



UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
Corso di Laurea Magistrale in Ingegneria dell'Automazione

TESI DI LAUREA

Analisi e controllo delle Boolean control networks

Laureando:
Riccardo Sterbizzi

Relatore:
Prof. Maria Elena Valcher

Contro-relatore:
Prof. Ettore Fornasini

Indice

| | |
|---|-----------|
| Introduzione | 1 |
| 1 Preliminari matematici | 5 |
| 1.1 Variabili e funzioni logiche | 5 |
| 1.2 Il prodotto semi-tensore di matrici | 7 |
| 1.3 Logica in forma matriciale | 9 |
| 1.4 Sistemi di equazioni logiche | 13 |
| 1.5 Sistemi positivi e matrici non negative | 19 |
| 1.5.1 Matrici di permutazione | 20 |
| 1.5.2 Classificazione delle matrici non negative | 21 |
| 2 Boolean Networks | 23 |
| 2.1 Dinamica di una Boolean Network | 23 |
| 2.1.1 Rappresentazione in spazio di stato di una BN | 23 |
| 2.2 Punti fissi e cicli limite | 26 |
| 2.3 Rappresentazione grafica | 30 |
| 2.3.1 Accessibilità e comunicazione | 32 |
| 2.3.2 Tipi di classi di comunicazione | 33 |
| 2.4 Stabilità | 35 |
| 2.5 Domini di attrazione e forma normale di L | 37 |
| 2.6 Forma normale per classi della matrice L | 43 |

| | | |
|----------|---|-----------|
| 3 | Boolean Control Networks | 47 |
| 3.1 | Dinamica e forma algebrica di una BCN | 47 |
| 3.1.1 | La BCN come switched system | 49 |
| 3.1.2 | La matrice L_{tot} | 51 |
| 3.2 | Rappresentazione grafica | 52 |
| 3.3 | Controllabilità | 55 |
| 3.3.1 | Definizioni di controllabilità | 55 |
| 3.3.2 | Criteri di controllabilità | 56 |
| 3.3.3 | Controllabilità globale forte | 62 |
| 3.4 | Stabilizzabilità | 64 |
| 3.4.1 | Punti di equilibrio ad ingresso costante | 64 |
| 3.4.2 | Criteri per la globale stabilizzabilità | 66 |
| 3.5 | Sequenze di ingresso | 67 |
| 3.5.1 | Matrice di controllo e sue proprietà | 68 |
| 3.5.2 | Control design | 70 |
| 3.5.3 | Controllo in retroazione | 72 |
| 4 | Applicazioni | 77 |
| 4.1 | Analisi del ciclo cellulare nel lievito a fissione | 77 |
| 4.1.1 | Il ciclo cellulare nel lievito a fissione | 78 |
| 4.1.2 | Rappresentazione del ciclo cellulare tramite Boolean network | 78 |
| 4.1.3 | Simulazione del ciclo cellulare del lievito a fissione | 82 |
| 4.2 | Il controllo del ciclo cellulare dei mammiferi | 84 |
| 4.2.1 | Modellizzazione del ciclo cellulare dei mammiferi tramite una BCN | 84 |
| 4.2.2 | Ciclo cellulare in assenza e in presenza di CycD | 87 |
| A | Codici MATLAB | 91 |
| A.1 | Codici per il ciclo cellulare nel lievito a fissione | 91 |
| A.1.1 | Fission_yeast.m | 91 |

| | |
|---|------------|
| <i>INDICE</i> | III |
| A.1.2 Altre funzioni utilizzate | 95 |
| A.2 Codici per il ciclo cellulare dei mammiferi | 97 |
| A.2.1 Mammalian_cell_cycle_cycd0.m | 98 |
| A.2.2 Mammalian_cell_cycle_cycd1.m | 99 |
| Conclusioni | 103 |

Notazioni

| | |
|----------------------------|--|
| \mathcal{B} | Insieme $\{1, 0\}$ |
| \neg | Operatore logico di negazione |
| \wedge | Operatore logico di congiunzione (AND) |
| \vee | Operatore logico di disgiunzione (OR) |
| \rightarrow | Operatore logico di implicazione logica |
| \leftrightarrow | Operatore logico di equivalenza logica |
| $\bar{\vee}$ | Operatore logico di disgiunzione esclusiva (XOR) |
| $[a]$ | Parte intera di a |
| \otimes | Prodotto di Kronecker |
| \otimes | Prodotto semi-tensore |
| δ_n^k | k -esima colonna della matrice identità I_n |
| Δ^k | Insieme $\{\delta_k^i : i = 1, \dots, k\}$ |
| $\text{Col}_i(A)$ | i -esima colonna della matrice A |
| $\text{Col}(A)$ | insieme delle colonne della matrice A |
| $\text{Row}_i(A)$ | i -esima riga della matrice A |
| $\text{Row}(A)$ | insieme delle righe della matrice A |
| $\mathcal{B}^{n \times m}$ | Insieme delle matrici Booleane $n \times m$ |
| $\mathcal{L}^{n \times m}$ | Insieme delle matrici logiche $n \times m$ |

Introduzione

Le Boolean networks, o reti Booleane, sono state definite da Kauffman nel 1969 in [25] per poter modellizzare reti genetiche e cellulari e in questi ultimi anni hanno avuto un'importanza sempre crescente.

Negli anni sessanta Jacob e Monod [24], nei loro studi, evidenziarono come ogni cellula contenesse un certo numero di geni regolatori capaci di attivarsi e disattivarsi reciprocamente; questo risultato portò ad una maggiore attenzione al rapporto e all'interazione dei geni.

Kauffman [25] mise in evidenza che gli stati principali di un gene sono due: “in trascrizione” e “quiescente”; si ha quindi che i geni sono approssimabili con variabili dotate solo di due livelli: acceso (ON) e spento (OFF), e per questo possono essere modellizzati come variabili Booleane ($1 = \text{ON}$, $0 = \text{OFF}$). Dato che il comportamento di un gene viene influenzato dallo stato di quelli vicini, Kauffman introdusse per primo il concetto di Boolean network per poter studiare il comportamento di un numero elevato di geni, visti come nodi di una rete, interagenti tra di loro. L'obiettivo era poterne analizzare caratteristiche quali ordine e stabilità.

Dagli studi di Kauffman si sono sviluppati due filoni differenti di ricerca: (1) l'analisi delle reti Booleane, viste come sistemi complessi, mediante strumenti matematici o della fisica statistica e (2) la modellizzazione/approssimazione di processi attraverso Boolean networks. Il primo filone di ricerca ha portato numerosi studi tra i quali [3],[27],[28],[33] fino ai recenti risultati di Cheng et al. [9] che hanno permesso di esprimere una Boolean Network come un sistema lineare a tempo discreto e quindi di poterla studiare avvalendosi degli strumenti della Teoria dei Sistemi. Per quanto riguarda il secondo campo di ricerca, il principale ambito applicativo delle Boolean networks è rappresentato dalla biologia. Sono infatti numerosi i processi biologici che sono stati approssimati con Boolean networks. Ad esempio: reti per il ciclo cellulare dei lieviti [30], reti di trascrizione dei lieviti [26], reti riguardanti il *segment polarity gene* nelle larve di *Drosophila melanogaster* [2], reti di geni regolatori nei fiori (Aradopsis) [15].

Ulteriori utilizzi delle Boolean networks si sono avuti nell'ambito della modellistica di processi più generali come la differenziazione cellulare per studiare l'evoluzione

dei primi microorganismi [31], la via metabolica (pathway) tra due neurotrasmettitori [21] e anche per modellizzare la carcinosi e confrontare i possibili interventi medici [39]. Nonostante le Boolean networks siano reti molto semplici possono prestarsi anche a modellizzazioni più complesse riguardanti i sistemi ecologici [4], poiché possono essere utilizzate per studiare le caratteristiche principali quali la stazionarietà di uno status biologico.

Altri campi di applicazioni delle Boolean networks riguardano: reti neuronali artificiali semplici (a due soli valori naturalmente) [22], reti fisiche [34] e reazioni biochimiche [23].

Le reti Booleane vengono inoltre utilizzate per studi di tipo sociale come generazione del consenso [20]. In questo caso i nodi della rete rappresentano i singoli individui (con due possibili stati e.g. accordo/disaccordo) e le interazioni tra i nodi rappresentano le relazioni sociali. Modelli di questo tipo servono a studiare il ruolo della cooperazione tra diversi individui e tematiche quali il ruolo del leader e l'influenza dei mass-media.

Introducendo un controllo esterno (sempre Booleano) alla dinamica di una rete Booleana si ottiene una Boolean control network. Lo studio delle Boolean control networks ha come obiettivo trovare una sequenza di ingresso che permetta di portare i valori dei nodi della rete da una configurazione ad un'altra desiderata. In ambito medico se ne ha un'immediata applicazione; queste analisi infatti possono essere utili per progettare e analizzare differenti interventi terapeutici quali, ad esempio, scegliere se somministrare o meno una certa medicina e in quale momento agire. Il problema del controllo di una rete Booleana è in generale NP-Hard come dimostrato da Akutsu et al. in [1] ed è stato affrontato solo recentemente. In [13] e in [32] il controllo di una BCN, visto come un problema di controllo ottimo, è stato affrontato mediante la programmazione dinamica, mentre in [9] è stato studiato mediante un approccio nello spazio di stato.

In questa tesi di laurea, seguendo l'approccio algebrico e la descrizione delle Boolean networks nello spazio di stato operata in [9], si sono affrontate principalmente tre tematiche: stabilità, stabilizzabilità e controllabilità.

In particolare la tesi è suddivisa nei seguenti capitoli:

- Nel Capitolo 1 è riportata una breve introduzione alla logica matematica, al prodotto semitensore di matrici e sue proprietà e ai sistemi positivi.
- Nel Capitolo 2 sono studiate e analizzate le proprietà di una Boolean network: la sua forma in spazio di stato, lo studio degli attrattori e dei propri bacini di attrazione, la stabilità e la forma normale per classi.

- Nel Capitolo 3 vengono introdotti gli ingressi e si sono studiate la controllabilità, la stabilizzazione e come trovare una sequenza di ingresso per una Boolean control network.
- Nel Capitolo 4 sono presentate due applicazioni per le reti Boolean: l'analisi del ciclo cellulare nel lievito a fissione e il controllo del ciclo cellulare dei mammiferi.
- Nell'Appendice sono contenuti infine i codici MATLAB utilizzati per ottenere i risultati riportati nelle applicazioni.

Capitolo 1

Preliminari matematici

Le Boolean networks e le Boolean control networks sono reti composte da un certo numero di nodi che possono assumere due soli valori, 1 e 0, e sono esprimibili quindi matematicamente come variabili logiche o Booleane. Essendo la dinamica dei nodi influenzata dal valore di quelli adiacenti e da eventuali ingressi mediante relazioni di tipo logico, è dunque necessario presentare brevemente i concetti base di logica matematica per poter definire rigorosamente una Boolean network. In aggiunta a questa breve trattazione, in questo Capitolo verrà anche introdotto un nuovo tipo di prodotto (il prodotto semi-tensore presentato in [9]) grazie al quale è possibile esprimere un sistema di equazioni logiche in forma algebrica. Dato che in questo modo una Boolean network può essere equivalentemente descritta da un sistema lineare positivo a tempo discreto, saranno riportate anche le principali proprietà di questi sistemi e delle matrici che li compongono.

1.1 Variabili e funzioni logiche

Si consideri l'insieme:

$$\mathcal{B} = \{1, 0\}.$$

Definizione 1.1.1. Una variabile α è una variabile logica (o variabile Booleana) se assume valori in \mathcal{B} , i.e. $\alpha \in \mathcal{B}$.

Le operazioni tra due variabili logiche α e β sono espresse mediante connettori logici; le principali operazioni sono tre: NOT, AND e OR:

- NOT: l'operazione di negazione di α è indicata con $\neg\alpha$. Si ha che $\neg\alpha = 1$ se e solo se $\alpha = 0$ e viceversa. Questa operazione è detta anche complementazione poiché trasforma una variabile nel suo complemento (1 in 0 e 0 in 1).
- AND: l'operazione di congiunzione tra α e β è indicata con $\alpha \wedge \beta$. La variabile $\alpha \wedge \beta$ è pari a 1 se e solo se $\alpha = 1$ e $\beta = 1$.
- OR: l'operazione di disgiunzione tra α e β è indicata con $\alpha \vee \beta$. La variabile $\alpha \vee \beta$ è pari a 1 se e solo se $\alpha = 1$ o $\beta = 1$.

Definizione 1.1.2. Una funzione logica F è una mappa che coinvolge variabili Booleane legate da operatori logici. Si ha quindi che una funzione logica con n variabili logiche $\alpha_1, \dots, \alpha_n$ è una mappa $F : \mathcal{B}^n \rightarrow \mathcal{B}$.

Esempio 1.1.3. Dati $\alpha, \beta, \gamma \in \mathcal{B}$, l'espressione

$$\delta = (\alpha \vee \beta) \wedge (\neg\gamma)$$

è una funzione logica i.e. si può essere scritta nella forma $\delta = F(\alpha, \beta, \gamma)$ con $F : \mathcal{B}^3 \rightarrow \mathcal{B}$. \diamond

Una funzione logica può essere espressa, oltre che da operazioni logiche, mediante una tabella di verità: ogni riga della tabella corrisponde ad una combinazione delle n variabili in ingresso (2^n in totale) e il risultato viene riportato nella medesima riga, in corrispondenza alla colonna corrispondente alla funzione.

Esempio 1.1.4. Si consideri la seguente funzione logica:

$$\beta = F(\alpha_1, \alpha_2, \alpha_3) = (\alpha_1 \wedge \alpha_2) \vee (\alpha_1 \wedge \alpha_3) \vee (\alpha_2 \wedge \alpha_3), \quad (1.1)$$

in cui $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{B}$. La funzione può essere anche rappresentata dalla tabella di verità riportata in Tabella 1.1. \diamond

Altri operatori logici utilizzati frequentemente sono ad esempio: implicazione logica ($\gamma = \alpha \rightarrow \beta$), equivalenza logica ($\gamma = \alpha \leftrightarrow \beta$), OR esclusivo o XOR ($\gamma = \alpha \bar{\vee} \beta$) le cui tabelle di verità sono riportate in Tabella 1.2.

Definizione 1.1.5. Le variabili logiche $\alpha_1, \alpha_2, \dots, \alpha_n$ sono dette indipendenti se $\forall i, j \in \{1, \dots, n\}$ il valore della variabile logica α_i non dipende dal valore della variabile α_j con $j \neq i$.

| α_1 | α_2 | α_3 | β |
|------------|------------|------------|---------|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

Tabella 1.1: Tabella di verità per la funzione riportata in (1.1).

| α | β | $\alpha \rightarrow \beta$ | $\alpha \leftrightarrow \beta$ | $\alpha \bar{\vee} \beta$ |
|----------|---------|----------------------------|--------------------------------|---------------------------|
| 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |

Tabella 1.2: Tabella di verità per i connettori \rightarrow , \leftrightarrow e $\bar{\vee}$.

1.2 Il prodotto semi-tensore di matrici

Per poter riportare le espressioni logiche in forma vettoriale è necessario introdurre una nuova forma di prodotto, chiamato prodotto semi-tensore di matrici.

Prima di procedere è necessario riportare la definizione di prodotto di Kronecker.

Definizione 1.2.1. Si considerino due matrici $A \in \mathbb{R}^{m \times n}$ e $B \in \mathbb{R}^{p \times q}$; il prodotto di Kronecker tra A e B è definito come:

$$C = A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix},$$

in cui a_{ij} rappresenta l'elemento (i, j) della matrice A , i.e. $[A]_{i,j} = a_{ij}$. Si noti che $C \in \mathbb{R}^{(mp) \times (nq)}$.

Definizione 1.2.2. [9] Siano A e B due matrici tali che $A \in \mathcal{B}^{m \times n}$ e $B \in \mathcal{B}^{p \times q}$ e sia α il minimo comune multiplo di n e p , i.e. $\alpha = \text{mcm}(n, p)$. Il prodotto semi-tensore di A e B è dato da:

$$A \times B = (A \otimes I_{\alpha/n}) (B \otimes I_{\alpha/p}).$$

Osservazione 1.2.3. Si noti che se $n = p$ si ha $A \times B = (A \otimes I_1)(B \otimes I_1) = AB$ e quindi si ottiene il prodotto standard, righe per colonne, tra matrici. Si ha quindi che il prodotto semi-tensore è una generalizzazione del prodotto di matrici che permette di moltiplicare tra loro matrici di dimensione arbitraria. Infatti in questo caso il prodotto semi-tensore modifica le dimensioni delle matrici coinvolte per renderle compatibili e poter poi applicare il prodotto righe per colonne. Dalla Definizione 1.2.2 segue inoltre che:

1. se $A \in \mathcal{B}^{m \times np}$ e $B \in \mathcal{B}^{p \times q}$ allora:

$$A \times B = A(B \otimes I_n);$$

2. se $A \in \mathcal{B}^{m \times n}$ e $B \in \mathcal{B}^{m \times q}$ allora:

$$A \times B = (A \otimes I_p) B.$$

Esempio 1.2.4. [29] Siano $a = [a_1 \ a_2]^T$ e $b = [b_1 \ b_2]^T$. Il prodotto semi-tensore tra a e b risulta:

$$\begin{aligned} a \times b &= (a \otimes I_2) b \\ &= \begin{bmatrix} a_1 & 0 \\ 0 & a_1 \\ a_2 & 0 \\ 0 & a_2 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \\ &= [a_1 b_1 \ a_1 b_2 \ a_2 b_1 \ a_2 b_2]^T. \end{aligned}$$

◇

Il prodotto semi-tensore di matrici soddisfa le proprietà distributiva e associativa. Precisamente, se $A, B, C \in \mathcal{B}^{\times \cdot 1}$, valgono le seguenti espressioni:

$$(A + B) \times C = (A \times C) + (B \times C),$$

$$C \times (A + B) = (C \times A) + (C \times B),$$

$$(A \times B) \times C = A \times (B \times C).$$

¹Il simbolo “.” viene utilizzato quando non è necessario specificare le dimensioni di matrici/vettori ai quali è riferito.

Altre proprietà del prodotto semi-tensore sono riportate nella seguente Proposizione.

Proposizione 1.2.5. *Siano $A, B \in \mathcal{B}^{\times \cdot}$, valgono allora le seguenti proprietà:*

1.

$$A \times B = B^T \times A^T.$$

2. Se $A \in \mathcal{B}^{m \times pn}$, allora

$$A \times I_n = A;$$

se $A \in \mathcal{B}^{pm \times n}$, allora

$$I_m \times A = A.$$

3. Se $A \in \mathcal{B}^{n \times n}$ e $B \in \mathcal{B}^{m \times m}$, allora:

$$\text{tr}(A \times B) = \text{tr}(B \times A).$$

1.3 Logica in forma matriciale

Si è visto come le variabili logiche prendano valori nell'insieme $\mathcal{B} = \{1, 0\}$; i valori 1 e 0 possono però essere equivalentemente rappresentati rispettivamente dai due vettori colonna della matrice I_2 . Infatti si può porre:

$$1 \approx \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad 0 \approx \begin{bmatrix} 0 \\ 1 \end{bmatrix};$$

indicando con δ_n^i la i -esima colonna della matrice identità I_n si ha quindi:

$$1 \approx \delta_2^1 \quad 0 \approx \delta_2^2.$$

Si indichi con il simbolo Δ^k l'insieme $\Delta^k \triangleq \{\delta_k^i : i = 1, \dots, k\}$, ogni variabile Booleana $\alpha \in \mathcal{B}$ può essere allora equivalentemente rappresentata da un vettore $a \in \Delta^2 = \{\delta_2^1, \delta_2^2\}$.

A sua volta ogni funzione Booleana $F : \mathcal{B}^n \rightarrow \mathcal{B}$ può essere equivalentemente espressa dalla mappa $f : (\Delta^2)^n \rightarrow \Delta^2$ in cui le variabili a_i sono vettori canonici.

Definizione 1.3.1. Una matrice $L \in \mathcal{B}^{n \times m}$ è detta matrice logica [9] ($L \in \mathcal{L}^{n \times m}$) se le sue colonne sono vettori canonici, i.e. indicando con $\text{Col}_i(L)$ la i -esima colonna della matrice L si ha $\text{Col}_i(L) \in \Delta^n \forall i \in \{1, \dots, m\}$.

