nel calcolare direttamente le $\mathbb{K}$-soluzioni del sistema (1) che sono pure soluzioni dei polinomi $\sigma^t(f) - b(t)$ che hanno tutti lo stesso grado. Questa strategia ha lo svantaggio di richiedere il calcolo di una base di Gröbner su un numero elevato di variabili.

- Quando calcoliamo una base di Gröbner per ottenere le soluzioni di un sistema, una strategia fondamentale consiste nell'aggiungere polinomi lineari all'ideale considerato $J \subset \mathbb{K}[x_1, \ldots, x_r]$ al fine di accellerare il calcolo.
- Questi polinomi lineari possono essere elementi di J (noti o calcolati) oppure corrispondenti alle assegnazioni di qualche sottinsieme di variabili $\{x_{i_1}, \ldots, x_{i_s}\} \subset \{x_1, \ldots, x_r\}$.
- Se qualcuna di queste assegnazioni è errata, abbiamo allora

$$J + \langle x_{i_1} - \alpha_{i_1}, \ldots, x_{i_s} - \alpha_{i_s} \rangle = \langle 1 \rangle$$

  ed il calcolo della base di Gröbner si arresta non appena l'elemento 1 è stato ottenuto.
- Per un SAT solver, invece, la risposta "UNSAT" arriva essenzialmente quando l'intero spazio $\mathbb{K}^r$ ($\mathbb{K} = \mathrm{GF}(2)$) è stato esaminato.

- Quando calcoliamo una base di Gröbner per ottenere le soluzioni di un sistema, una strategia fondamentale consiste nell'aggiungere polinomi lineari all'ideale considerato $J \subset \mathbb{K}[x_1, \ldots, x_r]$ al fine di accellerare il calcolo.
- Questi polinomi lineari possono essere elementi di J (noti o calcolati) oppure corrispondenti alle assegnazioni di qualche sottinsieme di variabili $\{x_{i_1}, \ldots, x_{i_s}\} \subset \{x_1, \ldots, x_r\}$.
- Se qualcuna di queste assegnazioni è errata, abbiamo allora

$$J + \langle x_{i_1} - \alpha_{i_1}, \ldots, x_{i_s} - \alpha_{i_s} \rangle = \langle 1 \rangle$$

ed il calcolo della base di Gröbner si arresta non appena l'elemento 1 è stato ottenuto.
- Per un SAT solver, invece, la risposta "UNSAT" arriva essenzialmente quando l'intero spazio $\mathbb{K}^r$ ($\mathbb{K} = \mathrm{GF}(2)$) è stato esaminato.

- Quando calcoliamo una base di Gröbner per ottenere le soluzioni di un sistema, una strategia fondamentale consiste nell'aggiungere polinomi lineari all'ideale considerato $J \subset \mathbb{K}[x_1, \ldots, x_r]$ al fine di accellerare il calcolo.
- Questi polinomi lineari possono essere elementi di J (noti o calcolati) oppure corrispondenti alle assegnazioni di qualche sottinsieme di variabili $\{x_{i_1}, \ldots, x_{i_s}\} \subset \{x_1, \ldots, x_r\}$.
- Se qualcuna di queste assegnazioni è errata, abbiamo allora

  $$J + \langle x_{i_1} - \alpha_{i_1}, \ldots, x_{i_s} - \alpha_{i_s} \rangle = \langle 1 \rangle$$

  ed il calcolo della base di Gröbner si arresta non appena l'elemento 1 è stato ottenuto.
- Per un SAT solver, invece, la risposta "UNSAT" arriva essenzialmente quando l'intero spazio $\mathbb{K}^r$ ($\mathbb{K} = \mathrm{GF}(2)$) è stato esaminato.

