

# Indice.

<b>1</b>	<b>Teoria degli automi: introduzione e concetti base</b>	<b>2</b>
1.1	Concetti centrali . . . . .	2
<b>2</b>	<b>Automi DFA e NFA</b>	<b>3</b>
2.1	I DFA . . . . .	3
2.2	Gli NFA . . . . .	4
2.3	DFA e NFA: linguaggi e proprietà dei linguaggi . . . .	5
2.4	Equivalenza tra NFA e DFA . . . . .	6
2.5	Esercizi su DFA e NFA . . . . .	8
<b>3</b>	<b>Proprietà dei linguaggi REC</b>	<b>9</b>
3.1	Chiusura per intersezione . . . . .	9
3.2	Chiusura per unione . . . . .	10

## – 1 – Teoria degli automi: introduzione e concetti base.

Si consideri il caso di un interruttore. Grazie agli automi è possibile rappresentare facilmente il passaggio tra i due stati come mostrato in 1.

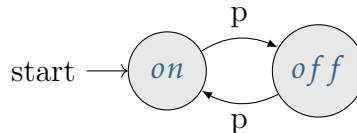


Figura 1: Automa rappresentante uno switch.

Dando una breve definizione di automa: questi è un sistema automatico, rappresentato da un grafo i cui nodi rappresentano gli stati e gli archi le transizioni tra stati.

L'utilizzo degli automi è da ricercare nello studio dei limiti computazionali, cui si legano

1. lo studio della decidibilità, che stabilisce cosa possa fare un computer in assoluto;
2. lo studio della trattabilità che stabilisce cosa possa fare un computer efficientemente.

Agli automi sono inoltre legati due importanti nozioni, quali le grammatiche e le espressioni regolari, che si discuteranno nelle successive sezioni.

### – 1.1 – Concetti centrali.

Concetti centrali della teoria degli automi sono gli alfabeti, le stringhe e i linguaggi.

- *Gli alfabeti*: si definisce alfabeto  $\Sigma$  un insieme finito di caratteri.
- *Le Stringhe*: dato  $\Sigma$  un alfabeto, si definisce stringa  $\omega$  una sequenza di simboli scelti dall'alfabeto.

Caso particolare è la stringa vuota  $\varepsilon$ : una stringa composta da zero simboli.

Data  $\omega$  una stringa, di questa è possibile stabilirne la lunghezza: ossia il numero di caratteri di cui si compone.

Infine, considerate  $\omega_1 = a_1 \cdots a_k$  e  $\omega_2 = b_1 \cdots b_j$  due stringhe, si definisce  $\omega_1 \circ \omega_2 = \omega_1 \omega_2 = a_1 \cdots a_k b_1 \cdots b_j$  concatenazione di  $\omega_1$  e  $\omega_2$ .

- *I Linguaggi*: dato  $\Sigma$  un alfabeto, si definisce linguaggio  $L$  su  $\Sigma$  un sottoinsieme delle stringhe ottenibili con l'alfabeto.

## – 2 – Automi DFA e NFA.

Come anticipato in sezione (1): un automa è un sistema automatico, rappresentato da un grafo.

Si tenga presente che esistono due classi di automi

- deterministici o DFA;
- non deterministici o NFA.

### – 2.1 – I DFA.

**Definizione:** Si definisce  $A = (Q, \Sigma, \delta, q_0, F)$  DFA, ove

- $Q$  rappresenta l'insieme di stati dell'automa;
- $\Sigma$  è l'alfabeto utilizzato dall'automa;
- $\delta$  definisce le transizioni tra gli stati;
- $q_0$  indica lo stato iniziale;
- $F$  definisce l'insieme di stati finali;

se considerata  $\delta$ , per ciascun simbolo dell'alfabeto e per ciascuno stato esiste un'unica transizione per quel carattere.

#### – 2.1.1 – Funzione di transizione e funzione di transizione estesa.

Dato un automa  $A$ , la funzione di transizione  $\delta$  stabilisce il comportamento dell'automa in ogni suo stato, per ogni simbolo dell'alfabeto.

**Esempio:** Sia considerato l'automa di *Figura* (1).

La funzione di transizione dello stesso, definisce le seguenti transizioni

$$\begin{aligned}\delta(on, p) &= (off) \\ \delta(off, p) &= (on)\end{aligned}$$

ossia: letto  $p$  dallo stato  $on$  passa allo stato  $off$ , da questi letto  $p$  passa a  $on$ .