Poiché $L \in \mathcal{L}^{n \times m}$ essa è esprimibile nella forma:

$$L = [\delta_n^{i_1} \delta_n^{i_2} \dots \delta_n^{i_m}],$$

che in seguito sarà anche espressa nella forma:

$$L = \delta_n [i_1 \ i_2 \ \dots \ i_m].$$

Una funzione logica $F(\alpha_1, \alpha_2, \dots, \alpha_n)$, $\alpha_i \in \mathcal{B}$, può essere espressa in forma matriciale per mezzo del prodotto semi-tensore. Per prima cosa, come si è appena visto, le variabili Booleane α_i si possono convertire in variabili vettoriali a_i ; si può quindi ricavare la funzione equivalente $f(a_1, a_2, \dots, a_n)$ con $a_i \in \Delta^2$.

Lemma 1.3.2. Siano a_i , $i = 1, 2, \dots, n$, variabili logiche in forma vettoriale i.e. $a_i \in \Delta^2$. Si definisca il vettore

$$x = \times_{i=1}^n a_i = a_1 \times a_2 \times \dots \times a_n.$$

Gli n vettori a_i sono univocamente determinati da x .

Dimostrazione. Si può dimostrare il Lemma trovando un'espressione per ricavare i vettori a_i . Si noti preliminarmente che, siccome $a_i \in \Delta^2$, allora $x \in \Delta^{2^n}$ e precisamente $\exists j \in \{1, \dots, 2^n\}$ tale che $x = \delta_{2^n}^j$. Si divida ora il vettore in due parti di uguale lunghezza nella forma:

$$x = [x_1^T \ x_2^T]^T,$$

con $x_1, x_2 \in \Delta^{2^{n-1}}$ in cui uno tra x_1 e x_2 necessariamente un vettore nullo. In accordo con la definizione di prodotto semi-tensore, se x_2 è vettore nullo si ricava allora $a_1 = \delta_2^1$ e se invece il vettore nullo è x_1 si ha allora $a_1 = \delta_2^2$. Si può ora procedere dividendo ulteriormente in due parti uguali il vettore non nullo x_k ($k = 1$ o $k = 2$) ottenendo $x_k = [x_{k1}^T \ x_{k2}^T]^T$, e poter quindi ricavare a_2 . Procedendo in questo modo si ricavano univocamente tutti i vettori a_i , $i = \{1, \dots, n\}$. \square

Si ha quindi che il vettore $x \in \Delta^{2^n}$ è in corrispondenza biunivoca con gli n vettori $a_i \in \Delta^2$ e dunque con le n variabili Booleane $\alpha_i \in \mathcal{B}$.

Dal vettore x si possono ricavare gli $\alpha_i \in \mathcal{B}$ (e quindi anche gli $a_i \in \Delta^{2^n}$) corrispondenti e viceversa grazie al seguente criterio:

1. Gli α_i si possono ricavare da $x_j = \delta_{2^n}^j$ ricorsivamente da:

$$\begin{cases} \alpha_i &= \lfloor \frac{q_{i-1}}{2^{n-i}} \rfloor, \\ q_i &= q_{i-1} - 2^{n-i} \alpha_i, \quad i = 1, 2, \dots, n; \end{cases}$$

inizializzando con $q_0 := 2^n - j$.

2. Il valore j si può ottenere dagli α_i mediante l'espressione:

$$j = \sum_{i=1}^n (1 - \alpha_i) 2^{n-i} + 1.$$

Esempio 1.3.3. Si consideri $x = \delta_8^3 = a_1 \times a_2 \times a_3$. Seguendo il criterio appena esposto si ha $j = 3$ e quindi $q_0 := 5$.

Da questo si ricava:

$$\begin{aligned} \alpha_1 &= \lfloor q_0/4 \rfloor = 1, & q_1 &= q_0 - 4\alpha_1 = 1, \\ \alpha_2 &= \lfloor q_1/2 \rfloor = 0, & q_2 &= q_1 - 2\alpha_2 = 1, \\ \alpha_3 &= \lfloor q_2/1 \rfloor = 1. \end{aligned}$$

Portando le variabili Booleane in forma vettoriale si ottiene: $a_1 = \delta_2^1$, $a_2 = \delta_2^2$ e $a_3 = \delta_2^1$. \diamond

Esempio 1.3.4. Si considerino ora $a_1 = \delta_2^2$, $a_2 = \delta_2^1$, $a_3 = \delta_2^1$ e $a_4 = \delta_2^2$; si vuole calcolare il valore $j \in \{1, \dots, 16\}$ tale che $x = \delta_{16}^j = a_1 \times a_2 \times a_3 \times a_4$. Dai vettori a_i si possono ricavare le variabili Booleane equivalenti $\alpha_1 = 0$, $\alpha_2 = 1$, $\alpha_3 = 1$ e $\alpha_4 = 0$. A questo punto applicando il criterio si ha:

$$j = 2^3 + 1 + 1 = 10,$$

si ha quindi $x = \delta_{16}^{10}$. \diamond

Data una funzione in forma vettoriale $f(a_1, a_2, \dots, a_n)$ con $\alpha_i \in \Delta^2$, equivalente alla funzione logica $F(\alpha_1, \alpha_2, \dots, \alpha_n)$ con $\alpha_i \in \mathcal{B}$, si può ottenere una tabella di verità per $f(\cdot)$ a partire da quella della funzione $F(\cdot)$. Infatti è sufficiente sostituire il valore 1 con il vettore δ_2^1 e il valore 0 con il vettore δ_2^2 , ottenendo una nuova tabella di verità. Ponendo inoltre $x = a_1 \times a_2 \times \dots \times a_n$, le 2^n combinazioni delle variabili Booleane possono essere espresse mediante i vettori x ; la tabella di verità di una funzione logica in forma vettoriale è dunque formata da due colonne: nella prima sono riportati i possibili valori di x e nella seconda il valore b ottenuto da $b = f(a_1, a_2, \dots, a_n)$.

Esempio 1.3.5. Si consideri nuovamente la funzione logica $F(\alpha_1, \alpha_2, \alpha_3)$ introdotta in (1.1) e riportata per comodità qui di seguito:

$$\beta = F(\alpha_1, \alpha_2, \alpha_3) = (\alpha_1 \wedge \alpha_2) \vee (\alpha_1 \wedge \alpha_3) \vee (\alpha_2 \wedge \alpha_3).$$

In base alle considerazioni espresse precedentemente esiste una funzione $f(a_1, a_2, a_3)$ tale che $b = f(a_1, a_2, a_3)$ con $a_i, b \in \Delta^2$. Posto $x = a_1 \times a_2 \times a_3$, la funzione $f(\cdot)$ può essere espressa mediante tabella di verità riportata in Tabella 1.3 ottenuta modificando la Tabella 1.1 della funzione logica $F(\cdot)$. \diamond

| x | b |
|--------------|--------------|
| δ_8^1 | δ_2^2 |
| δ_8^2 | δ_2^1 |
| δ_8^3 | δ_2^1 |
| δ_8^4 | δ_2^1 |
| δ_2^5 | δ_2^2 |
| δ_2^6 | δ_2^1 |
| δ_2^7 | δ_2^2 |
| δ_2^8 | δ_2^2 |

Tabella 1.3: Tabella di verità per la funzione (1.1) riportata in forma vettoriale.

Teorema 1.3.6. Sia $f(a_1, a_2, \dots, a_n)$ una funzione con $a_1, a_2, \dots, a_n \in \Delta^2$ e sia $x = \times a_1 \times a_2 \times \dots \times a_n$; esiste allora un'unica matrice $M_f \in \mathcal{L}^{2 \times 2^n}$ tale che:

$$f(a_1, a_2, \dots, a_n) = M_f \times a_1 \times a_2 \times \dots \times a_n = M_f x.$$

La matrice $M_f \in \mathcal{L}^{2 \times 2^n}$ è detta *matrice di struttura* (*structure matrix*) della funzione $f(\cdot)$.

Dimostrazione. La funzione $f(\cdot)$ in forma algebrica può essere espressa, come si è visto precedentemente, tramite la tabella di verità corrispondente. Da questa tabella è possibile ricavare la matrice di struttura M_f . Dato infatti $x = \delta_{2^n}^k$ con $k \in \{1, \dots, 2^n\}$, la tabella fornisce il corrispondente valore $b = f(a_1, a_2, a_3) = \delta_2^h$ con $h \in \{1, 2\}$; quindi per ottenere:

$$\delta_2^h = M_f \delta_{2^n}^k$$

deve valere

$$\text{Col}_k(M_f) = \delta_2^h.$$

Ripetendo l'operazione $\forall k \in \{1, \dots, 2^n\}$, si definisce l'intera matrice M_f e dato che le 2^n colonne sono tutte colonne della matrice I_2 si ottiene $M_f \in \mathcal{L}^{2 \times 2^n}$.

Per dimostrare l'unicità della matrice M_f si assuma per assurdo che esistano due matrici struttura $M_f \neq M'_f$. Indicate con c_i e c'_i le i -esime colonne rispettivamente di M_f e M'_f , dato che le due matrici devono avere almeno una colonna distinta, si supponga $c_i \neq c'_i$. Dato $\bar{x} = \delta_{2^n}^i$ si ottiene:

$$f(a_1, a_2, \dots, a_n) = M_f \bar{x} = c_i \neq c'_i = M'_f \bar{x} = f(a_1, a_2, \dots, a_n)$$

che è assurdo. □

Esempio 1.3.7. Si consideri nuovamente la funzione $f(a_1, a_2, a_3)$ in cui $a_i \in \Delta^2$ la cui tabella di verità è riportata in Tabella 1.3. In questo caso, utilizzando la procedura introdotta nel Teorema 1.3.6 si ha che:

$$f(a_1, a_2, \dots, a_n) = M_f x = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} x,$$

in cui $x = a_1 \times a_2 \times \dots \times a_n$. ◇

1.4 Sistemi di equazioni logiche

Definizione 1.4.1. Un sistema di n equazioni logiche si può esprimere come:

$$\left\{ \begin{array}{l} F_1(\alpha_1, \alpha_2, \dots, \alpha_n) = \beta_1, \\ F_2(\alpha_1, \alpha_2, \dots, \alpha_n) = \beta_2, \\ \vdots \\ F_m(\alpha_1, \alpha_2, \dots, \alpha_n) = \beta_m, \end{array} \right. \quad (1.2)$$

in cui le F_j sono funzioni logiche, gli $\alpha_i \in \mathcal{B}$ sono variabili Booleane e le $\beta_j \in \mathcal{B}$ sono costanti Booleane con $i = 1, \dots, n$ e $j = 1, \dots, m$. Un insieme di costanti Booleane $\gamma_i \in \mathcal{B}$, $i = 1, \dots, n$, tali che ponendo

$$\alpha_i = \gamma_i, \quad \forall i \in \{1, \dots, n\},$$

soddisfa (1.2), è detto soluzione del sistema (1.2).

Come per le funzioni logiche anche per i sistemi logici esiste un'espressione equivalente; precisamente il sistema (1.2) può essere espresso anche mediante la seguente famiglia di espressioni:

$$\begin{cases} f_1(a_1, a_2, \dots, a_n) = b_1, \\ f_2(a_1, a_2, \dots, a_n) = b_2, \\ \qquad \qquad \qquad \qquad \qquad \qquad \vdots \qquad \qquad \qquad \qquad \qquad \qquad \vdots \\ f_m(a_1, a_2, \dots, a_n) = b_m, \end{cases} \quad (1.3)$$

in cui, per $j = 1, \dots, m$ e $i = 1, \dots, n$, $f_j : \Delta^{2^n} \rightarrow \Delta^2$ e $a_i, b_j \in \Delta^2$ con b_j costanti.

In questo modo il sistema può essere riscritto ponendo $x = a_1 \times a_2 \times \dots \times a_n$ e applicando il Teorema 1.3.6 nella seguente forma:

$$\begin{cases} M_1 x = b_1, \\ M_2 x = b_2, \\ \qquad \qquad \qquad \qquad \qquad \qquad \vdots \qquad \qquad \qquad \qquad \qquad \qquad \vdots \\ M_m x = b_m, \end{cases} \quad (1.4)$$

in cui le matrici $M_i \in \mathcal{L}^{2 \times 2^n}$ sono le matrici struttura delle funzioni f_i .

Si consideri ora il sistema (1.4) e si ponga $y = M_1 x \times M_2 x \times \dots \times M_m x$; è possibile costruire la tabella di verità corrispondente al sistema in cui ad ogni valore del vettore $x \in \Delta^{2^n}$ è associato il vettore $y \in \Delta^{2^m}$. Il vettore \bar{x} è soluzione del sistema (1.4) se e solo se ponendo $\bar{y} = M_1 \bar{x} \times M_2 \bar{x} \times \dots \times M_m \bar{x}$ e $z = b_1 \times b_2 \times \dots \times b_m$ si ha $\bar{y} = z$. Le soluzioni del sistema possono quindi essere ricavate dalla tabella di verità associata, in cui sono riportati valori delle y e il vettore x corrispondente.

Esempio 1.4.2. Si consideri il seguente sistema di equazioni vettoriali:

$$\begin{cases} \alpha_1 \wedge \alpha_2 = 1, \\ \alpha_2 \vee \alpha_3 = 1, \\ \alpha_1 \leftrightarrow (\neg \alpha_1) = 1, \end{cases} \quad (1.5)$$

in cui le $\alpha_i \in \mathcal{B}$, $i = 1, 2, 3$, sono le variabili logiche. Si può vedere immediatamente che

$$\begin{cases} \alpha_1 = 1, \\ \alpha_2 = 1, \\ \alpha_3 = 0, \end{cases} \quad (1.6)$$

risolve il sistema (1.5).

Il sistema (1.5) riportato in forma vettoriale diventa:

$$\begin{cases} M_1x = M_1 \times a_1 \times a_2 \times a_3 = \delta_2^1, \\ M_2x = M_1 \times a_1 \times a_2 \times a_3 = \delta_2^1, \\ M_3x = M_1 \times a_1 \times a_2 \times a_3 = \delta_2^1, \end{cases} \quad (1.7)$$

in cui gli $a_i \in \Delta^2$ sono le versioni vettoriali delle variabili logiche α_i . Le matrici struttura risultano:

$$M_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$M_3 = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Dal sistema in (1.7) ponendo $y = M_1x \times M_2x \times M_3x$ si ottiene la tabella di verità associata al sistema e riportata in Tabella 1.4.

Dalla tabella di verità, posto $z = \delta_2^1 \times \delta_2^1 \times \delta_2^1 = \delta_8^1$, si ricava che esiste solo un valore di y tale che $y = z$, ottenuto per $x = \delta_8^2$.

Da $x = \delta_8^2$ si ricavano $a_1 = \delta_2^1$, $a_2 = \delta_2^1$ e $a_3 = \delta_2^2$ e quindi $\alpha_1 = 1$, $\alpha_2 = 1$ e $\alpha_3 = 0$ che corrisponde ovviamente a (1.6). \diamond

Un sistema di equazioni logiche si può riassumere in un'unica equazione matriciale grazie al seguente Teorema.

Teorema 1.4.3. *Si consideri il sistema logico in (1.2) e la sua versione matriciale in (1.4) e siano $x = a_1 \times a_2 \times \dots \times a_n \in \Delta^{2^n}$ e $z = b_1 \times b_2 \times \dots \times b_m \in \Delta^{2^m}$. Il sistema in (1.2) può essere allora convertito in un'unica equazione matriciale*

$$Lx = z, \quad (1.8)$$

in cui la matrice $L \in \mathcal{L}^{2^m \times 2^n}$ è unica.

| x | y |
|--------------|--------------|
| δ_8^1 | δ_8^2 |
| δ_8^2 | δ_8^1 |
| δ_8^3 | δ_8^6 |
| δ_8^4 | δ_8^7 |
| δ_8^5 | δ_8^5 |
| δ_8^6 | δ_8^6 |
| δ_8^7 | δ_8^5 |
| δ_8^8 | δ_8^8 |

Tabella 1.4: Tabella di verità del sistema in (1.7).

Dimostrazione. La matrice L si può ricavare dalla tabella di verità associata al sistema in (1.4). Come per il Teorema 1.3.6 dato infatti $x = \delta_{2^n}^k$ con $k \in \{1, \dots, 2^n\}$ la tabella fornisce il corrispondente valore $z = \delta_2^h$ con $h \in \{1, \dots, 2^m\}$ e quindi per ottenere:

$$\delta_{2^m}^h = L\delta_{2^n}^k$$

si ha che deve essere:

$$\text{Col}_k(L) = \delta_{2^m}^h.$$

Ripetendo l'operazione $\forall k \in \{1, \dots, 2^n\}$ si definisce l'intera matrice L e dato che le 2^n colonne sono tutte colonne della matrice identità I_{2^m} si ottiene $L \in \mathcal{L}^{2^m \times 2^n}$.

La dimostrazione dell'unicità della matrice L , qui omessa, si può ottenere per assurdo in modo analogo alla dimostrazione dell'unicità delle structure matrices M_{f_i} . \square

Dato dunque che $L \in \mathcal{L}^{2^m \times 2^n}$ e $z \in \Delta^{2^m}$ risulta che l'espressione matriciale in (1.8) ha soluzione se e solo se $\exists i \in \{1, \dots, m\}$ tale che

$$z = \text{Col}_i(L).$$

Teorema 1.4.4. *Si definisca l'insieme*

$$\Lambda = \{\lambda : \text{Col}_\lambda(L) = z\}.$$

L'insieme delle soluzioni del sistema (1.8) è

$$\{x = \delta_{2^n}^\lambda, \lambda \in \Lambda\}.$$

Esempio 1.4.5. Il sistema logico riportato in (1.5) e in forma vettoriale in (1.7) si può esprimere nella forma:

$$Lx = \delta_8^1 \quad (1.9)$$

in cui la matrice L , ricavabile dalla Tabella 1.4 risulta:

$$L = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Da L si ricava l'insieme $\Lambda = \{2\}$ e quindi l'unica soluzione del sistema in (1.9) è δ_8^2 in accordo con quanto già trovato nell'Esempio 1.4.2. \diamond

Esempio 1.4.6. [9] Si consideri il seguente problema logico: A dice: "B è un bugiardo", B dice: "C è un bugiardo"; C dice: "Sia A sia B sono bugiardi". Quanti sono i bugiardi e quali sono?

Introducendo le variabili logiche:

- α : A è sincero;
- β : B è sincero;
- γ : C è sincero;

il problema si può esprimere in forma logica con il seguente sistema:

$$\begin{cases} \alpha \leftrightarrow \neg\beta = 1, \\ \beta \leftrightarrow \neg\gamma = 1, \\ \gamma \leftrightarrow \neg\alpha \wedge \neg\beta = 1. \end{cases}$$

Riportando il sistema in forma (1.4) si ottiene:

$$\begin{cases} M_1 x = M_1 \times a \times b \times c = \delta_2^1 \\ M_2 x = M_1 \times a \times b \times c = \delta_2^1, \\ M_3 x = M_1 \times a \times b \times c = \delta_2^1, \end{cases} \quad (1.10)$$

in cui $a, b, c \in \Delta^2$ sono le versioni vettoriali delle variabili α, β e γ e le structure matrices risultano:

$$M_1 = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix},$$

$$M_3 = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Dalle structure matrices allora si ottiene la tabella di verità riportata in Tabella 1.5 in cui $y = a \times b \times c$.

| x | y |
|--------------|--------------|
| δ_8^1 | δ_8^8 |
| δ_8^2 | δ_8^5 |
| δ_8^3 | δ_8^2 |
| δ_8^4 | δ_8^3 |
| δ_8^5 | δ_8^4 |
| δ_8^6 | δ_8^1 |
| δ_8^7 | δ_8^5 |
| δ_8^8 | δ_8^8 |

Tabella 1.5: Tabella di verità del sistema in (1.10).

Il sistema in (1.10) si può riscrivere dunque come:

$$Lx = z = \delta_2^1 \times \delta_2^1 \times \delta_2^1 = \delta_8^1,$$

in cui:

$$L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \in \mathcal{L}^{8 \times 8}.$$

Poiché solo $\text{Col}_6(L) = \delta_8^1$ il sistema ha un'unica soluzione

$$x = \delta_8^6,$$

da cui si ricava $\alpha = 0$, $\beta = 1$ e $\gamma = 0$.

Si ha quindi che B è l'unico sincero. ◇

1.5 Sistemi positivi e matrici non negative

Si consideri il seguente sistema lineare a tempo discreto:

$$x(t+1) = Fx(t), \tag{1.11}$$

e si imponga che per ogni stato iniziale $x(0)$ a componenti non negative, la corrispondente evoluzione di stato abbia vettori $x(t)$ a componenti non negative $\forall t > 0$; sistemi di questo tipo sono detti sistemi lineari positivi discreti.

Condizione necessaria e sufficiente affinché il sistema in (1.12) sia un sistema positivo è che la matrice F non abbia alcun elemento negativo, i.e. $F \in \mathbb{R}_+^{n \times n}$. Poiché, come si vedrà successivamente, una Boolean network può essere espressa come un particolare sistema positivo è utile approfondire le proprietà di queste tipologie di sistemi e delle matrici non negative².