- Quando calcoliamo una base di Gröbner per ottenere le soluzioni di un sistema, una strategia fondamentale consiste nell'aggiungere polinomi lineari all'ideale considerato $J \subset \mathbb{K}[x_1, \ldots, x_r]$ al fine di accellerare il calcolo.
- Questi polinomi lineari possono essere elementi di J (noti o calcolati) oppure corrispondenti alle assegnazioni di qualche sottinsieme di variabili $\{x_{i_1}, \ldots, x_{i_s}\} \subset \{x_1, \ldots, x_r\}$.
- Se qualcuna di queste assegnazioni è errata, abbiamo allora

$$J + \langle x_{i_1} - \alpha_{i_1}, \ldots, x_{i_s} - \alpha_{i_s} \rangle = \langle 1 \rangle$$

  ed il calcolo della base di Gröbner si arresta non appena l'elemento 1 è stato ottenuto.
- Per un SAT solver, invece, la risposta "UNSAT" arriva essenzialmente quando l'intero spazio $\mathbb{K}^r$ ($\mathbb{K} = \mathrm{GF}(2)$) è stato esaminato.

- Per le assegnazioni errate, che sono tutte tranne una, le basi di Gröbner sono quindi generalmente piú efficienti dei SAT solver. Abbiamo verificato questo in pratica nei nostri esperimenti.
- Risolvere un sistema algebrico mediante l'assegnazione esaustiva di un sottinsieme di variabili si chiama una *guess-and-determine strategy*.
- La sua complessità è generalmente il tempo di calcolo medio per una assegnazione errata, moltiplicato per (la metà de) il numero di assegnazioni, cioè, $(1/2)\ q^s$ se $\mathbb{K} = \mathrm{GF}(q)$ ed s è il numero di variabili da assegnare.
- È chiaro quindi che una ottimizzazione fondamentale è la scelta di queste variabili. La parallelizzazione del calcolo per differenti assegnazioni ha pure un impatto importante.

- Per le assegnazioni errate, che sono tutte tranne una, le basi di Gröbner sono quindi generalmente piú efficienti dei SAT solver. Abbiamo verificato questo in pratica nei nostri esperimenti.
- Risolvere un sistema algebrico mediante l'assegnazione esaustiva di un sottinsieme di variabili si chiama una *guess-and-determine strategy*.
- La sua complessità è generalmente il tempo di calcolo medio per una assegnazione errata, moltiplicato per (la metà de) il numero di assegnazioni, cioè, $(1/2)\ q^s$ se $\mathbb{K} = \mathrm{GF}(q)$ ed s è il numero di variabili da assegnare.
- È chiaro quindi che una ottimizzazione fondamentale è la scelta di queste variabili. La parallelizzazione del calcolo per differenti assegnazioni ha pure un impatto importante.

- Per le assegnazioni errate, che sono tutte tranne una, le basi di Gröbner sono quindi generalmente piú efficienti dei SAT solver. Abbiamo verificato questo in pratica nei nostri esperimenti.
- Risolvere un sistema algebrico mediante l'assegnazione esaustiva di un sottinsieme di variabili si chiama una *guess-and-determine strategy*.
- La sua complessità è generalmente il tempo di calcolo medio per una assegnazione errata, moltiplicato per (la metà de) il numero di assegnazioni, cioè, $(1/2)\, q^s$ se $\mathbb{K} = \mathrm{GF}(q)$ ed s è il numero di variabili da assegnare.
- È chiaro quindi che una ottimizzazione fondamentale è la scelta di queste variabili. La parallelizzazione del calcolo per differenti assegnazioni ha pure un impatto importante.

- Per le assegnazioni errate, che sono tutte tranne una, le basi di Gröbner sono quindi generalmente piú efficienti dei SAT solver. Abbiamo verificato questo in pratica nei nostri esperimenti.
- Risolvere un sistema algebrico mediante l'assegnazione esaustiva di un sottinsieme di variabili si chiama una *guess-and-determine strategy*.
- La sua complessità è generalmente il tempo di calcolo medio per una assegnazione errata, moltiplicato per (la metà de) il numero di assegnazioni, cioè, $(1/2)\, q^s$ se $\mathbb{K} = \mathrm{GF}(q)$ ed s è il numero di variabili da assegnare.
- È chiaro quindi che una ottimizzazione fondamentale è la scelta di queste variabili. La parallelizzazione del calcolo per differenti assegnazioni ha pure un impatto importante.