**Definizione:** Sia  $\omega = a_1 \cdots a_n$  una stringa e  $\delta$  la funzione di transizione di un dato DFA: si definisce funzione di transizione estesa  $\delta^*$  la funzione che, letta  $\omega$  a partire da  $q_0$ , stabilisce lo stato di arrivo  $q_f$ . Cioè

$$\delta^*(q_0, \omega) = (q_f)$$

**Osservazione:** Dato un automa, la funzione di transizione estesa  $\delta^*$ , può essere intesa come la sequenziale applicazione della funzione di transizione  $\delta$ , per ogni simbolo in  $\omega$  a partire dallo stato  $q_0$ .

## – 2.2 – Gli NFA.

**Definizione:** Si definisce  $A = (Q, \Sigma, \delta, q_0, F)$  NFA, ove

- $Q$  rappresenta l'insieme di stati dell'automa;
- $\Sigma$  è l'alfabeto utilizzato dall'automa;
- $\delta$  definisce le transizioni tra gli stati;
- $q_0$  indica lo stato iniziale;
- $F$  definisce l'insieme di stati finali;

se considerata  $\delta$ , per ciascun simbolo dell'alfabeto e per almeno uno stato esistono più transizioni per quel carattere.

**Esempio:** Sia considerato l'automa in *Figura (1)*, questi può essere rappresentato come NFA dall'automa in *Figura (2)*.

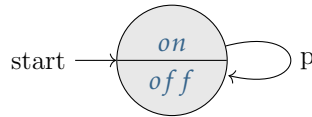


Figura 2: Automa rappresentante uno switch come NFA.

### – 2.2.1 – Funzione di transizione estesa.

**Definizione:** Sia  $\omega = a_1 \cdots a_n$  una stringa e  $\delta$  la funzione di transizione di un dato NFA: si definisce funzione di transizione estesa  $\delta^*$  la funzione che, letta  $\omega$  a partire da  $q_0$ , stabilisce lo stato di arrivo  $q_f$ .

$$\text{Per induzione si ha } \begin{cases} \delta^*(q_0, \varepsilon) = \{q_0\} & \text{base} \\ \delta^*(q_0, \omega) = \bigcup_{q_x \in \delta^*(q_0, \omega)} \delta(q_x, a) \end{cases}$$

### – 2.3 – DFA e NFA: linguaggi e proprietà dei linguaggi.

**Definizione:** Sia  $A$  un automa. Si definisce linguaggio di  $A$ ,  $L(A)$ , l'insieme delle stringhe  $\omega$  che accettate da  $A$ . Cioè

$$\begin{cases} L(A) = \{\omega : \delta^*(q_0, \omega) \in F\} & \text{se } A \text{ è un DFA} \\ L(A) = \{\omega : \delta^*(q_0, \omega) \cap F \neq \emptyset\} & \text{se } A \text{ è un NFA} \end{cases}$$

#### – 2.3.1 – Proprietà dei linguaggi.

Sia  $L$  il linguaggio riconosciuto da un automa; su di questi è possibile applicare le seguenti operazioni.

- *Potenza  $n$ -sima:* si intende la concatenazione di  $L$  un certo numero  $n$  di volte.

**Esempio:** Sia  $L = \{\omega : \omega \in \Sigma = \{a, b\}\}$ , sia  $n = 2$ . Segue

$$L^2 = L \circ L = \{aaaa, aaab, aabb, aaba, abaa, abab, abbb, abba, \dots\}$$

**Osservazione:** Se  $n = 0$  si ha che  $L^0 = \{\varepsilon\}$ .

- *Stella di Kleene:* rappresenta l'unione di tutte le potenze di  $L$ . Cioè

$$L^* = L^0 \cup L^1 \cup L^2 \cup \dots$$

**Osservazione:** Se  $L = \emptyset$  allora  $L^* = \{\varepsilon\}$ .

- *Croce:* indica l'unione di tutte le potenze di  $L$ , meno  $L^0$ . Cioè

$$L^+ = L^1 \cup L^2 \cup \dots$$

Vale dunque

$$L^+ = L \circ L^*$$

#### – 2.3.2 – Linguaggio universale e complemento.

**Definizione:** Sia  $\Sigma$  un alfabeto. Si definisce linguaggio universale  $\Sigma^*$ , l'insieme di tutte le parole applicando all'alfabeto Kleene.

**Definizione:** Sia  $L$  un linguaggio su un alfabeto  $\Sigma$ . Si definisce complemento di  $L$ ,  $L^C$ , l'insieme di stringhe che appartengono a  $\Sigma^*$  ma non a  $L$ .

## – 2.4 – Equivalenza tra NFA e DFA.

Si potrebbe erroneamente pensare che NFA e DFA riconoscano linguaggi diversi, ma si dimostra che non è così.

Prima di dimostrare il teorema di equivalenza tra NFA e DFA, è necessario parlare di *subset construction*.