Sia $F \in \mathbb{R}_+^{n \times n}$, nel prosieguo si useranno le seguenti notazioni:

- $F \gg 0$ se $f_{ij} > 0$ per ogni i, j : in questo caso si dice che la matrice F è “strettamente positiva”;

²La seguente analisi è presa da [19].

- $F > 0$ se $f_{ij} \geq 0 \forall i, j$ e almeno un elemento della matrice è positivo: in questo caso la matrice F è detta “positiva”;
- $F \geq 0$ se $f_{ij} \geq 0 \forall i, j$ senza escludere il caso che possa aversi $F = 0$: la matrice in questo caso viene detta “non negativa”.

1.5.1 Matrici di permutazione

Le trasformazioni di similarità applicate ad una matrice $F \in \mathbb{R}_+^{n \times n}$ non mantengono, in generale, la proprietà di non negatività per questo motivo la classe di trasformazioni che si applicano a F per studiarne la struttura si riduce alle trasformazioni di cogredienza indotte da matrici di permutazione.

Si consideri la seguente permutazione:

$$\sigma = \begin{pmatrix} 1 & 2 & \dots & n-1 & n \\ i_1 & i_2 & \dots & i_{n-1} & i_n \end{pmatrix}$$

a cui si associa la matrice di permutazione:

$$\Pi_\sigma = [\delta_n^{i_1} \quad \delta_n^{i_2} \quad \dots \quad \delta_n^{i_n}],$$

che trasforma la n -upla di vettori di base (v_1, v_2, \dots, v_n) nella n -upla permutata $(v_{i_1}, v_{i_2}, \dots, v_{i_n}) = (v_1, v_2, \dots, v_n)\Pi_\sigma$. Si noti che la matrice di permutazione è una matrice logica e ortogonale, i.e. si ha rispettivamente $\Pi_\sigma \in \mathcal{L}^{n \times n}$ e $\Pi_\sigma^{-1} = \Pi_\sigma^T$.

Essendo la matrice Π_σ ortogonale, il vettore colonna $x = [\xi_1 \quad \xi_2 \quad \dots \quad \xi_n]^T$ rappresentato nella base permutata diventa:

$$\Pi_\sigma^T \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_n \end{bmatrix} = \begin{bmatrix} (\delta_n^{i_1})^T \\ (\delta_n^{i_2})^T \\ \vdots \\ (\delta_n^{i_n})^T \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_n \end{bmatrix} = \begin{bmatrix} \xi_{i_1} \\ \xi_{i_2} \\ \vdots \\ \xi_{i_n} \end{bmatrix}.$$

Analogamente una trasformazione lineare rispetto alla base (v_1, v_2, \dots, v_n) dalla matrice $F \in \mathbb{R}^{n \times n}$, nella base permutata è rappresentata dalla matrice:

$$\begin{aligned}
\Pi_\sigma^T F \Pi_\sigma &= \begin{bmatrix} (\delta_n^{i_1})^T \\ (\delta_n^{i_2})^T \\ \vdots \\ (\delta_n^{i_n})^T \end{bmatrix} \begin{bmatrix} f_{1,1} & f_{1,2} & \cdots & f_{1,n} \\ f_{2,1} & f_{2,2} & \cdots & f_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n,1} & f_{n,1} & \cdots & f_{n,n} \end{bmatrix} \begin{bmatrix} \delta_n^{i_1} & \delta_n^{i_2} & \cdots & \delta_n^{i_n} \end{bmatrix} \\
&= \begin{bmatrix} f_{i_1, i_1} & f_{i_1, i_2} & \cdots & f_{i_1, i_n} \\ f_{i_2, i_1} & f_{i_2, i_2} & \cdots & f_{i_2, i_n} \\ \vdots & \vdots & \ddots & \vdots \\ f_{i_n, i_1} & f_{i_n, i_1} & \cdots & f_{i_n, i_n} \end{bmatrix}
\end{aligned}$$

in cui la permutazione è stata applicata alle colonne e alle righe di F .

1.5.2 Classificazione delle matrici non negative

Una matrice $F \in \mathbb{R}_+^{n \times n}$, $n > 1$, può essere :

- primitiva: se esiste un intero $h > 0$ per cui la matrice F^h risulti strettamente positiva; il più piccolo esponente h per cui $F^h \gg 0$ è detto “esponente di primitività” di F
- irriducibile: se $\forall i, j \in \{1, 2, \dots, n\}$ esiste un esponente h (in generale dipendente dalla coppia i, j) per cui

$$[F^h]_{i,j} > 0;$$

- riducibile: se esiste una coppia i, j tale che si abbia:

$$[F^h]_{i,j} = 0,$$

$$\forall h > 0.$$

Le matrici riducibili possono essere trasformate mediante opportune trasformazioni di cogredienza e messe in “forma normale”; si consideri a questo proposito la seguente Proposizione.

Proposizione 1.5.1. [19] Sia $F \in \mathbb{R}_+^{n \times n}$ una matrice non negativa riducibile di dimensione $n > 1$. Esiste allora una matrice di permutazione Π che per cogredienza porta la matrice F nella “forma normale”

$$\bar{F} = \Pi^T F \Pi = \left[\begin{array}{cccc|cccc} \bar{F}_{1,1} & & & & & & & \\ 0 & \bar{F}_{2,2} & & & & & & \\ 0 & \dots & \ddots & & & & 0 & \\ 0 & 0 & \dots & \bar{F}_{h,h} & & & & \\ \hline \star & \star & \dots & \star & \bar{F}_{h,h} & & & \\ \star & \star & \dots & \star & \star & \ddots & & \\ \star & \star & \dots & \star & \star & \star & \bar{F}_{k,k} & \end{array} \right] \quad (1.12)$$

in cui ciascun blocco diagonale $\bar{F}_{i,i}$ è una matrice irriducibile o la matrice nulla di dimensione 1×1 , e in ciascuna riga successiva all' h -esima (se $k > h$) uno almeno dei blocchi fuori diagonale e indicati con \star è una matrice positiva (quindi non nulla).

Capitolo 2

Boolean Networks

2.1 Dinamica di una Boolean Network

Definizione 2.1.1. Una Boolean network (BN) è un sistema dinamico a tempo discreto descritto da variabili Booleane, dette nodi della rete, che interagiscono tra di loro. In qualunque istante, quindi, ogni nodo può avere solo due valori: 1 o 0. Una BN con n nodi può essere descritta dal sistema di equazioni:

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \dots, X_n(t)) \\ X_2(t+1) = F_2(X_1(t), X_2(t), \dots, X_n(t)) \\ \vdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \dots, X_n(t)) \end{cases} \quad (2.1)$$

in cui $X_i \in \mathcal{B}$ e le F_i , $i = 1, 2, \dots, n$, sono funzioni logiche.

2.1.1 Rappresentazione in spazio di stato di una BN

Recentemente Daizhan Cheng et al. [9] hanno sviluppato una rappresentazione in spazio di stato per le BN utilizzando il prodotto semi-tensore di matrici. I loro studi comprendono controllabilità e raggiungibilità [10], teoria della realizzazione [6] e altro ancora [7], [11] e [12]. Tutti i loro studi sono riportati in [9]. Rappresentare il sistema in spazio di stato consente di studiare il comportamento delle BN tramite gli strumenti della Teoria dei Sistemi.

Come prima operazione, utilizzando gli strumenti introdotti nel Capitolo 1 si possono esprimere gli stati in forma vettoriale. Le variabili di stato $X_i \in \mathcal{B} = \{0, 1\}$ vengono convertite rispettivamente in vettori canonici $N_i \in \Delta^2 = \{\delta_2^2, \delta_2^1\}$, mentre

le funzioni Booleane $F_i : \mathcal{B}^n \rightarrow \mathcal{B}$ possono essere equivalentemente rappresentate da funzioni $f_i : (\Delta^2)^n \rightarrow \Delta^2$.

Una BN quindi può essere definita, in modo equivalente, dal sistema di equazioni:

$$\begin{cases} N_1(t+1) = f_1(N_1(t), N_2(t), \dots, N_n(t)) \\ N_2(t+1) = f_2(N_1(t), N_2(t), \dots, N_n(t)) \\ \vdots \\ N_n(t+1) = f_n(N_1(t), N_2(t), \dots, N_n(t)) \end{cases} \quad (2.2)$$

in cui $N_i \in \Delta^2$ e $f_i : \Delta^{2^n} \rightarrow \Delta^2$ con $i = 1, 2, \dots, n$.

Dai vettori $N_i(t) \in \Delta^2$ è possibile quindi ottenere il vettore di stato:

$$x(t) \triangleq N_1(t) \times N_2(t) \times \dots \times N_n(t) = \times_{i=1}^n N_i(t).$$

Il vettore di stato $x(t)$ rappresenta la configurazione dei nodi all'istante t ; più precisamente si ha che le variabili $x_i \triangleq \delta_{2^n}^i \in \Delta^{2^n}$ sono in corrispondenza biunivoca con le n -uple di possibili valori Booleani che le variabili $N_i(t)$ assumono. Gli stati possibili di una BN sono indicati come $x_i \triangleq \delta_{2^n}^i$, con $i = 1, \dots, 2^n$.

Osservazione 2.1.2. Si faccia attenzione a non confondere tra loro le notazioni appena introdotte:

- $X_i(t)$: rappresenta il nodo *iesimo* della Boolean network al tempo t e assume i valori 0 o 1 ($X_i \in \mathcal{B}$).
- $N_i(t)$: rappresenta l'equivalente vettoriale del valore del nodo X_i al tempo t e assume i valori δ_2^1 o δ_2^2 . Il nodo X_i e il vettore N_i sono in corrispondenza biunivoca tra loro infatti $\delta_2^1 \leftrightarrow 0$ e $\delta_2^2 \leftrightarrow 1$;
- $x(t)$: rappresenta le configurazioni degli n nodi della rete al tempo t . Data la corrispondenza biunivoca tra $x(t)$ e gli n valori $N_1(t), \dots, N_n(t)$ da uno si possono ricavare gli altri e viceversa seguendo il procedimento a pag. 11.

Definito lo stato $x(t)$, esistono allora delle “structure matrices, $M_i = M_{f_i}$, $i = 1, \dots, n$, tali che è possibile esprimere $N_i(t+1)$ nella seguente forma:

$$N_i(t+1) = M_i x(t), \quad i = 1, 2, \dots, n. \quad (2.3)$$

Definizione 2.1.3. La (2.3) è detta forma algebrica componente per componente della BN.

Date le n equazioni in (2.3) possono essere riassunte in un'unica equazione:

$$x(t+1) = Lx(t) \quad (2.4)$$

dove L è una matrice quadrata di dimensione $2^n \times 2^n$ ed è detta matrice di transizione.

Definizione 2.1.4. La (2.4) è detta forma algebrica della BN.

Proposizione 2.1.5. La matrice L è una matrice logica i.e. $L \in \mathcal{L}^{2^n \times 2^n}$.

Dimostrazione. Data la definizione di matrice logica si deve dimostrare che $\text{Col}(L) \subset \Delta^{2^n}$. Supponiamo, per assurdo, che esista un j ($1 \leq j \leq 2^n$) tale che $\text{Col}_j(L) \notin \Delta^{2^n}$. Allora ponendo $x(t) = \delta_{2^n}^j$ si ottiene:

$$x(t+1) = Lx(t) = \text{Col}_j(L) \notin \Delta^{2^n}$$

che porta ad una contraddizione. □

La matrice L contiene tutta l'informazione necessaria per descrivere la dinamica del sistema.

Teorema 2.1.6. [9] La dinamica della Boolean network in (2.1) è univocamente determinata dalla forma algebrica in (2.4).

Dimostrazione. Sviluppando a ritroso l'equazione (2.4) si ha che:

$$x(t) = Lx(t-1) = L^2x(t-2) = \dots = L^t x(0), \quad t = 0, 1, 2, \dots$$

L'equazione (2.3) diventa:

$$N_i(t) = M_i x(t-1) = M_i L^{t-1} x(0), \quad i = 1, 2, \dots$$

Quindi si ha che (2.4) determina univocamente la dinamica di (2.1). □

2.2 Punti fissi e cicli limite

Una Boolean network con n nodi ha un numero finito (2^n) di possibili configurazioni dei nodi, definite stati di una BN, per questo motivo per ogni condizione iniziale la dinamica deve terminare in un attrattore, i.e. un punto fisso oppure un ciclo limite. Uno dei problemi principali consiste appunto nel determinare gli attrattori di una BN e gli altri stati, detti transitori. In letteratura molti studi hanno affrontato questo problema ad esempio in [23] sono presentati due metodi (uno iterativo e uno in forma scalare) per determinare la struttura dei cicli e gli stati transitori che portano a questi. In [16], viene ricavata un'equazione scalare, lineare e ridotta da cui ottenere facilmente informazioni sia sui cicli che sulla parte transitoria della rete. Zhao in [41] ha dimostrato che trovare i punti fissi e i cicli limite in una BN è un problema NP-completo. In questa sezione si affronta lo studio dei punti fissi e dei cicli limite di una BN espressa in forma algebrica attraverso i risultati principali degli studi di Cheng e dei suoi colleghi.

Definizione 2.2.1.

1. Uno stato x_i è detto punto fisso del sistema (2.4) se $Lx_i = x_i$.
2. L'insieme di stati $\{x_i, Lx_i, \dots, L^{k-1}x_i\}$ è detto ciclo limite del sistema (2.4) se gli elementi dell'insieme sono tutti distinti e $L^k x_i = x_i$.

Teorema 2.2.2. [9] *Si consideri la Boolean network (2.1). $\delta_{2^n}^i$ è un punto fisso del sistema se e solo se, nella forma algebrica (2.4), l'elemento diagonale l_{ii} della matrice di transizione L è uguale a 1, i.e. $Col_i(L) = \delta_{2^n}^i$. Segue che, indicando con N_1 il numero di punti fissi della BN, si ha che:*

$$N_1 = \text{tr}(L).$$

Dimostrazione. Essendo $Col_i(L) = L\delta_{2^n}^i$, dalla definizione di punto fisso segue che $x_i = \delta_{2^n}^i$ è punto fisso se e solo se:

$$Col_i(L) = L\delta_{2^n}^i = \delta_{2^n}^i$$

che completa la dimostrazione. □

Il secondo risultato fondamentale riguarda i cicli limite di una Boolean network.

Teorema 2.2.3. [9] *Il numero dei cicli limite di lunghezza s si può ottenere ricorsivamente da:*

$$N_s = \frac{\text{tr}(L^s) - \sum_{k \in \mathcal{P}(s)} k N_k}{s} \quad 2 \leq s \leq 2^n, \quad (2.5)$$

dove $\mathcal{P}(s)$ indica l'insieme dei divisori propri¹ di s .

La ricorsione si inizializza con $N_1 = \text{tr}(L)$ in accordo con il Teorema 2.2.2.

Dimostrazione. Si noti per prima cosa che la traccia di L^s è pari al numero di cicli di lunghezza s o un divisore di s ciascuno moltiplicato per il numero di elementi che lo compongono ovvero per la sua lunghezza stessa. Infatti per un elemento $\delta_{2^n}^i$ di un ciclo di lunghezza s si ha che $L^s \delta_{2^n}^i = \delta_{2^n}^i$ però ugualmente anche per un elemento $\delta_{2^n}^j$ di un ciclo di lunghezza $k \in \mathcal{P}(s)$ esiste un $h \in \mathbb{N}$ tale che $L^s \delta_{2^n}^j = L^{hk} \delta_{2^n}^j = \delta_{2^n}^j$. Si ha quindi che il numero di stati appartenenti ad un ciclo di lunghezza s è pari a: $\text{tr}(L^s) - \sum_{k \in \mathcal{P}(s)} k N_k$. Per ottenere allora il numero di cicli di lunghezza s , basta dividere il numero degli stati appartenenti a tali cicli per la lunghezza dei cicli stessi (cioè s), ottenendo la formula in (2.5). \square

Data una BN a n nodi, come si è già detto, ci possono essere 2^n possibili stati e quindi non possono esistere cicli di lunghezza maggiore di 2^n .

L'insieme dei punti fissi unito all'insieme dei cicli limite costituisce l'insieme degli attrattori, indicato con Ω . Partendo da ogni possibile stato $x_i = \delta_{2^n}^i$ $i = 1, \dots, 2^n$, i.e. assumendo $x(0) = x_i$, dopo un numero di passi $T_t(x_i)$ si ha che $x(T_t(x_i)) \in \Omega$. $T_t(x_i)$ viene definito tempo di assorbimento o transitorio dello stato x_i .

Definizione 2.2.4. Il tempo di assorbimento T_t di una Boolean network è definito come:

$$T_t = \max_{x \in \Delta^{2^n}} T_t(x). \quad (2.6)$$

Si ha quindi che, partendo da una qualsiasi configurazione di una BN, dopo T_t passi il valore dei nodi della rete sarà necessariamente periodico o costante.

Si noti che L ha solo $r \triangleq 2^n \times 2^n$ possibili valori differenti; pertanto nella sequenza di $r + 1$ matrici:

$$I_{2^n}, L, L^2, \dots, L^r$$

ci devono necessariamente essere due matrici uguali. Sia $r_0 < r$ il più piccolo i tale che L^i compaia nuovamente nella sequenza, i.e. $\exists k > i : L^i = L^k$. Risulta allora,

¹L'insieme dei divisori propri $\mathcal{P}(s)$ comprende anche 1 e.g. $\mathcal{P}(12) = \{1, 2, 3, 4, 6\}$.

$$r_0 = \operatorname{argmin}_{0 \leq i < r} \{L^i \in \{L^{i+1}, L^{i+2}, \dots, L^r\}\}. \quad (2.7)$$

Proposizione 2.2.5. *Per ogni stato iniziale $x(0)$, la traiettoria di stato della BN a partire da $x(0)$ dopo r_0 passi si trova sicuramente in un ciclo.*

Dimostrazione. Dato uno stato qualsiasi $\delta_{2^n}^i$ e ponendo $x(0) = \delta_{2^n}^i$ si ha che $x(r_0) = L^{r_0}x(0)$. Per la definizione di r_0 , $\exists k$ tale che $L^{r_0} = L^{r_0+k}$ e quindi $x(r_0 + k) = x(r_0)$. \square

Osservazione 2.2.6. Dalla definizione di tempo di assorbimento si ha che $T_t \leq r_0$.

Sia $T > 0$ il più piccolo intero positivo tale che $L^{r_0} = L^{r_0+T}$. Per come sono stati definiti r_0 e T si ha che $r_0 + T \leq r$.

Si ha dunque il seguente Lemma:

Lemma 2.2.7. *Se esiste nella BN un ciclo di lunghezza t , allora t è un divisore di T .*

Dimostrazione. Supponiamo per assurdo che $s > 0$ sia il resto della divisione T/t . Sia x_i uno stato del ciclo allora $L^{r_0}x_i$ è ancora appartenente al medesimo ciclo.

Essendo $1 \leq s < t$, si ha quindi:

$$L^{r_0+T}x_i = L^{r_0}(L^T x_i) = L^{r_0}(L^s x_i) = L^s L^{r_0}x_i \neq L^{r_0}x_i$$

e dunque si ha:

$$L^{r_0} = L^{r_0+T} \neq L^{r_0}$$

da cui l'evidente contraddizione. \square

Teorema 2.2.8. [9] *Il tempo di assorbimento è proprio r_0 e quindi:*

$$T_t = r_0.$$

Dimostrazione. Dalla definizione di tempo di assorbimento segue che per ogni stato x_i , $L^{T_t}x_i$ è su un ciclo che per il Lemma 2.2.7 ha lunghezza divisibile per T . Quindi:

$$L^{T_t}x_i = L^T L^{T_t}x_i = L^{T_t+T}x_i, \quad \forall x_i \in \Delta^{2^n}$$

e conseguentemente $L^{T_t} = L^{T_t+T}$. Dalla definizione di r_0 risulta che $T_t \geq r_0$ e dall'Osservazione 2.2.6 si ha che $T_t \leq r_0$ e quindi $T_t = r_0$. \square

Esempio 2.2.9. Si consideri la seguente Boolean network in forma algebrica:

$$x(t+1) = Lx(t) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} x(t).$$

Dato che $\text{tr}(L) = 1$ e $[L]_{6,6} = 1$ si ha che x_6 è l'unico punto fisso della BN ($N_1 = 1$). Per trovare i cicli di lunghezza 2 si calcolano le potenze della matrice di transizione ottenendo:

$$L^2 = \delta_8[7 \ 1 \ 1 \ 7 \ 5 \ 6 \ 3 \ 8],$$

e quindi si ricava N_2 dalla equazione (2.5):

$$N_2 = \frac{\text{tr}(L^2) - N_1}{2} = \frac{3 - 1}{2} = 1.$$

Dato che $[L^2]_{5,5} = [L^2]_{6,6} = [L^2]_{8,8} = 1$ e siccome x_6 si è già visto essere un punto fisso si ottiene che l'unico ciclo di lunghezza 2 comprende gli stati x_5 e x_8 .