- Per il nostro attacco algebrico a Bivium, assegnamo in modo esaustivo 38 variabili. Grazie ad equazioni lineari presenti nel sistema, si ottiene di fatto l'assegnazione di 60 variabili sul totale di 177 variabili di stato.
- Per la risoluzione delle equazioni, si sono utilizzate un paio di implementazioni dell'algoritmo di Buchberger per le basi di Gröbner che si trovano nel sistema di calcolo simbolico SINGULAR.
- Abbiamo confrontato queste implementazioni con un paio di SAT solver molto efficienti: MiniSat e CryptoMinisat.
- La complessità del nostro attacco su un singolo processore è circa 2^{34} sec. L'attacco può essere naturalmente parallelizzato.

- Per il nostro attacco algebrico a Bivium, assegnamo in modo esaustivo 38 variabili. Grazie ad equazioni lineari presenti nel sistema, si ottiene di fatto l'assegnazione di 60 variabili sul totale di 177 variabili di stato.
- Per la risoluzione delle equazioni, si sono utilizzate un paio di implementazioni dell'algoritmo di Buchberger per le basi di Gröbner che si trovano nel sistema di calcolo simbolico SINGULAR.
- Abbiamo confrontato queste implementazioni con un paio di SAT solver molto efficienti: MiniSat e CryptoMinisat.
- La complessità del nostro attacco su un singolo processore è circa 2^{34} sec. L'attacco può essere naturalmente parallelizzato.

- Per il nostro attacco algebrico a Bivium, assegnamo in modo esaustivo 38 variabili. Grazie ad equazioni lineari presenti nel sistema, si ottiene di fatto l'assegnazione di 60 variabili sul totale di 177 variabili di stato.
- Per la risoluzione delle equazioni, si sono utilizzate un paio di implementazioni dell'algoritmo di Buchberger per le basi di Gröbner che si trovano nel sistema di calcolo simbolico SINGULAR.
- Abbiamo confrontato queste implementazioni con un paio di SAT solver molto efficienti: MiniSat e CryptoMinisat.
- La complessità del nostro attacco su un singolo processore è circa 2^{34} sec. L'attacco può essere naturalmente parallelizzato.

- Per il nostro attacco algebrico a Bivium, assegnamo in modo esaustivo 38 variabili. Grazie ad equazioni lineari presenti nel sistema, si ottiene di fatto l'assegnazione di 60 variabili sul totale di 177 variabili di stato.
- Per la risoluzione delle equazioni, si sono utilizzate un paio di implementazioni dell'algoritmo di Buchberger per le basi di Gröbner che si trovano nel sistema di calcolo simbolico SINGULAR.
- Abbiamo confrontato queste implementazioni con un paio di SAT solver molto efficienti: MiniSat e CryptoMinisat.
- La complessità del nostro attacco su un singolo processore è circa 2^{34} sec. L'attacco può essere naturalmente parallelizzato.

Assegnazione corretta

# ks bits	slimgb	std	MiniSat	CrMiniSat
180	175 ms	338 ms	15.41 s	14.66 s
185	162 ms	332 ms	12.68 s	13.25 s
190	119 ms	336 ms	10.71 s	11.85 s
195	151 ms	418 ms	10.95 s	14.39 s

Assegnazioni errate (media)

# ks bits	slimgb	std	MiniSat	CrMiniSat
180	173 ms	352 ms	52 s	25.52 s
185	170 ms	368 ms	34.62 s	19.61 s
190	119 ms	364 ms	37.28 s	19.72 s
195	129 ms	381 ms	37.41 s	18.85 s