### – 2.4.1 – Subset construction.

Sia  $N = (Q = \{q_0, q_1, \dots, q_k\}, \Sigma, \delta_N, q_0, F_N)$  un NFA.

Per ogni  $q_i \in Q, i = 0, 1, \dots, k$  e per ogni  $x \in \Sigma$ , si definisco gli stati di un DFA  $D$ , dati dall'insieme degli stati definite da  $\delta_N$ . Inoltre, uno stato di  $D$  sarà accettante se, almeno uno, degli sti di  $N$  da cui è definito è accettante. In fine le transizioni di  $D$ , sono analoghe a quelle di  $N$ .

**Esempio:** Sia considerato l'NFA di *Figura (3)*

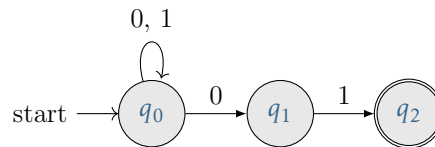


Figura 3: Automa per il linguaggio delle parole che terminano con 01.

Considerando  $\delta$  si ha

$$\delta(q_0, 0) = (q_0)$$

$$\delta(q_0, 1) = (q_0)$$

$$\delta(q_0, 0) = (q_1)$$

$$\delta(q_1, 1) = (q_2)$$

da ciò segue l'automa di *Figura (4)*.

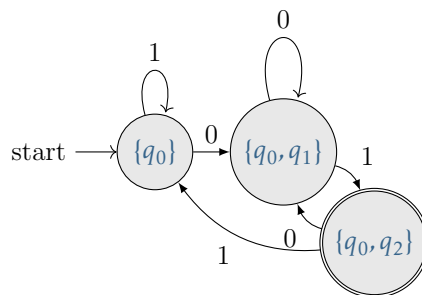


Figura 4: Subset construction dell'automa di *Figura (3)*.

Poiché gli stati  $\{q_1\}, \{q_2\}$  sono inaccessibili da  $\{q_0\}$ , questi sono stati trascurati.

## – 2.4.2 – Teorema di equivalenza tra NFA e DFA.

**Teorema 2.1.**

Sia  $D$  un DFA ottenuto per subset construction da un NFA  $N$ , allora  $L(D) = L(N)$ .

**Dimostrazione:** Per dimostrare che  $L(D) = L(N)$ , si procederà per induzione su  $|\omega|$  che

$$\delta_D^*(\{q_0\}, \omega) = \delta_N^*(q_0, \omega) \quad (1)$$

**Base:** Sia  $|\omega| = 0$ , ossia  $\omega = \varepsilon$ .

Per definizione di  $\delta^*$ , segue che  $\delta_D^*(\{q_0\}, \omega) = \delta_N^*(q_0, \omega) = \{q_0\}$ .

**Induzione:** Supposto che quanto detto finora sia vero per  $|\omega| = n$ , si consideri  $|\omega| = n + 1$ . Sia posta  $\omega = xa$ , ove  $a$  è l'ultimo carattere della stringa.

Per ipotesi induttiva  $\delta_D^*(\{q_0\}, x) = \delta_N^*(q_0, x) = \{p_1, \dots, p_k\}$ , segue dalla definizione induttiva di  $\delta^*$  per gli NFA

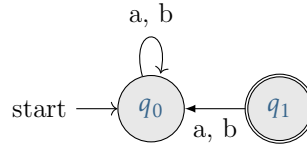
$$\delta_N^*(q_0, \omega) = \bigcup_{i=1}^k \delta_N(p_i, a)$$

ma  $\bigcup_{i=1}^k \delta_N(p_i, a) = \delta_D(\{p_1, \dots, p_k\}, a)$ , segue pertanto

$$\delta_D^*(\{q_0\}, \omega) = \delta_D(\{p_1, \dots, p_k\}, a) = \bigcup_{i=1}^k \delta_N(p_i, a)$$

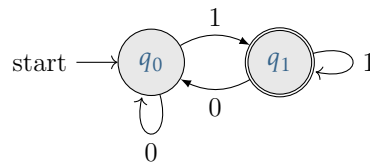
## – 2.5 – Esercizi su DFA e NFA.