Allo stesso modo si procede per il calcolo dei cicli di lunghezza 3 ottenendo in questo caso:

$$L^3 = \delta_8[1 \ 3 \ 3 \ 1 \ 8 \ 6 \ 7 \ 5],$$

e sempre dall'equazione (2.5) si ricava:

$$N_3 = \frac{\text{tr}(L^3) - N_1}{2} = \frac{4 - 1}{3} = 1.$$

Procedendo analogamente si ottiene che l'unico ciclo di lunghezza 3 è dato dagli stati x_1 , x_3 e x_7 . Gli ultimi due stati rimanenti x_2 e x_4 devono quindi essere stati transitori. Per calcolare il tempo di assorbimento T_t si può notare che:

$$L^8 = L^2 = \delta_8[7 \ 1 \ 1 \ 7 \ 5 \ 6 \ 3 \ 8],$$

e quindi $r_0 = 2$. Per il Teorema 2.2.8 si ha quindi che $T_t = 2$. ◇

2.3 Rappresentazione grafica

Le interazioni tra i diversi nodi di una BN possono essere rappresentate da un grafo orientato con n vertici.

Definizione 2.3.1. Un grafo di influenza è una coppia $\Sigma = \{\tilde{\mathcal{N}}, \tilde{\mathcal{E}}\}$ che consiste in un insieme di vertici, $\tilde{\mathcal{N}} = \{X_i : i = 1, \dots, n\}$, e un insieme di archi, $\tilde{\mathcal{E}} \subset \{X_1, \dots, X_n\} \times \{X_1, \dots, X_n\}$. $(X_i, X_j) \in \tilde{\mathcal{E}}$ equivale a dire che esiste un arco da X_j a X_i (tipicamente indicato con $X_i \rightarrow X_j$) e quindi che la dinamica del vertice X_j dipende dal valore di X_i .

Osservazione 2.3.2. Si noti che la Boolean network e il grafo di influenza non sono in corrispondenza biunivoca. Infatti mentre ad ogni BN corrisponde univocamente un grafo, un grafo può rappresentare più BN.

In una Boolean network il numero di archi che arrivano al vertice X_i si dice grado entrante del vertice X_i , mentre il numero di archi che partono dal vertice X_i è detto grado uscente del vertice X_i .

Definizione 2.3.3. Il grafo di influenza può essere espresso anche da una matrice booleana $n \times n$, detta matrice di incidenza \mathcal{I} , che è definita nel seguente modo:

$$[\mathcal{I}]_{i,j} = \begin{cases} 1, & \text{se } (X_j, X_i) \in \tilde{\mathcal{E}}, \\ 0, & \text{altrimenti.} \end{cases}$$

Il grafo di influenza e la matrice di incidenza sono equivalenti cioè danno le medesime informazioni e quindi dal primo si può ricavare la seconda e viceversa.

Esempio 2.3.4. Si consideri la seguente Boolean network:

$$\begin{cases} X_1(t+1) = X_1(t) \vee X_2(t), \\ X_2(t+1) = \neg X_3(t), \\ X_3(t+1) = X_1(t) \wedge X_2(t). \end{cases} \quad (2.8)$$

Il corrispondente grafo di influenza $\Sigma = (\tilde{\mathcal{N}}, \tilde{\mathcal{E}})$, in cui l'insieme dei vertici è $\tilde{\mathcal{N}} = \{X_1, X_2, X_3\}$ e l'insieme degli archi è composto da $\tilde{\mathcal{E}} = \{(X_1, X_1), (X_2, X_1), (X_3, X_2), (X_1, X_3), (X_2, X_3)\}$, è riportato in Figura 2.1.

La corrispondente matrice di incidenza risulta:

$$\mathcal{I} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

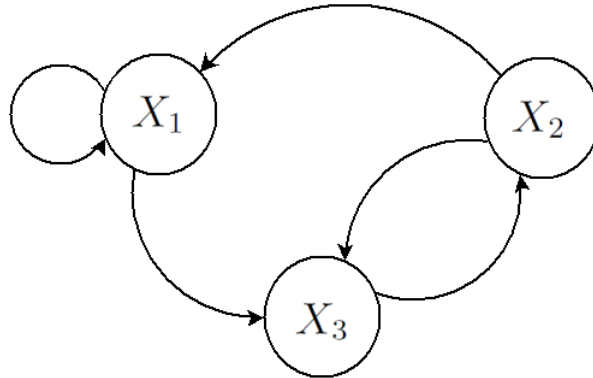


Figura 2.1: Grafo di influenza della BN riportata in (2.8)

Come abbiamo notato precedentemente il grafo di influenza, e quindi pure la matrice di incidenza, non rappresentano univocamente la rete; infatti si consideri la BN data dalle seguenti espressioni

$$\begin{cases} X_1(t+1) = X_1(t) \wedge X_2(t) \\ X_2(t+1) = X_3(t) \\ X_3(t+1) = X_1(t) \wedge X(t) \end{cases} \quad (2.9)$$

La BN in (2.9) è chiaramente diversa rispetto a quella in (2.8), però hanno la medesima rappresentazione tramite grafo di influenza. \diamond

Una migliore rappresentazione grafica di una BN si ottiene tramite il grafo di stato che fornisce un'alternativa di studio delle proprietà della rete rispetto a quello esposto nel paragrafo precedente. Esso consiste in un grafo orientato $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ costituito da 2^n vertici (che rappresentano le diverse configurazioni della BN) i.e. $\mathcal{N} = \{x_i = \delta_{2^n}^i : 1 \leq i \leq 2^n\}$. Si avrà un arco con origine nel vertice x_i e termine nel vertice x_j , i.e. $(x_i, x_j) \in \mathcal{E}$, se e solo se $[L]_{i,j} = 1$. Data la struttura speciale delle colonne della matrice L si ha quindi che ogni vertice ha un unico arco uscente. Si noti che l'informazione circa la BN fornita dal grafo di stato è la stessa apportata dalla matrice L e quindi sono in corrispondenza biunivoca.

Esempio 2.3.5. Si consideri la BN in (2.8). La BN in forma di stato risulta:

$$x(t+1) = Lx(t)$$

in cui la matrice L è pari a:

$$L = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} = \delta_8[3\ 1\ 4\ 2\ 4\ 2\ 8\ 6]$$

da cui si ricava il grafo di stato in Figura 2.2. ◇

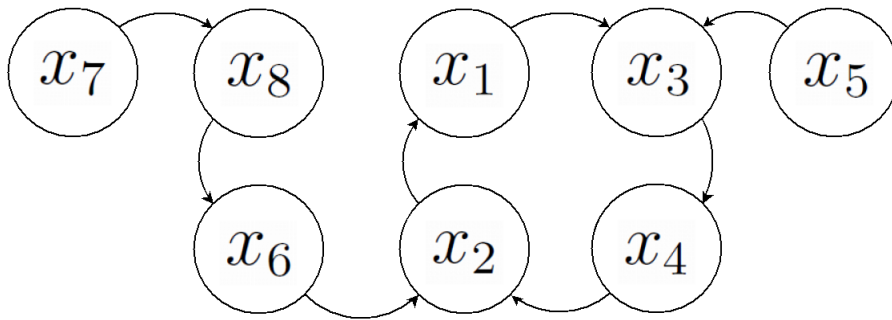


Figura 2.2: Grafo di stato della BN in (2.8)

Osservazione 2.3.6. Si faccia attenzione a non confondere il grafo di influenza precedentemente introdotto con il grafo di stato appena presentato. Essi rappresentano due situazioni differenti: mentre il primo raffigura la reciproca interazione tra i nodi della Boolean network, il secondo rappresenta la dinamica della rete. Dato che ad ogni matrice di transizione corrisponde univocamente un grafo di stato si preferisce usare questo tipo di rappresentazione per le BN.

2.3.1 Accessibilità e comunicazione

Dalla possibilità o impossibilità di transitare da un vertice all'altro, è possibile classificare gli stati della BN in modo da studiare alcune caratteristiche della rete stessa. Queste nozioni sono abitualmente utilizzate nello studio delle catene di Markov [19], ma sono applicabili anche in questo contesto.

Definizione 2.3.7. Uno stato x_k è accessibile dallo stato x_i se esiste un cammino orientato nel grafo degli stati che porta dal vertice x_i al vertice x_k .

La relazione di accessibilità è riflessiva (avendo l'accortezza di includere i cammini di lunghezza nulla) e transitiva (se esiste un cammino da x_i a x_k e un altro da x_k a x_j ne esiste allora sicuramente uno da x_i a x_j) ma non è simmetrica poiché l'accessibilità di x_i da x_k non implica che x_k sia accessibile da x_i .

Definizione 2.3.8. Uno stato x_k comunica con lo stato x_i se x_k è accessibile dallo stato x_i e x_i è accessibile dallo stato x_k .

La relazione di comunicabilità oltre ad essere riflessiva e transitiva come la precedente è anche simmetrica (data la simmetria della definizione) e quindi è una relazione di equivalenza. Questo permette di partizionare l'intero insieme degli stati in classi di equivalenza (chiamate classi di comunicazione) a due a due disgiunte.

Definizione 2.3.9. Una classe \mathcal{K}_s è accessibile da una classe \mathcal{K}_t se esiste uno stato $x_i \in \mathcal{K}_s$ accessibile da uno stato $x_j \in \mathcal{K}_t$.

Date due classi distinte \mathcal{K}_s e \mathcal{K}_t possono esserci tre situazioni mutuamente esclusive:

1. \mathcal{K}_s è accessibile da una classe \mathcal{K}_t ;
2. \mathcal{K}_t è accessibile da una classe \mathcal{K}_s ;
3. \mathcal{K}_s e \mathcal{K}_t sono non interagenti.

Chiaramente stati di classi diverse non possono comunicare e quindi non può esistere un percorso chiuso che esce da una classe e ci fa ritorno.

2.3.2 Tipi di classi di comunicazione

Definizione 2.3.10. Data una Boolean network con n ($n < \infty$) nodi, una classe di comunicazione \mathcal{K} si dice:

- attrattiva (o chiusa) se non esiste uno stato, fuori di \mathcal{K} , accessibile da \mathcal{K} ;
- transitoria se non è attrattiva, i.e. esistono cammini da \mathcal{K} verso stati di altre classi.

Dalla relazione biunivoca tra la BN e il relativo grafo, segue naturalmente il seguente risultato.

Proposizione 2.3.11. *Data la BN riportata in (2.1), si consideri la matrice L ricavata dalla forma algebrica in (2.4) e il relativo grafo degli stati. Si ha che:*

- *una classe chiusa definita sul grafo degli stati corrisponde ad un attrattore nella rispettiva BN e quindi nel grafo deve esistere sempre almeno una classe attrattiva;*
- *una classe chiusa composta da un solo stato è un punto fisso della BN. Graficamente questo è rappresentato da uno stato in cui l'arco uscente si richiude su se stesso (self loop);*
- *le classi transitorie sono formate da un solo stato e quindi possono essere chiamate stati transitori.*

Da queste considerazioni segue che, una volta descritte le configurazioni della BN attraverso la forma grafica e dopo aver classificato gli stati, si hanno automaticamente i punti fissi e i cicli limite. Si noti che ad ogni stato transitorio è associato un solo attrattore. Questo accade perché ogni stato ha un solo arco uscente che ne determina l'unicità della traiettoria; ponendo come configurazione iniziale uno stato transitorio, la traiettoria convergerà all'attrattore associato.

Esempio 2.3.12. Si consideri la seguente Boolean network in spazio di stato:

$$x(t+1) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x(t). \quad (2.10)$$

In Figura 2.3 è riportato il grafo di stato della BN in (2.10) grazie al quale si possono trovare facilmente le differenti classi di comunicazione. Gli stati si possono dividere in 5 classi (evidenziate dai riquadri in Figura) di cui 2 stati transitori (x_3, x_5), 1 punto fisso (x_1) e 2 cicli limite ($\{x_6, x_7\}$ e $\{x_2, x_4, x_8\}$).

◇

Dalla rappresentazione grafica è possibile determinare anche il tempo di assorbimento. Considerando ogni stato transitorio x_i si ha che il numero di archi che compongono il cammino più breve fino all'attrattore corrisponde a $T_t(x_i)$. Dalla definizione (2.6) si trova facilmente T_t ².

²Se x_j appartiene ad una classe attrattrice si ha che $T_t(x_j) = 0$.

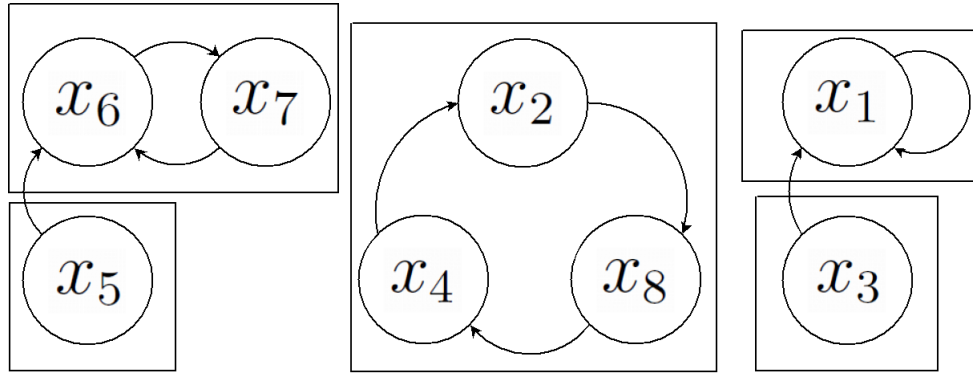


Figura 2.3: Partizioni in classi di comunicazione della BN in (2.10)

2.4 Stabilità

Una volta descritta la dinamica di una Boolean network nello spazio di stato si possono adattare alcune proprietà della teoria dei sistemi come ad esempio la stabilità.

Definizione 2.4.1. Uno stato di una Boolean network viene definito stabile se è un punto fisso.

Esempio 2.4.2. Si consideri la seguente BN in spazio di stato:

$$x(t+1) = Lx(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} x(t).$$

Poiché $[L]_{1,1} = [L]_{4,4} = 1$ si ha che x_1 e x_4 sono punti stabili della BN. \diamond

Esempio 2.4.3. Si consideri ora invece la BN:

$$x(t+1) = Lx(t) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x(t).$$

Dato che $\text{tr}(L) = 0$ si può affermare che la BN non ha punti stabili. \diamond

In una BN, come si è visto dall'Esempio 2.4.3, possono non esistere stati stabili ma solo cicli limite.

Definizione 2.4.4. Una Boolean network è detta globalmente stabile se l'insieme dei suoi attrattori Ω consiste di un unico punto fisso x_i . Infatti, partendo da un qualunque stato $x(0) = x_j \in \Delta^{2^n}$ si ha che $x(k) = x_i$ con $k \geq T_t$.

Per verificare se una BN è globalmente stabile non è necessario trovare tutti gli attrattori, ma si può applicare il seguente Teorema:

Teorema 2.4.5. *La BN è globalmente stabile se e solo $\exists i$ tale che*

$$L^{T_t} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ \vdots & & & \\ 1 & 1 & \dots & 1 \\ 0 & 0 & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & 0 \end{bmatrix} \leftarrow i$$

in cui T_t è il tempo di assorbimento della rete. Lo stato $x_i = \delta_{2^n}^i$ è il punto di stabilità globale.

Osservazione 2.4.6. Si noti che se il sistema è globalmente stabile, allora $\forall k \geq T_t$ si ha che $L^k = \delta_{2^n}[i, \dots, i]$. Dato che $T_t \leq 2^n$, è dunque possibile verificare la globale stabilità di un sistema calcolando direttamente la potenza L^{2^n} .

Esempio 2.4.7. Si consideri la seguente Boolean network in forma algebrica:

$$\begin{aligned} x(t+1) &= L(x(t)) \\ &= \delta_{16}[9 \ 10 \ 9 \ 12 \ 5 \ 12 \ 15 \ 1 \ 5 \ 5 \ 9 \ 16 \ 5 \ 16 \ 1 \ 10]x(t). \end{aligned}$$

Calcolando le potenze successive della matrice L si ottiene:

$$\begin{aligned} L^2 &= \delta_{16}[5 \ 5 \ 5 \ 16 \ 5 \ 16 \ 1 \ 9 \ 5 \ 5 \ 5 \ 10 \ 5 \ 10 \ 9 \ 5] \\ L^3 &= \delta_{16}[5 \ 5 \ 5 \ 10 \ 5 \ 10 \ 9 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5] \\ L^4 &= \delta_{16}[5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5] \\ L^i &= \delta_{16}[5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5] \quad i > 4, \end{aligned}$$

e quindi si ha che $T_t = 4$ e la BN è globalmente stabile in x_5 .

Come si è già notato precedentemente, si può verificare se la BN è globalmente stabile o meno, senza conoscere il tempo di assorbimento, elevando la matrice di transizione al numero di stati, in questo caso 16. Come previsto si ottiene:

$$L^{16} = \delta_{16}[5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5].$$

da cui si ricava comunque che x_5 è uno stato di globale stabilità. \diamond

2.5 Domini di attrazione e forma normale di L

Come evidenziato in precedenza, la struttura particolare di una BN fa sì che dato lo stato iniziale la traiettoria venga definita deterministicamente. Pertanto ad ogni stato transitorio x_i è associato un unico attrattore in cui, ponendo come stato iniziale $x(0) = x_i$, termina la traiettoria.

Definizione 2.5.1. Sia C_i l'insieme di stati che compongono un attrattore (un singolo stato nel caso di punto fisso o gli stati che compongono un ciclo limite). L'insieme $S_i = \{x : \text{se } x(0) = x \text{ allora } x(t) \in C_i, t \geq T_t\}$ è detto dominio di attrazione di C_i .

Dalla definizione di dominio di attrazione segue che $C_i \subseteq S_i$. Da queste considerazioni si ricava la seguente Proposizione.

Proposizione 2.5.2. *Ogni stato appartiene ad uno e un solo dominio di attrazione. Si ha quindi che i domini di attrazione S_i partizionano l'intero insieme delle configurazioni di una BN.*

Data una BN con più classi di comunicabilità si ha che la corrispondente matrice L è una matrice positiva riducibile. In questo caso si può modificare il Teorema (nel cap 1) per porre una matrice riducibile in forma normale ottenendo in questo caso una divisione tra le classi attrattive e quelle transitorie.

Teorema 2.5.3. *Sia $L \in \mathcal{L}^{2^n \times 2^n}$ una matrice di transizione riducibile di una BN (con $n > 1$ nodi). Esiste allora una matrice di permutazione Π che per cogredienza porta la matrice L nella seguente "forma normale"*

$$\bar{L} = \Pi^T L \Pi = \left[\begin{array}{cccc|ccc} \bar{L}_{1,1} & 0 & 0 & 0 & * & * & * \\ 0 & \bar{L}_{2,2} & \vdots & 0 & * & * & * \\ 0 & \dots & \ddots & \vdots & * & * & * \\ 0 & 0 & \dots & \bar{L}_{m,m} & * & * & * \\ \hline & & & 0 & \bar{L}_{m+1,m+1} = 0 & * & * \\ & & & & & \ddots & * \\ & & & & & & \bar{L}_{k,k} = 0 \end{array} \right] \quad (2.11)$$

in cui i blocchi $\bar{L}_{i,i}$ con $i = 1, \dots, m$ sono matrici irriducibili logiche o scalari di valore 1 e rappresentano le classi attrattive mentre i blocchi $\bar{L}_{j,j}$ con $j = m + 1, \dots, k$ (se $k > m$) rappresentano gli stati transitori e quindi sono scalari e nulli. In ogni colonna successiva alla m -esima (se $k > m$) uno e un solo elemento è uguale a 1 mentre gli altri sono tutti nulli.

Data una qualunque BN a n stati ($n > 1$) si può ottenere la matrice di transizione \bar{L} in forma normale ordinando le 2^n configurazioni secondo il seguente criterio:

1. gli stati che appartengono alla medesima classe di comunicazione hanno indici consecutivi;
2. le classi attrattive precedono tutti gli stati transitori;
3. gli stati transitori vengono numerati in ordine crescente rispetto al proprio tempo di assorbimento.

Una volta ordinate le classi e conseguentemente gli stati secondo le regole appena descritte, la matrice di transizione della stessa BN assume la forma normale riportata in (2.11). Si ottiene quindi un sistema in spazio di stato equivalente a quello precedente all'ordinamento, ma separando le due tipologie di classi.