Definizione

Un cifrario a blocchi alle differenze $\mathcal{C}$ *è per definizione un sistema invertibile e riducibile* (1) *insieme ad un intero* $T \geq 0$. *Se* (3) *è il sottosistema di* (1), *poniamo* $k = r_1 + \ldots + r_m$, $l = r_{m+1} + \ldots + r_n$ *(quindi* $k + l = r$*). Se* $(u(t), v(t)) \in \mathbb{K}^k \times \mathbb{K}^l$ *è lo stato al clock* t *di una* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$*, chiamiamo* $u(0)$ *la* chiave, $v(0)$ *il* plaintext *e* $v(T)$ *il* ciphertext di $(a_1, \ldots, a_n)$. *Inoltre, chiamiamo* $(u(T), v(T))$ *lo* stato finale di $(a_1, \ldots, a_n)$ *e* (3) *il* sottosistema di chiave del cifrario $\mathcal{C}$. *La* funzione di cifratura $E_{u(0)} : \mathbb{K}^l \to \mathbb{K}^l$ *è data dalla mappa* $v(0) \mapsto v(T)$*, dove la coppia* $(u(0), v(0))$ *varia nello spazio* $\mathbb{K}^k \times \mathbb{K}^l$.

Definizione

Un cifrario a blocchi alle differenze $\mathcal{C}$ *è per definizione un sistema invertibile e riducibile* (1) *insieme ad un intero* $T \geq 0$. *Se* (3) *è il sottosistema di* (1), *poniamo* $k = r_1 + \ldots + r_m, l = r_{m+1} + \ldots + r_n$ *(quindi* $k + l = r$*). Se* $(u(t), v(t)) \in \mathbb{K}^k \times \mathbb{K}^l$ *è lo stato al clock t di una* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$, *chiamiamo* $u(0)$ *la* chiave, $v(0)$ *il* plaintext *e* $v(T)$ *il* ciphertext di $(a_1, \ldots, a_n)$. *Inoltre, chiamiamo* $(u(T), v(T))$ *lo* stato finale di $(a_1, \ldots, a_n)$ *e* (3) *il* sottosistema di chiave del cifrario $\mathcal{C}$. *La* funzione di cifratura $E_{u(0)} : \mathbb{K}^l \to \mathbb{K}^l$ *è data dalla mappa* $v(0) \mapsto v(T)$, *dove la coppia* $(u(0), v(0))$ *varia nello spazio* $\mathbb{K}^k \times \mathbb{K}^l$.

Definizione

Un cifrario a blocchi alle differenze $\mathcal{C}$ *è per definizione un sistema invertibile e riducibile* (1) *insieme ad un intero* $T \geq 0$. *Se* (3) *è il sottosistema di* (1), *poniamo* $k = r_1 + \ldots + r_m, l = r_{m+1} + \ldots + r_n$ *(quindi* $k + l = r$*). Se* $(u(t), v(t)) \in \mathbb{K}^k \times \mathbb{K}^l$ *è lo stato al clock t di una* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$, *chiamiamo* $u(0)$ *la* chiave, $v(0)$ *il* plaintext *e* $v(T)$ *il* ciphertext di $(a_1, \ldots, a_n)$. *Inoltre, chiamiamo* $(u(T), v(T))$ *lo* stato finale di $(a_1, \ldots, a_n)$ *e* (3) *il* sottosistema di chiave del cifrario $\mathcal{C}$. La funzione di cifratura $E_{u(0)} : \mathbb{K}^l \to \mathbb{K}^l$ *è data dalla mappa* $v(0) \mapsto v(T)$, *dove la coppia* $(u(0), v(0))$ *varia nello spazio* $\mathbb{K}^k \times \mathbb{K}^l$.

