# Indice.

# – 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.

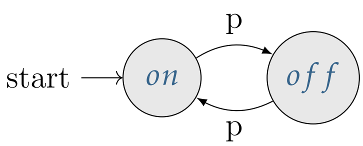


Figura 1: Automa rappresentante uno switch.

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 compute 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 stringe 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 trasizione 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

$$\delta(on, p) = (off)$$
$$\delta(off, p) = (on)$$

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).

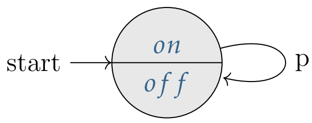


Figura 2: Automa rappresentante uno switch come NFA.

### – 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 $q_f$.

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

## – 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

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

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

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

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

  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 $N = (Q = \{q_0, q_1, \ldots, q_k\}, \Sigma, \delta_N, q_0, F_N)$ un NFA.
Per ogni $q_i \in Q, i = 0, 1, \ldots, k$ 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 $D$ sara 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)

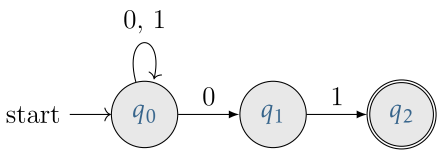


Figura 3: Automa per il linguaggio delle parole che terminano con 01.

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).

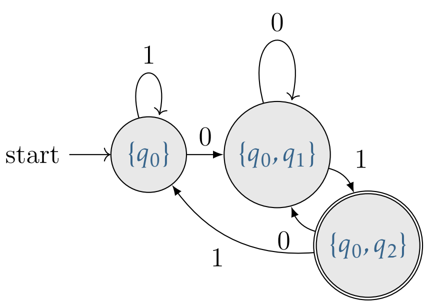


Figura 4: Subset construction dell'automa di *Figura* (3).

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 2.4.1.**
*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

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

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

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