Osservazione 2.5.4. Si faccia attenzione che una volta operato l'ordinamento delle classi bisogna associare ogni stato alla corrispondente configurazione. Mentre prima ad ogni $x_i \triangleq \delta_{2^n}^i$ era associata la configurazione di nodi calcolabile tramite il criterio esposto nel primo capitolo, in questo caso il rapporto tra lo stato e l'effettivo valore dei nodi dipende dalla matrice di permutazione Π utilizzata. Si consideri, a questo proposito, la BN in spazio di stato (2.4) e sia invece:

$$\bar{x}(t+1) = \bar{L}\bar{x}(t) \quad (2.12)$$

il sistema con gli stati ordinati.

Poiché esiste una matrice di permutazione Π tale che $\bar{L} = \Pi^T L \Pi$, si ha che

$$\bar{x}(t+1) = \bar{L}\bar{x}(t) = \Pi^T L \Pi \bar{x}(t)$$

e dunque, per l'ortogonalità di Π , si ottiene

$$\Pi \bar{x}(t+1) = L \Pi \bar{x}(t)$$

da cui

$$x(t) = \Pi \bar{x}(t)$$

$\forall t \in \mathbb{N}$. Fissato dunque uno stato \bar{x} del sistema in (2.12), il corrispondente stato della BN ante ordinamento in (2.4) è $x = \Pi \bar{x}$. Dato quindi $\bar{x}_i = \delta_{2^n}^i$ per il sistema si ha che esiste un indice $j \in 1, \dots, 2^n$ tale che $\Pi \bar{x}_i = \delta_{2^n}^j$ e si può trovare la configurazione di nodi associata allo stato \bar{x}_i grazie al criterio riportato a pag. 11 applicato a $\delta_{2^n}^j$. Procedendo allo stesso modo per tutti gli stati, si può dunque costruire una tabella delle corrispondenze assegnando ad ogni stato \bar{x}_i il valore corrispondente dei nodi. Si consideri il seguente Esempio.

Esempio 2.5.5. Si consideri una BN in forma (2.4) con la seguente matrice di transizione:

$$L = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Si consideri ora la seguente BN:

$$\bar{x}(t+1) = \bar{L}\bar{x}(t)$$

in cui la matrice di transizione è ottenuta mediante un ordinamento degli stati della BN precedente. Si supponga:

$$\bar{L} = \Pi^T L \Pi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

ottenuta attraverso la matrice di permutazione:

$$\Pi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Si supponga di voler trovare la configurazione di nodi corrispondente allo stato $\bar{x}_7 = \delta_{2^n}^7$. Lo stato \bar{x}_7 corrisponde allo stato x_3 della BN originale, infatti si ha:

$$\Pi^T \bar{x}_7 = \delta_8^3.$$

L'equazione $N_1 \times N_2 \times N_3 = \delta_{2^n}^3$ è verificata da $N_1 = \delta_2^1$, $N_2 = \delta_2^2$ e $N_3 = \delta_2^1$ quindi i valori dei nodi corrispondenti allo stato \bar{x}_7 sono $X_1 = 1$, $X_2 = 2$ e $X_3 = 1$.

La tabella delle corrispondenze in questo caso è riportata in Tabella 2.1.

◇

Data una BN con n nodi, supponiamo che ci siano $0 < h \leq 2^n$ stati attrattivi suddivisi in m attrattori C_i , $i = 1, \dots, m$; per calcolare i domini di attrazione di ogni attrattore si può procedere come segue:

1. Si associa ogni attrattore al rispettivo dominio, ovvero $\forall x \in C_i$ si ha che $x \in S_i$. Se $h = 2^n$ non occorre procedere oltre.

| \bar{x} | X_1 | X_2 | X_3 |
|-------------|-------|-------|-------|
| \bar{x}_1 | 1 | 1 | 1 |
| \bar{x}_2 | 1 | 1 | 0 |
| \bar{x}_3 | 0 | 0 | 0 |
| \bar{x}_4 | 1 | 0 | 0 |
| \bar{x}_5 | 0 | 1 | 0 |
| \bar{x}_6 | 0 | 0 | 1 |
| \bar{x}_7 | 1 | 0 | 1 |
| \bar{x}_8 | 0 | 1 | 1 |

Tabella 2.1: Tabella delle corrispondenze per la matrice di transizione \bar{L}

- Si ordinano gli stati per ottenere la matrice di transizione in forma normale.
- Si eleva la matrice \bar{L} al tempo di assorbimento T_t . Si otterrà una matrice del tipo:

$$\bar{L}^{T_t} = \left[\begin{array}{cccc|cc} \bar{L}_{1,1}^{T_t} & 0 & 0 & 0 & T_{1,h+1} & T_{1,2^n} \\ 0 & \bar{L}_{2,2}^{T_t} & & 0 & T_{2,h+1} & \vdots \\ 0 & \dots & \ddots & \vdots & \vdots & \\ 0 & 0 & \dots & \bar{L}_{m,m}^{T_t} & T_{m,h+1} & T_{m,2^n} \\ \hline & & & 0 & & \end{array} \right]$$

in cui i blocchi $\bar{L}_{i,i}^{T_t}$ sono matrici logiche e i blocchi $T_{i,j}$ sono vettori colonna. Per la logicità della matrice \bar{L}^{T_t} in ciascuna colonna successiva all' h -esima, esiste un solo elemento uguale a 1 mentre tutti gli altri sono nulli; fissato quindi $j \in \{h+1, \dots, 2^n\}$, $\exists \bar{i} \leq m$ tale che $T_{\bar{i},j}$ è un vettore non nullo. Trovato \bar{i} si ha quindi che la traiettoria ponendo come stato iniziale $x(0) = x_j$ confluirà nell'attrattore $C_{\bar{i}}$.

- Ripetendo il ragionamento per ogni stato transitorio $x_j, h < j \leq 2^n$, si possono costruire tutti i domini di attrazione $S_i, i = 1, \dots, m$.

Esempio 2.5.6. Si consideri la seguente Boolean network in forma normale:

$$\begin{aligned}
x(t+1) &= Lx(t) \\
&= \delta_{16} [1 \ 3 \ 2 \ 5 \ 6 \ 4 \ 1 \ 3 \ 6 \ 5 \ 7 \ 10 \ 7 \ 8 \ 14 \ 14] x(t),
\end{aligned} \tag{2.13}$$

con i 3 attrattori: $C_1 = \{x_1\}$, $C_2 = \{x_2, x_3\}$ e $C_3 = \{x_4, x_5, x_6\}$.

Per trovare i domini di attrazione S_1 , S_2 e S_3 si può procedere con il criterio spiegato precedentemente. Dato che per la BN in (2.13) si ha che $T_t = 3$ e la matrice di transizione è già in forma normale si può procedere a calcolare L^{T_t} ottenendo:

$$L^3 = \left[\begin{array}{ccc|c} L_1 & & & T \\ & L_2 & & \\ \hline & & L_3 & \\ & & & 0 \end{array} \right]$$

in cui:

$$T = \left[\begin{array}{cccccccccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array} \right] \begin{array}{l} \rightarrow C_1 \\ \rightarrow C_2 \\ \rightarrow C_2 \\ \rightarrow C_3 \\ \rightarrow C_3 \\ \rightarrow C_3 \end{array}$$

e quindi si ha che $x_7, x_{11}, x_{13} \in S_1$, $x_8, x_{14}, x_{15}, x_{16} \in S_2$ e $x_9, x_{10}, x_{12} \in S_3$. Si possono quindi definire i tre domini di attrazione:

$$\begin{aligned}
S_1 &= \{x_1, x_7, x_{11}, x_{13}\}; \\
S_2 &= \{x_2, x_3, x_8, x_{14}, x_{15}, x_{16}\}; \\
S_3 &= \{x_4, x_5, x_6, x_9, x_{10}, x_{12}\}.
\end{aligned}$$

◇

2.6 Forma normale per classi della matrice L

Una volta terminata l'analisi di una Boolean network, i.e. determinati gli stati transitori, gli attrattori ed i loro relativi domini di attrazione ed il tempo di assorbimento, può essere utile ordinare gli stati per avere la matrice di transizione L in una forma più ordinata e che permetta di rendere immediatamente leggibili i risultati ottenuti.

Supponiamo inizialmente che la matrice L sia irriducibile cioè tutti gli stati facciano parte della stessa classe e quindi la traiettoria visita periodicamente ogni stato. Poiché si ha che $L \in \mathcal{L}^{2^n \times 2^n}$, e quindi l'indice di imprimitività η è sempre maggiore di 1 ($\eta = 2^n > 1$), la matrice può essere posta in forma ciclica di Frobenius, introdotta nel primo capitolo, i.e. esiste una matrice di permutazione Π tale che:

$$\bar{L} = \Pi^T L \Pi = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & & 0 \\ & & & \ddots & \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}. \quad (2.14)$$

Questa forma, che può sempre essere ottenuta tramite un ordinamento degli stati, mette in evidenza la natura ciclica di un attrattore e la periodica visita di ogni stato. Si ha infatti che $\exists t$ tale che:

$$\bar{L}^{t\eta} = I_{2^n}.$$

Si possono portare in forma (2.14) anche le singole classi attrattive di una BN. Si consideri a questo proposito una BN priva di stati transitori con m attrattori; la matrice di transizione L , per mezzo di una rinumerazione degli stati (un'opportuna trasformazione di cogredienza), può essere portata in struttura diagonale a blocchi:

$$L = \begin{bmatrix} L_1 & & & \\ & L_2 & & \\ & & \ddots & \\ & & & L_m \end{bmatrix},$$

in cui i blocchi L_i sono irriducibili o scalari di valore 1 e in forma (2.14). In questo caso si è in presenza di m traiettorie disgiunte; se lo stato iniziale appartiene ad una data classe gli altri attrattori non saranno mai visitati.

Dalle considerazioni espresse in questo e nei precedenti paragrafi si possono ordinare gli stati in modo da mettere la matrice di transizione L in forma normale per classe.

Teorema 2.6.1. *Si consideri una BN a n stati con $m > 1$ attrattori e $k = 2^n > m$ configurazioni. Esiste quindi una matrice di permutazione Π tale che:*

$$\hat{L} = \Pi^T L \Pi = \begin{bmatrix} B_1 & & & \\ & B_2 & & \\ & & \ddots & \\ & & & B_m \end{bmatrix} \quad (2.15)$$

in cui le m matrici B_i sono nella forma:

$$B_i = \left[\begin{array}{c|ccc} M_i & & T_i & \\ \hline & 0 & * & * \\ 0 & & \ddots & * \\ & 0 & & 0 \end{array} \right]$$

con M_i matrici irriducibili in forma ciclica di Frobenius, e i blocchi T_i e gli elementi $*$ tali da garantire la logicità delle matrici B_i i.e. $B_i \in \mathcal{L}^{k \times k}$. La matrice \hat{L} viene detta in forma normale per classi.

La matrice in forma (2.15) mette in risalto gli attrattori con i propri domini di attrazione catalogando tutti gli stati della Boolean network e conservando la struttura logica della matrice di transizione. Avendo a disposizione la matrice in forma normale per classi, l'analisi risulta immediata; l'unica informazione non direttamente ricavabile rimane il tempo di assorbimento il cui calcolo è comunque agevolato dall'ordinamento degli stati transitori in ciascuno degli m domini di attrazione. Infatti si può ricavare un upper bound poiché il tempo di assorbimento sarà minore o uguale al massimo delle dimensioni dei blocchi di transizione in ciascuna classe di attrazione.

Esempio 2.6.2. Si consideri la seguente BN in forma di stato a 16 stati con la matrice di transizione L in forma normale per classi:

$$x(t+1) = \begin{bmatrix} B_1 & & \\ & B_2 & \\ & & B_3 \end{bmatrix} x(t)$$

in cui:

$$B_1 = \left[\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right], \quad B_2 = \left[\begin{array}{c|cccc} 1 & 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{array} \right],$$

$$B_3 = \left[\begin{array}{cccc|ccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right].$$

Dalla forma normale per classi è facile ricavare gli attrattori (C_1 , C_2 e C_3) ed i rispettivi domini di attrazione (S_1 , S_2 e S_3). Si ha infatti: $C_1 = \{x_1, x_2\}$, $C_2 = \{x_5\}$ e $C_3 = \{x_{10}, x_{11}, x_{12}, x_{13}\}$. I domini di attrazione invece risultano: $S_1 = \{x_1, x_2, x_3, x_4\}$, $S_2 = \{x_5, x_6, x_7, x_8, x_9\}$ e $S_3 = \{x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}\}$. Inoltre si può ricavare un upper bound per il tempo di assorbimento dato che il maggiore dei blocchi di transizione ha dimensione 4 e quindi si ha che $T_t \leq 4$. Dai tre blocchi B_1 , B_2 e B_3 che compongono L in forma normale di stato si può ottenere il grafo di stato della BN riportato nelle Figure 2.4-2.6.

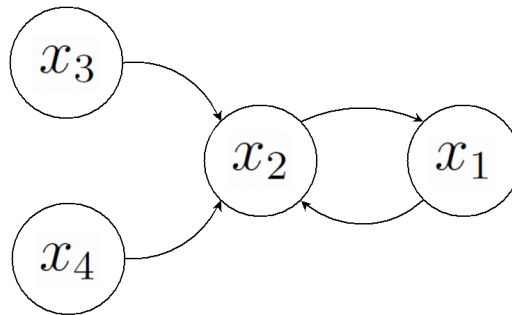
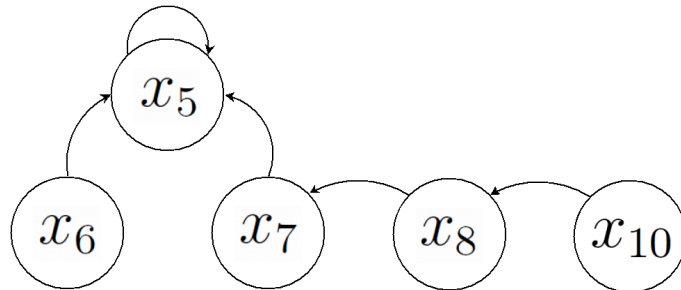
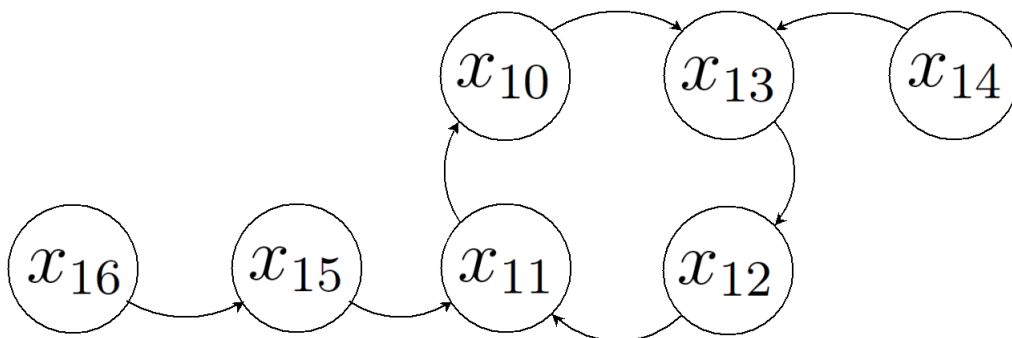


Figura 2.4: Grafo di stato del blocco B_1

Dal grafo si può notare che il numero massimo di archi da uno stato transitorio ad una classe attrattiva è 3 (da x_9 a x_5) e quindi si ha: $T_t = 3$. \diamond

Figura 2.5: Grafo di stato del blocco B_2 Figura 2.6: Grafo di stato del blocco B_3

Capitolo 3

Boolean Control Networks

Introducendo un controllo esterno (sempre Booleano) alla dinamica di una rete Booleana si ottiene una Boolean control network. La presenza degli ingressi permette, ad un utente che ne detiene il controllo, di poter influire sulla dinamica della rete ma introduce alcune criticità quali raggiungibilità, stabilizzabilità e determinazione della sequenza di controllo che verranno affrontati in questo capitolo.

3.1 Dinamica e forma algebrica di una Boolean control network

Definizione 3.1.1. Una Boolean control network è un sistema dinamico a tempo discreto, con n variabili di stato X_i e m ingressi U_i , descritto dal seguente sistema di equazioni:

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \dots, X_n(t), U_1(t), U_2(t), \dots, U_m(t)) \\ X_2(t+1) = F_2(X_1(t), X_2(t), \dots, X_n(t), U_1(t), U_2(t), \dots, U_m(t)) \\ \vdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \dots, X_n(t), U_1(t), U_2(t), \dots, U_m(t)) \end{cases} \quad (3.1)$$

in cui $X_i, U_i \in \mathcal{B}$ e le F_i sono funzioni logiche i.e. $F_i : \mathcal{B}^{n+m} \rightarrow \mathcal{B}$ con $i = 1, 2, \dots, n$.

Ogni variabile Booleana $X_i, U_i \in \mathcal{B}$, come si è già visto per una Boolean network senza ingressi, può essere equivalentemente rappresentata da un vettore $N_i, v_i \in \Delta^2$ con il quale è in relazione biunivoca. Le funzioni $F_i : \mathcal{B}^{n+m} \rightarrow \mathcal{B}$ riportate in (3.1) possono essere sostituite dalle equivalenti $f_i : (\Delta^2)^{n+m} \rightarrow \Delta^2$. Una BCN quindi può essere definita, allo stesso modo, dal sistema di equazioni:

$$\begin{cases} N_1(t+1) = f_1(N_1(t), N_2(t), \dots, N_n(t), v_1(t), v_2(t), \dots, v_m(t)) \\ N_2(t+1) = f_2(N_1(t), N_2(t), \dots, N_n(t), v_1(t), v_2(t), \dots, v_m(t)) \\ \vdots \\ N_n(t+1) = f_n(N_1(t), N_2(t), \dots, N_n(t), v_1(t), v_2(t), \dots, v_m(t)) \end{cases} \quad (3.2)$$

in cui $N_i, v_i \in \Delta^2$ e le $f_i : \Delta^{2^{n+m}} \rightarrow \Delta^2$ con $i = 1, 2, \dots, n$.

Dato che la dinamica di una BCN è anch'essa determinata da funzioni logiche può essere posta, utilizzando le proprietà del prodotto semi-tensore come nel caso delle BN, in spazio di stato. Precisamente si ha il seguente Teorema.

Teorema 3.1.2. *Si consideri la BCN in (3.1) con variabili di stato N_1, \dots, N_n e ingressi v_1, \dots, v_m con $N_i, v_i \in \Delta^2$. Ponendo $x(t) = N_1(t) \times \dots \times N_n(t)$ e $u(t) = v_1(t) \times \dots \times v_m(t)$, esiste un'unica matrice $L \in \mathcal{L}^{2^n \times 2^{n+m}}$ tale che:*

$$x(t+1) = L \times u(t) \times x(t). \quad (3.3)$$

La matrice L anche in questo contesto viene detta matrice di transizione e la forma riportata in (3.3) è detta forma algebrica di una Boolean control network.

Esempio 3.1.3. [29] Si consideri la seguente Boolean control network con due nodi e due ingressi:

$$\begin{cases} X_1(t+1) = X_1(t) \vee [X_2(t) \wedge U_1(t)], \\ X_2(t+1) = X_2(t) \wedge U_2(t). \end{cases} \quad (3.4)$$

La matrice di transizione associata alla forma vettoriale della BCN in (3.3) in questo caso è:

$$L = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix},$$

con $L \in \mathcal{L}^{4 \times 16}$ dato che $n = 2$ e $m = 2$.