Definizione

Un cifrario a blocchi alle differenze $\mathcal{C}$ *è per definizione un sistema invertibile e riducibile* (1) *insieme ad un intero* $T \geq 0$. *Se* (3) *è il sottosistema di* (1), *poniamo* $k = r_1 + \ldots + r_m, l = r_{m+1} + \ldots + r_n$ *(quindi* $k + l = r$*). Se* $(u(t), v(t)) \in \mathbb{K}^k \times \mathbb{K}^l$ *è lo stato al clock t di una* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$, *chiamiamo* $u(0)$ *la* chiave, $v(0)$ *il* plaintext *e* $v(T)$ *il* ciphertext di $(a_1, \ldots, a_n)$. *Inoltre, chiamiamo* $(u(T), v(T))$ *lo* stato finale di $(a_1, \ldots, a_n)$ *e* (3) *il* sottosistema di chiave del cifrario $\mathcal{C}$. *La* funzione di cifratura $E_{u(0)} : \mathbb{K}^l \to \mathbb{K}^l$ *è data dalla mappa* $v(0) \mapsto v(T)$, *dove la coppia* $(u(0), v(0))$ *varia nello spazio* $\mathbb{K}^k \times \mathbb{K}^l$.

Definizione

Sia (2) *il sistema inverso e* (3) *il sottosistema di chiave di* (1). *Se* $u(0)$ *è la chiave di una* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$ *di* (1), *possiamo calcolare* $u(T)$ *mediante* (3) *senza conoscere* $v(0)$. *Se abbiamo il ciphertext* $v(T)$, *otteniamo allora lo stato finale* $(u(T), v(T))$ *di* $(a_1, \ldots, a_n)$. *Il sistema inverso* (2) *è capace quindi di calcolare lo stato iniziale* $(u(0), v(0))$ *di* $(a_1, \ldots, a_n)$ *ed in particolare il plaintext* $v(0)$. *In altri termini, la funzione di decifratura* $D_{u(0)} : \mathbb{K}^l \to \mathbb{K}^l$ *si ottiene come la mappa* $v(T) \to v(0)$ *che è calcolabile mediante i sistemi* (2) *e* (3).

Definizione

Sia (2) *il sistema inverso e* (3) *il sottosistema di chiave di* (1)*. Se* $u(0)$ *è la chiave di una* $\mathbb{K}$*-soluzione* $(a_1, \ldots, a_n)$ *di* (1)*, possiamo calcolare* $u(T)$ *mediante* (3) *senza conoscere* $v(0)$*. Se abbiamo il ciphertext* $v(T)$*, otteniamo allora lo stato finale* $(u(T), v(T))$ *di* $(a_1, \ldots, a_n)$*. Il sistema inverso* (2) *è capace quindi di calcolare lo stato iniziale* $(u(0), v(0))$ *di* $(a_1, \ldots, a_n)$ *ed in particolare il plaintext* $v(0)$*. In altri termini, la funzione di decifratura* $D_{u(0)} : \mathbb{K}^l \to \mathbb{K}^l$ *si ottiene come la mappa* $v(T) \to v(0)$ *che è calcolabile mediante i sistemi* (2) *e* (3)*.*

Definizione

Sia (2) il sistema inverso e (3) il sottosistema di chiave di (1). Se $u(0)$ è la chiave di una $\mathbb{K}$-soluzione $(a_1, \ldots, a_n)$ di (1), possiamo calcolare $u(T)$ mediante (3) senza conoscere $v(0)$. Se abbiamo il ciphertext $v(T)$, otteniamo allora lo stato finale $(u(T), v(T))$ di $(a_1, \ldots, a_n)$. Il sistema inverso (2) è capace quindi di calcolare lo stato iniziale $(u(0), v(0))$ di $(a_1, \ldots, a_n)$ ed in particolare il plaintext $v(0)$. In altri termini, la funzione di decifratura $D_{u(0)} : \mathbb{K}^l \to \mathbb{K}^l$ si ottiene come la mappa $v(T) \to v(0)$ che è calcolabile mediante i sistemi (2) e (3).