1. Sia  $L = \{\}$  definito su  $\Sigma = \{a, b\}$ . Si realizzi un automa che lo riconosca.

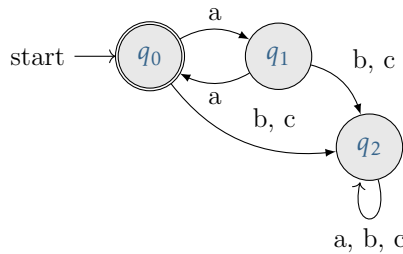


**Nota:** Soluzione al problema è un qualsiasi DFA, o NFA che sia, al cui stato accettante non è possibile accedere.

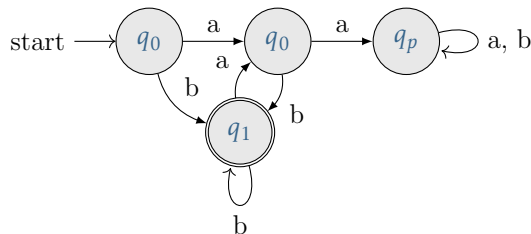
2. Sia  $L = \{a^{2^n} : n \geq 0\}$  definito su  $\Sigma = \{a, b, c\}$ . Si realizzi un automa che lo riconosca.



3. Sia  $L = \{\omega : \omega = \Sigma^*1\}$  definito su  $\Sigma = \{0, 1\}$ . Si realizzi un automa che lo riconosca.



4. Sia  $L = \{\omega : \omega = (\Sigma^*aa\Sigma^*)^C\}$ . Si realizzi un automa che lo riconosca.





### – 3 – Proprietà dei linguaggi REC.

**Definizione:** Sia  $L$  un linguaggio. Questo si definisce regolare se accettato da un DFA.

I linguaggi regolari sono chiusi, cioè rimangono regolari, rispetto operazioni quali

- *intersezione*;
- *unione*;
- *complemento*;
- *Kleene*;
- *croce*.

#### – 3.1 – Chiusura per intersezione.

**Teorema 3.1.**

Siano  $L_1$  e  $L_2$  linguaggi REC. Allora  $L = L_1 \cap L_2$  è REC.

**Dimostrazione:** Sia  $A_1$  un automa che riconosce  $L_1$ , sia  $A_2$  un automa che riconosce  $L_2$ .

$$A_1 = (Q_1, \Sigma, \delta_1, q_{0_1}, F_1) \quad A_2 = (Q_2, \Sigma, \delta_2, q_{0_2}, F_2)$$

Sia  $A = (Q, \Sigma, \delta, q_0, F)$  un automa che riconosce  $L$ . Ponendo

- $Q = \{(q_1, q_2) : q_1 \in Q_1 \wedge q_2 \in Q_2\}$  o analogamente  $Q = Q_1 \times Q_2$ ;
- $q_0 = (q_{0_1}, q_{0_2})$ ;
- $\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$  per ogni  $a$  tale che la transizione sia definita sia in  $A_1$  che  $A_2$ ;
- $F = \{(q_1, q_2) : q_1 \in F_1 \wedge q_2 \in F_2\}$  o analogamente  $F = F_1 \times F_2$ .

Sia  $\omega \in L$ , segue

$$\begin{aligned} \omega \in L &\iff \omega \in L_1 \wedge \omega \in L_2 \\ &\iff \delta_1^*(q_{0_1}, \omega) \in F_1 \wedge \delta_2^*(q_{0_2}, \omega) \in F_2 \\ &\iff \delta^*((q_{0_1}, q_{0_2}), \omega) \in F \end{aligned}$$

### – 3.2 – Chiusura per unione.

**Teorema 3.2.**

Siano  $L_1$  e  $L_2$  linguaggi REC. Allora  $L = L_1 \cup L_2$  è REC.

**Dimostrazione:** Sia  $A_1$  un automa che riconosce  $L_1$ , sia  $A_2$  un automa che riconosce  $L_2$ .

$$A_1 = (Q_1, \Sigma, \delta_1, q_{0_1}, F_1) \quad A_2 = (Q_2, \Sigma, \delta_2, q_{0_2}, F_2)$$

Sia  $A = (Q, \Sigma, \delta, q_0, F)$  un automa che riconosce  $L$ . Ponendo

- $Q = \{(q_1, q_2) : q_1 \in Q_1 \wedge q_2 \in Q_2\}$  o analogamente  $Q = Q_1 \times Q_2$ ;
- $q_0 = (q_{0_1}, q_{0_2})$ ;
- $\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$  per ogni  $a$  tale che la transizione sia definita sia in  $A_1$  che  $A_2$ ;
- $F = \{(q_1, q_2) : q_1 \in F_1 \wedge q_2 \in F_2\}$ .

Sia  $\omega \in L$ , segue

$$\begin{aligned} \omega \in L &\iff \omega \in L_1 \vee \omega \in L_2 \\ &\iff \delta_1^*(q_{0_1}, \omega) \in F_1 \vee \delta_2^*(q_{0_2}, \omega) \in F_2 \\ &\iff \delta^*((q_{0_1}, q_{0_2}), \omega) \in F \end{aligned}$$

**Nota:** Similarmente si dimostra anche la chiusura per complemento, Kleene, croce.