Si consideri ad esempio il caso $X_1(t) = 0$, $X_2(t) = 1$, $U_1(t) = 1$ e $U_2(t) = 0$. In questo caso da (3.4) si ottiene:

$$X_1(t+1) = 1 \quad X_2(t+1) = 0. \quad (3.5)$$

I valori dei nodi riportati in forma vettoriale diventano: $N_1(t) = \delta_2^2$, $N_2(t) = \delta_2^1$, $v_1(t) = \delta_2^1$ e $v_2(t) = \delta_2^2$ da cui si ottengono il vettore di stato:

$$x(t) = N_1 \times N_2(t) = \delta_4^3,$$

e il vettore degli ingressi:

$$u(t) = v_1(t) \times v_2(t) = \delta_4^2.$$

Si può calcolare dunque:

$$\begin{aligned} x(t+1) &= L \times u(t) \times x(t) \\ &= L \times \delta_4^2 \times \delta_4^3 \\ &= L \times [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T \\ &= [0 \ 1 \ 0 \ 0]^T. \end{aligned}$$

Dati $N_1(t+1) = [v \ \bar{v}]^T$ e $N_2(t+1) = [w \ \bar{w}]^T$ si ha $x(t+1) = N_1(t+1) \times N_2(t+1) = [vw \ v\bar{w} \ \bar{v}w \ \bar{v}\bar{w}]^T$. Poiché in questo caso si ha $v = \bar{w} = 1$ allora $N_1(t+1) = \delta_2^1$ e $N_2(t+1) = \delta_2^2$ che corrisponde, una volta riportato in variabili Booleane, con il risultato in (3.5). \diamond

3.1.1 La BCN come switched system

Si consideri la BCN con m ingressi riportata in (3.3); essa può essere considerata come uno switched system Booleano composto da 2^m sottosistemi (ognuno dei quali è una BN) in cui il valore degli ingressi determina, istante per istante, quale di questi sottosistemi è attivo. Noto il valore dell'ingresso all'istante \bar{t} , $\bar{u} = u(\bar{t})$, si ha che la dinamica del passo successivo si può scrivere come:

$$x(\bar{t}+1) = L \times u(\bar{t}) \times x(\bar{t}) = L(\bar{u})x(\bar{t})$$

in cui $L(\bar{u}) \in \mathcal{L}^{2^n \times 2^n}$. Si ha quindi che ad ogni configurazione degli ingressi $u_i \triangleq \delta_{2^m}^i$, $i = 1, \dots, m$, è associata una BN corrispondente, la cui matrice di transizione

è $L_i \triangleq L(u_i) = L \times u_i$. Queste matrici $L_i \in \mathcal{L}^{2^n \times 2^n}$ rappresentano i sottosistemi della Boolean control network; più precisamente la matrice di transizione della BCN si può scrivere come:

$$L = [L_1 \quad L_2 \quad \dots \quad L_i \quad \dots \quad L_{2^m}] \quad (3.6)$$

in cui viene messa in evidenza la struttura dello switched system; la dinamica degli ingressi $u(t)$ agisce da funzione di switching.

Esempio 3.1.4. Si consideri la seguente BCN in forma di stato:

$$x(t+1) = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_L u(t)x(t), \quad (3.7)$$

in cui la matrice di transizione L appartiene a $\mathcal{L}^{4 \times 8}$, dato che $n = 2$ e $m = 1$. Poiché l'ingresso è unico, la matrice di transizione è composta da due BN attive alternativamente a seconda del valore di $u(t)$. La matrice L del sistema in (3.7) si può infatti esprimere come:

$$L = [L_1 \quad L_2],$$

da cui si possono ottenere le due BN:

$$x(t+1) = L_i x(t) \quad i = 1, 2.$$

La dinamica della BCN dipende chiaramente dal valore dell'ingresso infatti si ha:

$$L_1 = L \times u_1(t) = L \times [1 \quad 0]^T = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

$$L_2 = L \times u_2(t) = L \times [0 \quad 1]^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

◇

3.1.2 La matrice L_{tot}

Per studiare molte proprietà di una BCN non è necessario utilizzare la matrice di transizione L ; essa infatti associa ad ogni configurazione degli ingressi (2^m in totale) lo stato attivo all'istante successivo. Questa informazione può essere trascurata ogniqualvolta sia necessario conoscere quali stati possono seguire alla configurazione attiva dato un certo numero di ingressi, ma non a quale specifico ingresso sia legata una particolare dinamica.

Si consideri a questo scopo la matrice calcolata come segue:

$$L_{tot} \triangleq \sum_{i=1}^{2^m} L_i \quad ^1$$

dove le matrici $L_i \in \mathcal{L}^{2^n \times 2^n}$ si ricavano dall'espressione di L in (3.6); si ha dunque che $L_{tot} \in \mathbb{Z}^{2^n \times 2^n}$.

Per la logicità delle matrici L_i si ha che la somma di tutti gli elementi di ogni colonna della matrice L_{tot} è pari a 2^m i.e. $\sum_{i=1}^{2^n} [L_{tot}]_{i,j}$. Indicando con $\mathbf{1}_j$ il vettore colonna di lunghezza j in cui tutti gli elementi sono 1 e notando che $\sum_{i=1}^{2^m} L_i = (L \otimes I_1)(\mathbf{1}_{2^m} \otimes I_{2^n})$ si può dare una definizione equivalente di L_{tot} sfruttando il prodotto semi-tensore. Si ha infatti che:

$$L_{tot} = L \times \mathbf{1}_{2^m}.$$

Esempio 3.1.5. Si consideri ancora la BCN in (3.7).

In questo caso si ha $L_{tot} \in \mathbb{Z}^{4 \times 4}$, infatti:

$$L_{tot} = L_1 + L_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

◇

La matrice L_{tot} è di notevole importanza nello studio della controllabilità e della stabilizzabilità di una BCN e inoltre permette di rappresentare graficamente la rete.

¹Si noti che la somma delle L_i è quella usuale tra matrici e non una somma booleana. Questa scelta, come si vedrà più avanti, permette di mantenere l'informazione sul numero dei differenti percorsi da uno stato ad un altro.

3.2 Rappresentazione grafica

Come per le BN anche per le BCN si possono rappresentare gli stati graficamente tramite un grafo di stato $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. Anche in questo caso i vertici sono dati dalle 2^n possibili configurazioni degli n nodi della BCN, $\mathcal{N} = \{x_i = \delta_{2^n}^i : 1 \leq i \leq 2^n\}$, mentre gli archi orientati si ricavano dalla matrice L_{tot} definita nella precedente sezione. Precisamente si avrà $(x_i, x_j) \in \mathcal{E}$ (arco con origine in x_i e termine in x_j) se e solo se $[L_{tot}]_{i,j} > 0$. Dato che la somma degli elementi di ogni colonna della matrice L_{tot} è pari a 2^m , da ogni vertice partono al più 2^m archi: possono esistere infatti due o più ingressi di controllo che portano dallo stesso stato x_i allo stesso stato x_j . Il grafo degli stati, per come è stato precedentemente definito, dà la medesima informazione della matrice L_{tot} e quindi ad ogni arco corrisponde l'esistenza di almeno un ingresso, mentre ingressi di controllo distinto non corrispondono necessariamente ad archi distinti. Anche l'insieme degli stati di una BCN può essere partizionato in classi di comunicazione, le stesse proposte nel precedente Capitolo: chiuse² o transitorie. Le classi transitorie di una BCN, al contrario di quanto accade in una BN, possono essere composte da più stati e possono esistere sequenze di controllo tali per cui la dinamica visita sempre gli stati di una stessa classe transitoria e quindi per una BCN non è detto sia possibile definire il tempo di assorbimento. Per come sono state definite le classi di comunicazione se la traiettoria passa da uno stato di una classe \mathcal{K}_1 ad uno stato di una classe \mathcal{K}_2 , gli stati della classe \mathcal{K}_1 non potranno più essere visitati dalla dinamica della BCN. Anche per le BCN vale una proprietà già vista per le BN:

Proposizione 3.2.1. *Ogni BCN con un numero di stati $2^n < \infty$ possiede almeno una classe chiusa. Per ogni stato x_i appartenente ad una classe transitoria esiste almeno un percorso orientato che porta ad una classe chiusa \mathcal{K} .*

Si noti che, differentemente da quanto accade nelle BN, ad una classe transitoria di una BCN non è generalmente associata un'unica classe chiusa, ma le classi visitate successivamente variano a seconda dell'arco scelto. Una volta che la dinamica di una BCN visita uno stato di una classe chiusa si potranno visitare solo gli stati appartenenti a quella determinata classe.

Esempio 3.2.2. Si consideri la seguente BCN a due ingressi in forma di stato (3.3) con matrice di transizione

²Per una BCN è preferibile riferirsi a classi chiuse piuttosto che a classi attrattive poiché l'attrattività perde significato a causa del controllo arbitrario.

$$L = \left[\begin{array}{cccccccc|cccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array} \right],$$

da cui si ricava la matrice

$$L_{tot} = \left[\begin{array}{cccccccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 2 \end{array} \right]. \quad (3.8)$$

A sua volta dalla matrice L_{tot} si può ottenere il grafo degli stati che è riportato in Figura 3.1 in cui sono evidenziate anche le differenti classi.

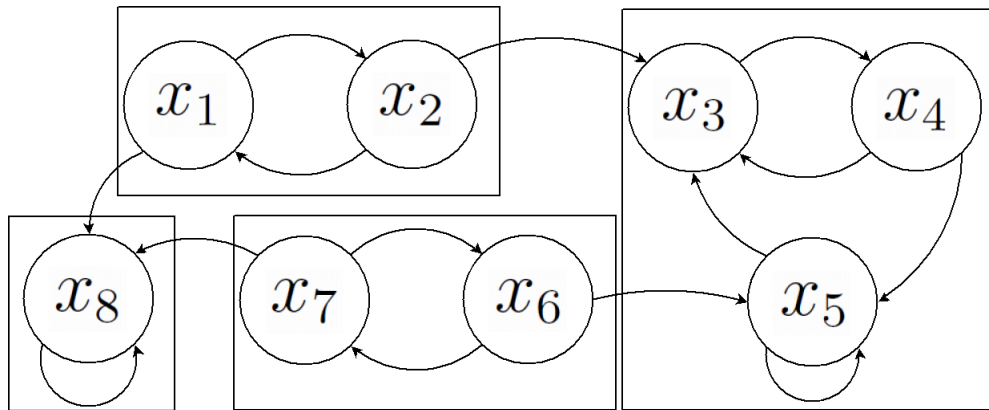


Figura 3.1: Grafo di stato ricavato dalla matrice L_{tot} in (3.8)

Le classi della BCN presa in considerazione sono quattro: $\mathcal{K}_1 = \{x_1, x_2\}$, $\mathcal{K}_2 = \{x_3, x_4, x_5\}$, $\mathcal{K}_3 = \{x_6, x_7\}$ e $\mathcal{K}_4 = \{x_8\}$; di cui due classi transitorie (\mathcal{K}_1 e \mathcal{K}_3) e due chiuse (\mathcal{K}_2 e \mathcal{K}_4). Si può notare che in questo esempio le classi \mathcal{K}_2 e \mathcal{K}_4

sono entrambe accessibili dalle classe transitorie \mathcal{K}_1 e \mathcal{K}_3 a seconda del percorso (e quindi conseguentemente del controllo) scelto, fatto che non poteva accadere per le BN. \diamond

Si noti che, mentre in una BN il valore dei nodi entro un certo numero di passi diventava costante o periodico, in una BCN grazie alla presenza degli ingressi tale periodicità può non esistere. Questo accade anche se la dinamica coinvolge solo una classe chiusa; si consideri a questo proposito il seguente Esempio.

Esempio 3.2.3. Si consideri la seguente BCN a due stati:

$$x(t+1) = Lx(t) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} x(t), \quad (3.9)$$

da cui si può ricavare il seguente grafo di stato riportato in Figura 3.2 da cui si possono ricavare le possibili traiettorie della BCN.

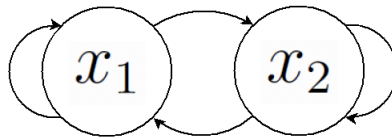


Figura 3.2: Grafo di stato della BCN in (3.9)

Si consideri ora la seguente traiettoria:

$$x_1, x_2, x_1, x_1, x_2, x_2, x_1, x_1, x_1, x_2, x_2, x_2, \dots \quad (3.10)$$

La sequenza in (3.10) risulta una traiettoria plausibile, poiché dal grafo di stato si ricava che entrambi gli stati sono accessibili (in un passo) da se stessi e uno dall'altro, inoltre non è periodica. Questo semplice controesempio mostra quindi che la traiettoria di una BCN può non essere periodica. \diamond

Si faccia comunque attenzione che le differenti classi rappresentano solo la possibilità o meno di passare da un gruppo di stati comunicanti ad un altro; la dinamica reale potrebbe coinvolgere solo un numero molto ridotto di stati di una classe togliendo all'operazione di partizione gran parte dell'utilità.

3.3 Controllabilità

L'introduzione degli ingressi in una Boolean network, come si è visto, elimina l'unicità della traiettoria (dato uno stato iniziale) e apre nuove possibilità di studio. Infatti in questo caso si può influire sulla dinamica della rete cercando di ottenere l'andamento desiderato per il valore dei nodi. Per poter ottenere questi obiettivi si deve innanzitutto studiare quali siano effettivamente gli stati raggiungibili a partire da ogni condizione iniziale. Questo rappresenta un problema affrontato diffusamente nella Teoria dei Sistemi: la controllabilità. In questa sezione si vuole quindi contestualizzare la controllabilità per quanto riguarda le BCN e fornire i criteri per studiarla.

3.3.1 Definizioni di controllabilità

Come si è già evidenziato, uno dei vantaggi principali nel porre una BCN in forma di stato è quello di poter usare gli strumenti della Teoria dei Sistemi; si può quindi riportare in modo naturale la definizione di controllabilità anche in questo ambito.

Un ruolo fondamentale in questo studio viene attribuito alla matrice L_{tot} il cui significato è strettamente correlato all'esistenza di ingressi che permettano di passare da uno stato ad un altro.

Definizione 3.3.1. Si consideri la BCN in (3.1), la relativa forma algebrica in (3.3) e due stati $x_a, x_b \in \Delta^{2^n}$. La BCN viene detta controllabile da x_a a x_b in k passi se esiste una sequenza di controllo $u(t) \in \mathcal{B}^m$, $t = 0, 1, \dots, k-1$, tale che ponendo come stato iniziale $x(0) = x_a$ si raggiunge lo stato di destinazione al tempo k , i.e. $x(k) = x_b$. Lo stato x_b viene detto raggiungibile a partire da x_i e l'insieme degli stati raggiungibili da x_a in k passi è indicato con $R_k(x_a) \subseteq \Delta^{2^n}$ ³. Di conseguenza si ha che insieme degli stati raggiungibili da x_a è

$$R(x_a) = \bigcup_{k=1}^{\infty} R_k(x_a).$$

Definizione 3.3.2. Se $R(x_a) \equiv \Delta^{2^n}$ il sistema viene detto globalmente raggiungibile da x_a . Se $x_b \in R(x_a)$, $\forall i \in \{1, \dots, 2^n\}$, il sistema viene detto globalmente controllabile a x_b .

Definizione 3.3.3. Una BCN a n nodi viene detta globalmente controllabile se per ogni coppia di stati $x_a, x_b \in \Delta^{2^n}$, si ha che $x_b \in R(x_a)$.

³Si noti che in questo contesto controllabilità e raggiungibilità esprimono il medesimo concetto da due punti di vista differenti. Precisamente, l'espressione " x_a è controllabile a x_b " è equivalente a " x_b è raggiungibile da x_a ".

3.3.2 Criteri di controllabilità

I criteri per verificare se una BCN è controllabile o meno variano in base ai diversi tipi di ingressi: ingressi come sequenze Booleane libere da vincoli, ingressi ottenuti per retroazione dello stato, ecc. In questo paragrafo ci occupiamo di controllabilità supponendo che gli ingressi siano dati da sequenze Booleane arbitrarie. Sappiamo che se nel grafo di stato esiste un arco dallo stato x_a allo stato x_b , i.e. $(x_a, x_b) \in \mathcal{E}$, allora esiste un ingresso \bar{u} tale che se $x(t) = x_a$ allora $x(t+1) = L(\bar{u})x(t) = x_b$. Si ha dunque che in questo caso l'accessibilità di uno stato x_b definita nel secondo capitolo coincide con il concetto di raggiungibilità dello stato x_b data nel precedente paragrafo. Come conseguenza diretta si ha il seguente risultato.

Proposizione 3.3.4. *Se x_a e x_b sono due stati appartenenti alla medesima classe di comunicazione \mathcal{K} allora $x_b \in R(x_a)$ e $x_a \in R(x_b)$.*

E data la transitività dell'accessibilità/raggiungibilità si ha anche:

Proposizione 3.3.5. *Siano x_a e x_b due stati appartenenti alla medesima classe di comunicazione \mathcal{K} . Se x_c è uno stato tale che $x_c \in R(x_a)$, allora si ha che $x_c \in R(x_b)$.*

Le precedenti considerazioni portano a fissare un limite sul numero di passi in cui uno stato può essere raggiungibile da un altro nel contesto considerato in presenza cioè di un controllo Booleano arbitrario. Precisamente si ha:

Proposizione 3.3.6. *Nella BCN in forma di stato (3.3), se uno stato x_b non è raggiungibile da x_a in al più 2^n passi allora non è raggiungibile da x_a .*

Dimostrazione. Si può procedere dimostrando l'antinominale i.e.: se uno stato x_b è raggiungibile da x_a allora x_b è raggiungibile da x_a in $k \leq 2^n$ passi. Si supponga dunque x_b raggiungibile da x_a e si consideri la più corta sequenza di stati della BCN che porti da x_a a x_b : $x(0) = x_a, x(1), \dots, x(k) = x_b$. Per $0 < t \leq k$ si ha $x(t) \neq x_a$ e dati $0 < t_1 < t_2 \leq k$ si ha che $x(t_1) \neq x(t_2)$ altrimenti la sequenza considerata non sarebbe la più corta. Si supponga ora $x_a \neq x_b$; dovendo gli stati della sequenza definita prima essere tutti distinti ed essendo 2^n gli stati della BCN, si ha che $k \leq 2^n - 1$. Se invece $x_a = x_b$ si ottiene $k \leq 2^n$. In generale si ha quindi che x_b è raggiungibile da x_a in $k \leq 2^n$ passi. \square

Si può quindi, in questo caso, modificare la definizione di insieme degli stati raggiungibili da x_a che diventa:

$$R(x_a) = \bigcup_{k=1}^{2^n} R_k(x_a).$$

Si è visto quindi che il concetto di controllabilità è legato al grafo degli stati; essendo quest'ultimo ricavabile dalla matrice L_{tot} , si possono trovare le proprietà di questa matrice che determinano la controllabilità di una BCN. Dati due stati $x_a, x_b \in \Delta^{2^n}$ e un intero $k > 0$ si indichi con $l(k; x_a, x_b)$ il numero di diverse sequenze di controllo che consentono di portare la dinamica di una BCN da $x(0) = x_a$ a $x(k) = x_b$. In [29] si dimostra che la quantità $l(k; x_a, x_b)$, la cui utilità è rilevante per la controllabilità o meno di un sistema, è calcolabile mediante la matrice L_{tot} . Più precisamente si ha il seguente Teorema.

Teorema 3.3.7. *Il numero di differenti sequenze di controllo che portano una BCN dallo stato x_a allo stato x_b in k passi è:*

$$l(k; x_a, x_b) = x_b^T L_{tot}^k x_a.$$

Dimostrazione. Siano $u^1, u^2, \dots, u^{l(k; x_a, x_b)}$ le sequenze di controllo distinte che portino il sistema (3.3) da $x(0) = x_a$ a $x(k) = x_b$. In altre parole si ha che $\forall i \in \{1, \dots, l(k; x_a, x_b)\}$:

$$x_b = L \times u^i(k-1) \times \dots \times L \times u^i(0) \times x_a. \quad (3.11)$$

Poiché si ha che le differenti sequenze di controllo ammissibili di lunghezza k sono 2^{mk} , questo implica l'esistenza di $t(k; x_a, x_b) \triangleq 2^{mk} - l(k; x_a, x_b)$ sequenze w^j distinte tali che:

$$x_b \neq L \times w^j(k-1) \times \dots \times L \times w^j(0) \times x_a \quad j \in \{1, \dots, t(k; x_a, x_b)\}. \quad (3.12)$$

Si indichi con x_c^j la parte destra della disuguaglianza (3.12), i.e. $x_c^j \triangleq L \times w^j(k-1) \times \dots \times L \times w^j(0) \times x_a$ si noti che $x_c^j \in \Delta^{2^n}$. Dunque moltiplicando a sinistra per x_b^T le due espressioni a destra in (3.11) e (3.12) si ottiene:

$$\begin{aligned} 1 &= x_b^T L \times u^i(k-1) \times \dots \times L \times u^i(0) \times x_a, & i \in \{1, \dots, l(k; x_a, x_b)\} \\ 0 &= x_b^T L \times w^j(k-1) \times \dots \times L \times w^j(0) \times x_a, & j \in \{1, \dots, t(k; x_a, x_b)\}. \end{aligned}$$

Sommando queste $l(k; x_a, x_b) + t(k; x_a, x_b) = 2^{mk}$ equazioni si ottiene:

$$\begin{aligned} l(k; x_a, x_b) &= x_b^T L \times 1_{2^m} \times \dots \times L \times 1_{2^m} \times x_a = \\ &= x_b^T L_{tot}^k x_a, \end{aligned}$$

in cui l'ultimo passaggio è dato dall'espressione alternativa di L_{tot} in (3.1.2) e si conclude così la dimostrazione. \square

Il legame tra $l(k; x_a, x_b)$ e $x_b^T L_{tot}^k x_a$ introdotto nel Teorema 3.3.7 porta al seguente risultato.