Definizione

Sia $\mathcal{C}$ un cifrario a blocchi alle differenze e sia $(u(t), v(t)) \in \mathbb{K}^k \times \mathbb{K}^l$ *lo stato al clock t di una* $\mathbb{K}$*-soluzione di* (1). *Denotato*

$$v(t) = (a_{m+1}(t), \ldots, a_{m+1}(t + r_{m+1} - 1), \ldots, a_n(t), \ldots, a_n(t + r_n - 1)),$$

si consideri il corrispondente ideale lineare

$$J(t) = \sum_{m+1 \leq i \leq n} \langle x_i(t) - a_i(t), \ldots, x_i(t + r_i - 1) - a_i(t + r_i - 1) \rangle \subset R.$$

Infine, poniamo $J = J(0) + J(T)$. *Un* attacco algebrico a $\mathcal{C}$ mediante la coppia plaintext-ciphertext $(v(0), v(T))$ *consiste nel calcolare le* $\mathbb{K}$*-soluzioni* $(a_1, \ldots, a_n)$ *tali che* $(a_1, \ldots, a_n) \in V_{\mathbb{K}}(J)$. *Se* $I = \langle x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n \rangle_\sigma \subset R$ *vogliamo, cioè, calcolare* $V_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(I) \cap V_{\mathbb{K}}(J)$.

Definizione

Sia $\mathcal{C}$ un cifrario a blocchi alle differenze e sia $(u(t), v(t)) \in \mathbb{K}^k \times \mathbb{K}^l$ lo stato al clock t di una $\mathbb{K}$-soluzione di (1). Denotato

$$v(t) = (a_{m+1}(t), \ldots, a_{m+1}(t + r_{m+1} - 1), \ldots, a_n(t), \ldots, a_n(t + r_n - 1)),$$

si consideri il corrispondente ideale lineare

$$J(t) = \sum_{m+1 \leq i \leq n} \langle x_i(t) - a_i(t), \ldots, x_i(t + r_i - 1) - a_i(t + r_i - 1)\rangle \subset R.$$

Infine, poniamo $J = J(0) + J(T)$. Un attacco algebrico a $\mathcal{C}$ mediante la coppia plaintext-ciphertext $(v(0), v(T))$ *consiste nel calcolare le $\mathbb{K}$-soluzioni $(a_1, \ldots, a_n)$ tali che $(a_1, \ldots, a_n) \in V_{\mathbb{K}}(J)$. Se $I = \langle x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n\rangle_\sigma \subset R$ vogliamo, cioè, calcolare $V_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(I) \cap V_{\mathbb{K}}(J)$.*

Definizione

Sia $\mathcal{C}$ *un cifrario a blocchi alle differenze e sia* $(u(t), v(t)) \in \mathbb{K}^k \times \mathbb{K}^l$ *lo stato al clock t di una* $\mathbb{K}$*-soluzione di* (1)*. Denotato*

$$v(t) = (a_{m+1}(t), \ldots, a_{m+1}(t + r_{m+1} - 1), \ldots, a_n(t), \ldots, a_n(t + r_n - 1)),$$

si consideri il corrispondente ideale lineare

$$J(t) = \sum_{m+1 \leq i \leq n} \langle x_i(t) - a_i(t), \ldots, x_i(t + r_i - 1) - a_i(t + r_i - 1)\rangle \subset R.$$

Infine, poniamo $J = J(0) + J(T)$*. Un* attacco algebrico a $\mathcal{C}$ mediante la coppia plaintext-ciphertext $(v(0), v(T))$ *consiste nel calcolare le* $\mathbb{K}$*-soluzioni* $(a_1, \ldots, a_n)$ *tali che* $(a_1, \ldots, a_n) \in V_{\mathbb{K}}(J)$*. Se* $I = \langle x_1(r_1) - f_1, \ldots, x_n(r_n) - f_n\rangle_\sigma \subset R$ *vogliamo, cioè, calcolare* $V_{\mathbb{K}}(I + J) = V_{\mathbb{K}}(I) \cap V_{\mathbb{K}}(J)$*.*