Corollario 3.3.8. Per ogni $k > 0$, la somma degli elementi di ogni colonna di L_{tot}^k è pari a 2^{mk} .

Dimostrazione. La somma degli elementi nella i -esima colonna di L_{tot}^k può essere espressa come:

$$\sum_{j=1}^{2^n} (\delta_{2^n}^j)^T L_{tot}^k \delta_{2^n}^i.$$

Dal Teorema 3.3.7, questa espressione equivale al numero di controlli distinti che portano una BCN da $x(0) = \delta_{2^n}^i$ ad uno stato dell'insieme $\{\delta_{2^n}^1, \dots, \delta_{2^n}^{2^n}\}$ in k passi. Questo non è altro però che il numero totale delle sequenze di controllo distinte di lunghezza k che è sempre uguale a 2^{km} . \square

Grazie al Teorema (3.3.7) ed alle considerazioni sul grafo di stato, le condizioni necessarie e sufficienti per la controllabilità di una BCN si possono riassumere con il seguente risultato.

Teorema 3.3.9. Si consideri una BCN a n stati riportata in (3.1) e la corrispondente forma in spazio di stato (3.3), da cui si possono ottenere le matrici L e L_{tot} e il relativo grafo degli stati. Allora si ha che:

1. Lo stato $x_b = \delta_{2^n}^b$ è raggiungibile da $x_a = \delta_{2^n}^a$ in k passi se e solo se

$$[L_{tot}^k]_{b,a} > 0,$$

o equivalentemente esiste, sul grafo di stato, un cammino orientato formato da k archi che va dallo stato x_a allo stato x_b .

2. Lo stato $x_b = \delta_{2^n}^b$ è raggiungibile da $x_a = \delta_{2^n}^a$ se e solo se

$$\left[\sum_{k=1}^{2^n} L_{tot}^k \right]_{b,a} > 0,$$

o equivalentemente esiste, sul grafo di stato, un cammino orientato che va dallo stato x_a allo stato x_b .

3. La BCN è globalmente raggiungibile da x_a se e solo se

$$\sum_{k=1}^{2^n} \text{Col}_a (L_{tot}^k) \gg 0,$$

o equivalentemente, sul grafo di stato, esiste un cammino orientato per ogni stato x_b del sistema (3.3) che porta dallo stato x_a allo stato x_b .

4. La BCN è globalmente controllabile a x_b se e solo se

$$\sum_{k=1}^{2^n} \text{Row}_b (L_{tot}^k) \gg 0,$$

i.e. lo stato $x_b \in \mathcal{K}$ con \mathcal{K} unica classe chiusa della BCN. Sul grafo di stato questa condizione è equivalente all'esistenza di un cammino orientato da ogni stato x_a allo stato x_b .

5. La BCN è globalmente controllabile se e solo se

$$\sum_{k=1}^{2^n} L_{tot}^k \gg 0,$$

i.e. la matrice L_{tot} è irriducibile. La condizione equivalente sul grafo di stato consiste nell'esistenza, da ogni stato x_a , di un cammino orientato per ogni altro stato x_b , i.e. tutti gli stati appartengono alla medesima classe di comunicazione.

Esempio 3.3.10. Si consideri la BCN ad un ingresso rappresentata algebricamente come in (3.3) con matrice di transizione:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

da cui si ottiene:

$$L_{tot} = \begin{bmatrix} 2 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Si supponga ora che $x(0) = x_8 = \delta_8^8$ e di voler trovare il numero di sequenze di controllo in 3 passi da x_8 a x_6 i.e. $l(3; \delta_8^8, \delta_6^6)$. Per prima cosa si calcola la matrice:

$$L_{tot}^3 = \begin{bmatrix} 8 & 2 & 4 & 2 & 8 & 2 & 4 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \end{bmatrix},$$

da cui si ottiene:

$$l(3; \delta_8^8, \delta_6^6) = x_6^T L_{tot}^3 x_8 = [L_{tot}^3]_{6,8} = 1.$$

Per trovare l'insieme degli stati raggiungibili da x_8 applicando il Teorema 3.3.9 si ha:

$$\sum_{k=1}^8 \text{Col}_8 [L_{tot}^k] = \begin{bmatrix} 336 \\ 0 \\ 20 \\ 20 \\ 33 \\ 33 \\ 34 \\ 34 \end{bmatrix},$$

da cui si ottiene che l'unico stato non raggiungibile da x_8 è x_2 . ◇

Esempio 3.3.11. Si consideri ora la seguente BCN:

$$x(t+1) = Lx(t) = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} x(t), \quad (3.13)$$

in cui

$$L_{tot} = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

Dal grafo di stato in Figura 3.3 si può notare che esiste una sola classe chiusa $\mathcal{K}_c = \{x_1, x_3, x_5\}$. Dal Teorema 3.3.9 si ha quindi che il sistema è globalmente controllabile a x_i , $i = 1, 3, 5$.

Questo si poteva ugualmente vedere considerando:

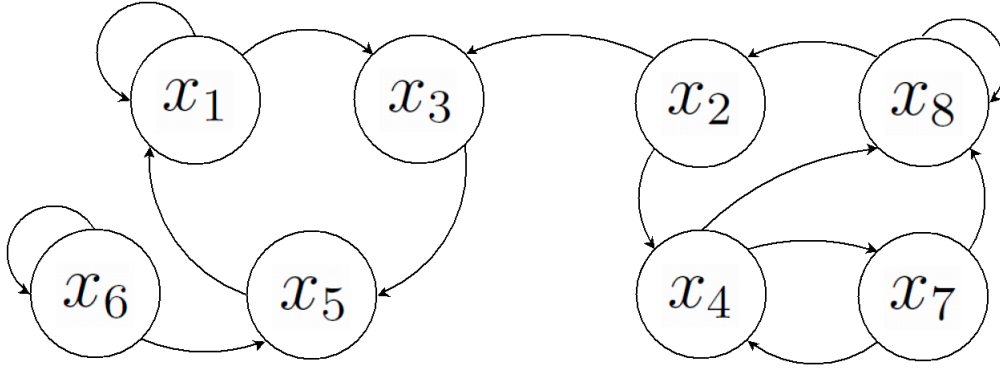


Figura 3.3: Grafo di stato dell BCN in (3.13)

$$\sum_{k=1}^8 [L_{tot}^k] = \begin{bmatrix} 252 & 160 & 252 & 100 & 264 & 248 & 100 & 144 \\ 0 & 36 & 0 & 64 & 0 & 0 & 64 & 49 \\ 132 & 105 & 120 & 76 & 126 & 122 & 76 & 88 \\ 0 & 31 & 0 & 54 & 0 & 0 & 54 & 41 \\ 126 & 96 & 138 & 72 & 120 & 132 & 72 & 80 \\ 0 & 0 & 0 & 0 & 0 & 8 & 0 & 0 \\ 0 & 18 & 0 & 31 & 0 & 0 & 31 & 23 \\ 0 & 64 & 0 & 113 & 0 & 0 & 113 & 85 \end{bmatrix},$$

da cui si ricava che $\sum_{k=1}^8 \text{Row}_i [L_{tot}^k] \gg 0$ per $i = 1, 3, 5$. \diamond

Si noti che la condizione $x_b \in R_k(x_a)$ non implica che $x_b \in R_{k+1}(x_a)$; quindi x_b può essere raggiungibile da x_a in k passi ma non esserlo in $k + 1$. Per ovviare a questo problema Laschov e Margaliot hanno presentato in [29] un'altra definizione di controllabilità che in questo contesto chiameremo controllabilità globale forte.

3.3.3 Controllabilità globale forte

Definizione 3.3.12. La BCN (3.3) è globalmente controllabile in senso forte (g.c.f.) in k passi se $\forall x_a, x_b \in \Delta^{2^n}$ esiste una sequenza di controllo di lunghezza k tale che partendo da $x(0) = x_a$ si abbia $x(k) = x_b$.

Osservazione 3.3.13. La definizione appena fornita differisce in modo significativo dalla Definizione 3.3.3 seppur si pretenda in entrambe che da $x(0) = x_a$ si arrivi in $x(k) = x_b$. Infatti nella Definizione 3.3.3 ad ogni coppia di stati è associato un

valore k opportuno, mentre nella Definizione 3.3.12 un unico k deve valere per ogni coppia di stati.

Dalla Definizione 3.3.12 e dal Teorema 3.3.7 si ottiene direttamente il seguente risultato.

Corollario 3.3.14. La BCN (3.1) è globalmente controllabile in senso forte in k passi se e solo se tutti gli elementi della matrice L_{tot}^k sono diversi da zero.

Dimostrazione. Si supponga che l'elemento in posizione (i, j) della matrice L_{tot}^k sia nullo. Ne consegue che $(\delta_{2^n}^i)^T L_{tot}^k \delta_{2^n}^j = 0$ e per il Teorema 3.3.7 non esistono controlli che portano la BCN da $\delta_{2^n}^j$ a $\delta_{2^n}^i$ in k passi. Si ha quindi che la BCN non è g.c.f. in k passi.

Per dimostrare la sufficienza dell'enunciato si noti preliminarmente che il Teorema 3.3.7 implica $L_{tot} \geq 0$. Quindi se tutti gli elementi di L_{tot}^k sono diversi da zero devono necessariamente essere strettamente positivi e quindi $L_{tot}^k \gg 0$. Questo implica che $(\delta_{2^n}^i)^T L_{tot}^k \delta_{2^n}^j > 0$ per ogni $i, j \in \{1, \dots, 2^n\}$, e per il Teorema 3.3.7 che la BCN è g.c.f. in k passi. \square

Una conseguenza fondamentale della controllabilità in senso forte è data dal seguente Corollario.

Corollario 3.3.15. Se esiste un valore k finito tale che la BCN sia globalmente controllabile in senso forte in k passi, allora la BCN è g.c.f. in j passi $\forall j \geq k$.

Dimostrazione. Si supponga che la BCN sia g.c.f. in k passi e quindi per il Corollario 3.3.14 si ha che la matrice L_{tot}^k è strettamente positiva. Si assuma ora per assurdo che l'elemento in posizione (i, j) della matrice L_{tot}^{k+1} sia nullo. Si ha quindi che il prodotto della riga i -esima della matrice L_{tot}^k e della colonna j -esima di L_{tot} è zero. Dato che $L_{tot}^k \gg 0$ e $L_{tot} \geq 0$, allora ogni valore della colonna j -esima di L_{tot} deve essere nullo ma se fosse così ovviamente L_{tot} non potrebbe essere strettamente positiva. Questa contraddizione dimostra che se L_{tot}^k è strettamente positiva lo deve essere anche L_{tot}^{k+1} . \square

Questo risultato giustifica la definizione seguente.

Definizione 3.3.16. La BCN (3.1) è globalmente controllabile in senso forte se esiste un k finito tale che la BCN sia g.c.f. in k passi.

Da questo deriva una condizione necessaria e sufficiente per la g.c.f. di una BCN.

Teorema 3.3.17. La BCN (3.1) è g.c.f. se e solo se la matrice L_{tot} ottenuta dalla forma in spazio di stato (3.3) è una matrice primitiva.

La globale controllabilità in senso forte di una BCN è una condizione desiderabile perché lascia un'ampia libertà di configurazione della rete all'utente però è fortemente restrittiva. Infatti si possono comunque ottenere risultati interessanti e globali sulla stabilizzabilità anche per reti che non abbiano questa proprietà come si vedrà nel prossimo paragrafo.

3.4 Stabilizzabilità

La possibilità di gestire gli ingressi esterni può essere utilizzata per stabilizzare una Boolean control network, i.e. per portare il valore dei nodi della rete ad essere costante da un certo istante in poi. Anche in questa sezione gli ingressi a cui si farà riferimento sono sequenze Booleane arbitrarie.

Definizione 3.4.1. Una BCN è detta stabilizzabile se esiste uno stato x_s tale che, per ogni stato iniziale $x(0) = \delta_{2^n}^i$, esiste una sequenza di ingresso u^i che porta lo stato in x_s in un numero finito di passi e ve lo fa rimanere indefinitamente.

3.4.1 Punti di equilibrio ad ingresso costante

Per poter stabilizzare il sistema (3.3) ad uno stato x_s deve esistere un ingresso costante tale che la dinamica, partendo dallo stato x_s , ci rimanga indefinitamente. Più precisamente si ha la seguente Definizione.

Definizione 3.4.2. Uno stato x_s di una BCN a m ingressi viene definito un punto di equilibrio ad ingresso costante (o semplicemente punto di equilibrio) se esiste un ingresso $\bar{u} \in \Delta^{2^m}$ tale che se $x(t) = x_s$ si ha $x(t+1) = L(\bar{u})x(t) = x_s$.

Dalla matrice L_{tot} si può dedurre se uno stato x_s è punto di equilibrio ad ingresso costante o meno; si ha infatti:

Proposizione 3.4.3. Uno stato $x_s = \delta_{2^n}^i$ di una BCN è un punto di equilibrio ad ingresso costante se e solo se $[L_{tot}]_{i,i} > 0$.

Dimostrazione. La matrice di transizione si può sempre esprimere, come si è visto in (3.6), affiancando le 2^m matrici $L_j \in \mathcal{L}^{2^n \times 2^n}$ con $j = 1, \dots, 2^m$. L'affermazione “ $\delta_{2^n}^i$ è punto di equilibrio ad ingresso costante” corrisponde al fatto che esista almeno una matrice L_j in cui l'elemento (i, i) sia uguale a 1. Infatti se $\exists \bar{j}$ tale che $[L_{\bar{j}}]_{i,i} = 1$ allora ponendo $x(t) = x_i$ e prendendo come ingresso $u_{\bar{j}} = \delta_{2^m}^{\bar{j}}$ si ha che $x(t+1) = L_{\bar{j}}x_i = x_i$. Poiché $L_{tot} = \sum_{j=1}^{2^m} L_j$ si ha che se $\exists \bar{j}$ tale che $[L_{\bar{j}}]_{i,i} = 1$ allora $[L_{tot}]_{i,i} > 0$ concludendo la dimostrazione. \square

Anche dal grafo degli stati si può dedurre se uno stato x_s è punto di equilibrio ad ingresso costante o meno; si ha infatti che uno stato x_s è punto di equilibrio se e solo se uno dei suoi archi uscenti si richiude su x_s .

Ad ogni punto di equilibrio è associato un corrispondente insieme di stati, detto insieme controllabile al punto di equilibrio.

Definizione 3.4.4. Si consideri un punto di equilibrio ad ingresso costante x_s del sistema riportato in (3.3). L'insieme $S_s = \{x : x_s \in R(x)\}$ è detto insieme controllabile al punto di equilibrio x_s .

Inizializzando la dinamica della BCN in $x(0) = \bar{x} \in S_s$ esisterà sicuramente una sequenza di controllo tale che il sistema converga in x_i . Si noti che in questo caso, diversamente dalle BCN, uno stato può appartenere all'insieme controllabile di più punti di equilibrio. Dalla Definizione (3.4.4) segue che gli stati comunicanti appartengono al medesimo insieme controllabile del punto di equilibrio e quindi:

Proposizione 3.4.5. *Data una BCN descritta come in (3.3), siano x_a, x_b due stati appartenenti alla medesima classe di comunicazione \mathcal{K} . Se $x_a \in S_s$, insieme controllabile del punto di equilibrio x_s , allora anche $x_b \in S_s$.*

Dimostrazione. Si supponga per assurdo che $x_b \notin S_s$ e quindi per la Definizione 3.4.4 $x_b \notin R(x_s)$. Poiché $x_a \in R(x_s)$ e x_b e x_a appartengono alla stessa classe di comunicazione si ha che $x_b \in R(x_s)$ in contraddizione con l'ipotesi assurda. \square

Se $x_a \in S_s$ e $x_a \in \bar{\mathcal{K}}$ classe di comunicazione, $\bar{\mathcal{K}}$ è detta classe stabilizzabile a x_s . Si ha dunque che l'insieme controllabile del punto di equilibrio x_s può sempre essere espresso nella forma:

$$S_s = \bigcup_j \mathcal{K}_j,$$

in cui \mathcal{K}_j sono le classi stabilizzabili a x_s .

In una BCN globalmente stabilizzabile esiste un insieme S_s per un punto di equilibrio x_s che coincide con l'intero spazio degli stati i.e. $S_s \equiv \Delta^{2^n}$.

In un sistema globalmente stabilizzabile esiste, dunque, una sequenza di ingressi tale che la traiettoria converga ad un stato x_s , a prescindere dallo stato di partenza.

3.4.2 Criteri per la globale stabilizzabilità

In questo paragrafo si vogliono definire i criteri per poter verificare se una BCN è globalmente stabilizzabile. Affinché la traiettoria converga ad uno stato x_s partendo da qualsiasi stato iniziale è necessario che esista un ingresso che porti e mantenga la traiettoria nello stato x_s . Più precisamente si ha il seguente Teorema.

Teorema 3.4.6. *La BCN in (3.3) è globalmente stabilizzabile se sussistono le due seguenti condizioni:*

1. *esiste uno stato $x_s \in \Delta^{2^n}$ punto di equilibrio ad ingresso costante;*
2. *la BCN è globalmente controllabile a x_s .*

Esempio 3.4.7. Si consideri ancora la BCN in (3.13) introdotta nell'Esempio 3.3.11 in cui è stato dimostrato che il sistema è globalmente controllabile a x_i $i = 1, 3, 5$. Si riporta ancora qui di seguito la matrice L_{tot} del sistema in questione:

$$L_{tot} = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix};$$

poiché $[L_{tot}]_{1,1} > 0$ si ha che x_{11} è un punto di equilibrio ad ingresso costante e quindi, essendo soddisfatte le condizioni del Teorema 3.4.6, si ha che la BCN in (3.13) è globalmente stabilizzabile nello stato x_1 . \diamond

In caso di sequenza di ingresso costante le due condizioni per la globale stabilizzabilità possono essere riscritte, infatti in questo caso la dinamica della rete obbedisce alle funzioni di uno solo dei 2^m sottosistemi e quindi il sistema diventa una Boolean network. Precisamente dato un controllo costante $\bar{u} = \delta_{2^m}^i$ si ha che la forma di stato in (3.3) diventa $x(t+1) = L \times \bar{u} \times x(t) = L_i x(t)$ che corrisponde ad un BN con matrice di transizione $L_i \in \mathcal{L}^{2^n \times 2^n}$. La globale stabilizzabilità in questo caso si ottiene se una delle 2^m BN è globalmente stabile. Dato il Teorema 2.4.5 si ottiene:

Corollario 3.4.8. Il sistema (3.3) è stabilizzabile mediante un ingresso costante se e solo se tra le matrici

$$L_i^k, \quad 1 \leq k \leq 2^n,$$

ne esiste una che abbia le colonne tutte uguali i.e. $\exists \bar{i}, \bar{k} : L_{\bar{i}}^{\bar{k}} = \delta[j, \dots, j]$ con $j \in \{1, \dots, 2^n\}$. Il corrispondente controllo stabilizzante è dato da $\bar{u} = \delta_{2^m}^{\bar{i}}$ che stabilizza la dinamica nello stato $x_s = \delta_{2^n}^{\bar{j}}$.

Esempio 3.4.9. Si consideri una BCN in forma (3.3) con $m = 2$ con la matrice di transizione espressa in forma (3.6) i.e.

$$\begin{aligned} L &= [L_1 \mid L_2 \mid L_3 \mid L_4] \\ &= \left[\begin{array}{cccc|cccc|cccc|cccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right]. \end{aligned}$$

Dato che:

$$L_3^2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

per il Corollario 3.4.8 la BCN è globalmente stabilizzabile mediante l'ingresso costante $u_3 = \delta_4^3$. \diamond

Si consideri ora una BCN globalmente controllabile, in cui la matrice L_{tot} corrispondente è una matrice irriducibile. In questo caso particolare la seconda condizione del Teorema 3.4.6 è sempre valida e si ha quindi il seguente risultato.

Corollario 3.4.10. Una BCN globalmente controllabile è globalmente stabilizzabile ad ogni stato $x_s = \delta_{2^n}^i$ per cui $[L_{tot}]_{i,i} > 0$ i.e. ad ogni stato x_s che sia stabilizzabile.

3.5 Sequenze di ingresso

Nei paragrafi precedenti si sono identificate le proprietà che devono essere soddisfatte da una BCN per poter essere definita controllabile o stabilizzabile. Se queste proprietà sono soddisfatte esiste quindi una sequenza Booleana che, posta

come ingresso, controlla/stabilizza la BCN. Per procedere al controllo della rete è indispensabile la determinazione della sequenza di ingresso, problema che sarà affrontato in questa sezione. Precisamente, dati $x_i, x_j \in \Delta^{2^n}$ e supponendo che $x_j \in R(x_i)$, si vuole identificare una sequenza di controllo che porti la traiettoria di una BCN da x_i a x_j . Per poter procedere bisogna preliminarmente introdurre un nuovo strumento di studio: la matrice di controllo in s passi, $P^{(s)}$.

3.5.1 Matrice di controllo e sue proprietà

Definizione 3.5.1. Data una BCN in forma di stato (3.3) e un intero positivo $s \in \mathbb{N}$, si definisce matrice di controllo in s passi della BCN, $P^{(s)}$, la matrice ottenuta attraverso il seguente modo ricorsivo:

$$\begin{cases} P^{(1)} = L \\ P^{(s)} = L_{tot}P^{(s-1)} \end{cases}, \quad s > 1. \quad (3.14)$$

in cui, $\forall s \geq 1$, le matrici $P^{(s)}$ vanno interpretate come matrici in $\mathbb{Z}^{2^n \times 2^{n+m}}$.

Si consideri ora la matrice di controllo $P^{(s)}$ per un generico $s \in \mathbb{N}$; risolvendola iterativamente si ottiene:

$$\begin{aligned} P^{(s)} &= L_{tot}P^{(s-1)} = L_{tot}^2P^{(s-2)} = \dots = \\ &= L_{tot}^iP^{(s-i)} = \dots = L_{tot}^{s-1}P^{(1)} = L_{tot}^{s-1}L. \end{aligned}$$

Si può dunque dare una definizione alternativa di matrice di controllo in s passi:

$$P^{(s)} = L_{tot}^{s-1}L, \quad (3.15)$$

che permette di evitare di dover calcolare le matrici precedenti quella desiderata. Essendo $L_{tot}^0 = I$ si ha che $P^{(1)} = IL = L$ e quindi la formula in (3.15) vale anche per $s = 1$.

La matrice $P^{(s)}$, $\forall s \in \mathbb{N}$, può essere riscritta, mettendo in evidenza i blocchi che la compongono, nel seguente modo:

$$P^{(s)} = \begin{bmatrix} P_1^{(s)} & \dots & P_i^{(s)} & \dots & P_{2^m}^{(s)} \end{bmatrix} \quad (3.16)$$

con $P_i^{(s)} \in \mathbb{Z}^{2^n \times 2^n}$.

Si ha che i blocchi $P_i^{(s)}$ possono essere calcolati singolarmente grazie alla seguente Proposizione.

Proposizione 3.5.2. *Si consideri una BCN in forma di stato in (3.3) con matrice di transizione L composta dalle matrici L_i $i = 1, \dots, 2^m$ come in (3.6). Allora i 2^m blocchi $P_i^{(s)}$ della matrice di controllo in s passi $P^{(s)}$, presentati in (3.16), soddisfano la seguente equazione:*

$$P_i^{(s)} = L_{tot}^s L_i. \quad (3.17)$$

Dimostrazione. Dall'equazione (3.15), scomponendo la matrice L nei suoi diversi blocchi L_i si ottiene:

$$\begin{aligned} P^{(s)} &= L_{tot}^{s-1} L = L_{tot}^{s-1} [L_1 \quad \dots \quad L_i \quad \dots \quad L_{2^m}] = \\ &= [L_{tot}^{s-1} L_1 \quad \dots \quad L_{tot}^{s-1} L_i \quad \dots \quad L_{tot}^{s-1} L_{2^m}], \end{aligned}$$

ricavando in questo modo l'equazione (3.17). □

Una proprietà molto importante delle sottomatrici $P_i^{(s)}$ è data dalla seguente Proposizione.

Proposizione 3.5.3.

$$L_{tot}^s = \sum_{i=1}^{2^m} P_i^{(s)}.$$

Dimostrazione. Svolgendo la sommatoria e applicando la Proposizione 3.5.2 si ottiene:

$$\begin{aligned} \sum_{i=1}^{2^m} P_i^{(s)} &= P_1^{(s)} + \dots + P_{2^m}^{(s)} = \\ &= L_{tot}^{s-1} L_1 + \dots + L_{tot}^{s-1} L_{2^m} = \\ &= L_{tot}^{s-1} (L_1 + \dots + L_{2^m}) = \\ &= L_{tot}^{s-1} L_{tot} = \\ &= L_{tot}^s, \end{aligned}$$

e quindi $L_{tot}^s = \sum_{i=1}^{2^m} P_i^{(s)}$. □

Dalle Proposizioni 3.5.2 e 3.5.3 si può ricavare il significato dei vari blocchi $P_\alpha^{(s)}$, $\alpha = 1, \dots, 2^m$, della matrice di controllo. Infatti l'elemento $[P_\alpha^{(s)}]_{j,i}$ indica il numero di sequenze di controllo differenti che portano in s passi dallo stato $x_i = \delta_{2^n}^i$ allo stato $x_j = \delta_{2^n}^j$ con l'ingresso all'istante 0 pari a $u_\alpha = \delta_{2^m}^\alpha$ i.e. $u(0) = u_\alpha$. La somma $\sum_{\alpha=1}^{2^m} [P_\alpha^{(s)}]_{j,i}$ per la Proposizione 3.5.3 è pari a $[L_{tot}^s]_{j,i}$ e quindi corrisponde al numero di ingressi che in s passi portano da x_i a x_j in accordo con quanto affermato precedentemente. In particolare dato il sistema in (3.3) e le matrici $P_\alpha^{(s)}$, $\alpha = 1, \dots, 2^m$, si ha che:

1. Lo stato $x_j = \delta_{2^n}^j$ è raggiungibile da $x_i = \delta_{2^n}^i$ in s passi se e solo se

$$\sum_{\alpha=1}^{2^m} [P_\alpha^{(s)}]_{j,i} > 0;$$

2. Lo stato $x_j = \delta_{2^n}^j$ è raggiungibile da $x_i = \delta_{2^n}^i$ se e solo se

$$\sum_{s=1}^{2^n} \sum_{\alpha=1}^{2^m} [P_\alpha^{(s)}]_{j,i} > 0.$$

3.5.2 Control design

La caratteristica fondamentale della matrice $P^{(s)}$ è quella di conservare l'informazione sugli ingressi ed è quindi indispensabile per poter definire una sequenza di ingressi per ottenere la traiettoria desiderata. In particolare si ha che dato $x(0) = x_i = \delta_{2^n}^i$ e supponendo che l'elemento (j, i) della matrice $P_\alpha^{(1)}$ sia positivo, i.e. $[P_\alpha^{(1)}]_{j,i} > 0$, ponendo come ingresso al sistema $u(0) = u_\alpha = \delta_{2^m}^\alpha$ si ottiene che $x(1) = x_j = \delta_{2^n}^j$. Si considerino ora due stati $x_i, x_j \in \Delta^{2^n}$ di una Boolean control network con x_j raggiungibile da x_i i.e. $x_j \in R(x_i)$; poiché in generale la sequenza di ingressi che porta la traiettoria di una BCN da x_i a x_j non è unica, si vuole determinare una tra le sequenze di lunghezza minima. Supponendo $x(0) = x_i$, si può trovare una sequenza di controllo di lunghezza minima che porti la BCN in x_j procedendo con il seguente criterio:

1. Si trovi il più piccolo esponente $s \leq 2^n$ tale che $[L_{tot}^s]_{j,i} > 0$. Si ha quindi che x_j è raggiungibile in s passi ed $\exists \alpha$ tale che

$$[P_\alpha^{(s)}]_{j,i} > 0,$$

per le considerazioni espresse precedentemente si può porre $u(0) = \delta_{2^m}^\alpha$ e $x(s) = x_j$. Se $s = 1$ la sequenza di ingressi è determinata, altrimenti si prosegue con il punto 2 per trovare stati e controlli intermedi.

2. Si trovino k, β tali che

$$\left[P_\beta^{(1)} \right]_{j,k} > 0 \quad \left[P_\alpha^{(s-1)} \right]_{k,i} > 0.$$

Si ha quindi $u(s-1) = \delta_{2m}^\beta$ e $x(s-1) = \delta_{2n}^k$.

3. Se $s-1 = 1$ la sequenza è interamente determinata. In caso contrario si devono aggiornare i valori di s e i ponendo $s := s-1$ e $i := k$ e riprendere dal punto 2.

Esempio 3.5.4. Si consideri una BCN in forma di stato con matrice di transizione

$$L = \left[L_1 \mid L_2 \right]$$

$$= \left[\begin{array}{cccccccc|cccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{array} \right],$$

in cui $L \in \mathcal{L}^{8 \times 16}$, dal momento che $n = 3$ e $m = 1$.

Dalla matrice L si ricava la matrice L_{tot} ottenendo:

$$L_{tot} = \left[\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{array} \right] \quad (3.18)$$

Dato che L_{tot} è una matrice irriducibile la BCN è globalmente controllabile. Si supponga ora di voler trovare la sequenza di controllo che partendo dallo stato x_1 porti allo stato x_3 . Dato che $[L_{tot}]_{3,1} = 0$ ma $[L_{tot}]_{3,1}^2 > 0$ esiste una sequenza di ingressi tale che partendo da $x(0) = x_1$ si ha $x(2) = x_3$. Seguendo il criterio esposto, da $[P_2^{(2)}]_{3,1} > 0$ si ricava $u(0) = u_2 = \delta_2^2$. Poiché $[P_1^{(1)}]_{3,5} = [L_1]_{3,5} > 0$ e $[P_1^{(1)}]_{5,1} > 0$, si ottiene $x(1) = 5$ e $u(1) = u_1 = \delta_1^1$. La sequenza di ingressi u_2, u_1 porta dunque come richiesto la dinamica della BCN da x_1 a x_3 . \diamond

Grazie al criterio presentato, è dunque possibile trovare una sequenza di ingressi mediante la quale la BCN passa da uno stato ad un altro raggiungibile dal primo. In questo modo si possono anche trovare le sequenze per stabilizzare una BCN che soddisfi le proprietà viste nella sezione precedente. Per ogni stato della rete si può infatti determinare la sequenza di ingresso che porti la dinamica nello stato stabilizzabile. Inoltre dato che gli stati appartenenti ad una stessa classe di comunicazione sono tutti raggiungibili uno dall'altro, si possono trovare sequenze di ingresso tali che la BCN visiti periodicamente alcuni (o tutti) gli stati appartenenti alla stessa classe \mathcal{K} , sia essa chiusa o transitoria, a condizione che $R(x(0)) \subseteq \mathcal{K}$.

3.5.3 Controllo in retroazione

Si supponga ora che la BCN in (3.3) sia globalmente stabilizzabile ad uno stato x_s . Si ha quindi che partendo da un qualunque stato $x_i \in \Delta^{2^n}$ esiste una sequenza di ingressi che porta la traiettoria della BCN nello stato x_s . In questo paragrafo si vuole dimostrare che la stabilizzazione di una tale rete può avvenire anche attraverso un controllo in retroazione, i.e. esprimendo l'ingresso nella forma:

$$u(t) = Kx(t) \quad (3.19)$$

con $K \in \mathcal{L}^{2^m \times 2^n}$. La stabilizzazione mediante retroazione è molto utilizzata nella Teoria dei Sistemi e in questo contesto permette di stabilizzare la BCN partendo da qualsiasi stato. Si consideri il seguente Lemma.

Lemma 3.5.5. *Siano x_i, x_j due stati della BCN in forma di stato (3.3) e sia $x_j \in R(x_i)$, allora esiste una matrice $K \in \mathcal{L}^{2^m \times 2^n}$ e un intero s tali che, ponendo $x(0) = x_i$ e applicando l'ingresso $u(t)$ in forma (3.19), si ha che $x(s) = x_j$.*

Dimostrazione. Dato che $x_j \in R(x_i)$ allora esiste una sequenza che dallo stato iniziale $x(0)$ porta la dinamica della BCN nello stato x_j . Come si è visto nel paragrafo precedente è sempre possibile costruire una sequenza di controllo di lunghezza minima $u(0), \dots, u(s-1)$ tale che $x(s) = x_j$. Supponendo che l'ingresso all'istante zero sia pari a $\delta_{2^m}^\alpha$, i.e. $u(0) = \delta_{2^m}^\alpha$, la colonna i -esima della matrice K deve allora essere uguale al vettore $\delta_{2^m}^\alpha$. Si ha infatti:

$$\begin{aligned} Kx(0) = u(0) &\iff K\delta_{2^n}^i = \delta_{2^m}^\alpha \\ &\iff \text{Col}_i(K) = \delta_{2^m}^\alpha. \end{aligned}$$

Procedendo allo stesso modo per tutti gli istanti $t = 1, \dots, s-1$ si possono costruire s colonne della matrice K desiderata. Se s è minore di 2^n , le colonne non definite

della matrice K possono essere assunte come vettori logici arbitrari ad esempio $\delta_{2^m}^1$. Si noti che la matrice K costruita secondo questo procedimento è ben definita; infatti gli stati nell'insieme $\{x(0), x(1), \dots, x(s-1)\}$ sono distinti poiché, se per assurdo esistessero i valori $t_1, t_2 \in \{0, \dots, s-1\}$ con $t_1 < t_2$ tali che $x(t_1) = x(t_2)$, la sequenza $u(0), u(1), \dots, u(t_1-1), u(t_2), \dots, u(s-1)$ porterebbe la BCN dallo stato x_i allo stato x_j in un numero di passi minore rispetto alla sequenza minima e questo chiaramente non è possibile. Le colonne della matrice K vengano quindi definite una sola volta, inoltre essendo $\text{Col}_l(K) \in \Delta^{2^n} \forall l \in \{1, \dots, 2^n\}$ si ha che K è una matrice logica i.e. $K \in \mathcal{L}^{2^m \times 2^n}$. \square

La stabilizzazione di una BCN tramite un ingresso in retroazione è garantita dal seguente Teorema.

Teorema 3.5.6. *Si consideri una BCN in forma di stato (3.3) globalmente stabilizzabile ad uno stato $x_s = \delta_{2^n}^j \in \Delta^{2^n}$, allora esiste una matrice $K \in \mathcal{L}^{2^m \times 2^n}$ tale che l'ingresso $u(t)$ in forma (3.19) stabilizza la BCN a x_s per ogni stato iniziale $x(0)$.*

Dimostrazione. Si supponga, senza perdita di generalità, $j = 2^n$ i.e. lo stato stabilizzabile sia $\delta_{2^n}^{2^n}$. Si consideri ora la sequenza di stati più breve $x^1(0) = \delta_{2^n}^1, \dots, x^1(s_1) = \delta_{2^n}^{2^n}$ dallo stato $\delta_{2^n}^1$ allo stato $\delta_{2^n}^{2^n}$ e si costruisca l'insieme $C_1 = \{x^1(0) = \delta_{2^n}^1, \dots, x^1(s_1-1)\}$. Si possono allora definire s_1 colonne della matrice di retroazione K seguendo il procedimento illustrato nel Lemma 3.5.5. Allo stesso modo si consideri ora la sequenza più breve $x^2(0) = \delta_{2^n}^2, \dots, x^2(s_2) = \delta_{2^n}^{2^n}$ dallo stato $\delta_{2^n}^2$ allo stato $\delta_{2^n}^{2^n}$ e si costruisca l'insieme $C_2 = \{x^2(0) = \delta_{2^n}^2, \dots, x^2(s_2-1)\}$. Se $C_1 \cap C_2 = \emptyset$ si può procedere a definire ulteriori s_2 colonne di K distinte dalle precedenti. Se invece $\exists t < s_1$ ed $\exists g < s_2$ tali che $x^1(t) = x^2(g)$, si deve sostituire la sequenza da $\delta_{2^n}^2$ a x_{2^n} con $x^2(0) = \delta_{2^n}^2, x^2(1), \dots, x^2(g), x^1(t+1), \dots, x^1(s_1) = x^2(s_2) = \delta_{2^n}^{2^n}$. La sequenza appena determinata è comunque una sequenza di lunghezza minima poiché se per assurdo non lo fosse non lo sarebbe nemmeno la sequenza dallo stato $\delta_{2^n}^1$ allo stato $\delta_{2^n}^{2^n}$ e questo sarebbe in contraddizione con le ipotesi effettuate. Allo stesso modo si può procedere per gli stati $\delta_{2^n}^i$, $i = 3, \dots, 2^n - 1$, come per lo stato $\delta_{2^n}^2$ e considerando l'insieme $C_1 \cap C_2 \cap \dots \cap C_i$. Finito questo procedimento rimane da definire solo la 2^n -esima colonna; poiché $\delta_{2^n}^{2^n}$ è uno stato stabilizzabile, esiste sicuramente un controllo $u_\alpha = \delta_{2^m}^\alpha$ tale che la dinamica della BCN rimanga sempre nello stato $\delta_{2^n}^{2^n}$. Si ha quindi:

$$\begin{aligned} K\delta_{2^n}^{2^n} = u_\alpha &\iff K\delta_{2^n}^{2^n} = \delta_{2^m}^\alpha \\ &\iff \text{Col}_{2^n}(K) = \delta_{2^m}^\alpha. \end{aligned}$$

In questo modo si ottiene una matrice $K \in \mathcal{L}^{2^n \times 2^n}$ ben definita e tale che, partendo da ogni stato $x(0) \in \Delta^{2^n}$ e applicando il controllo in retroazione in forma (3.19), la BCN si stabilizzi in x_{2^n} . \square

Esempio 3.5.7. Si consideri la seguente BCN:

$$\begin{aligned} x(t+1) &= Lx(t) \\ &= \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} x(t), \end{aligned} \quad (3.20)$$

e si supponga di voler trovare un controllo in retroazione per stabilizzare la rete nello stato x_3 . Partendo dallo stato x_1 la sequenza più corta che porta a x_3 è data da x_1, x_2 e x_3 ed è ottenuta tramite la sequenza di controllo u_1, u_1 . Dal Lemma 3.5.5 si ha quindi:

$$\text{Col}_1(K) = \delta_2^1,$$

$$\text{Col}_2(K) = \delta_2^1.$$

Dato che la sequenza da x_2 a x_3 è già stata definita si può passare direttamente allo stato x_4 . Dallo stato x_4 la sequenza più corta che porta a x_3 è data da x_4, x_2 e x_3 e dunque rimane da definire, con un procedimento analogo a prima, la quarta colonna della matrice K i.e.:

$$\text{Col}_4(K) = \delta_2^2.$$

Affinché la dinamica, raggiunto lo stato x_3 , ci rimanga indefinitamente si deve definire:

$$\text{Col}_3(K) = \delta_2^2;$$

la matrice di retroazione K risulta quindi:

$$K = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

Applicando il controllo:

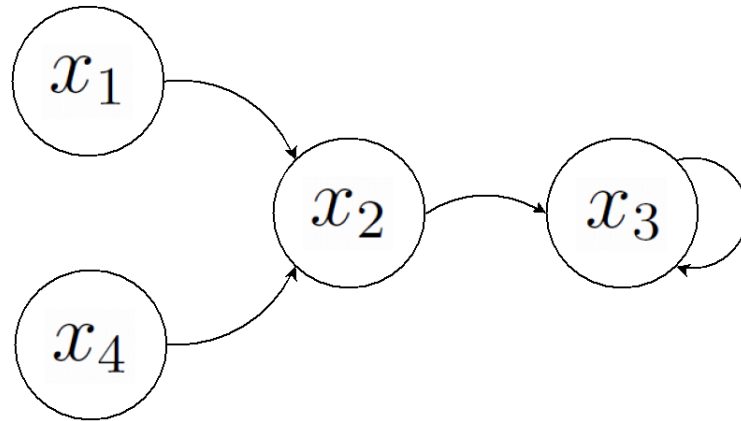


Figura 3.4: Dinamica della BCN controllata

$$u(t) = Kx(t)$$

si ottiene dunque la stabilizzazione della BCN in (3.20) nello stato x_3 e la dinamica controllata è riportata in Figura 3.4.

Si noti che la matrice di retroazione stabilizzante può non essere l'unica. Nell'esempio appena analizzato può essere ottenuta una seconda matrice:

$$K' = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

mediante la quale il controllo in retroazione stabilizza la rete sempre nello stato x_3 . \diamond

Capitolo 4

Applicazioni

Le reti Booleane, come accennato in precedenza, rivestono un ruolo importante nella modellizzazione delle reazioni biochimiche e nello studio della dinamica di complesse reti molecolari che regolano le principali funzioni biologiche. In particolare in questi ultimi anni sta crescendo il consenso per un approccio logico allo studio delle reti genetiche; l'utilizzo delle Boolean network infatti consente di prevedere le sequenze proteiche e l'attività dei geni con un numero di variabili minore rispetto al classico approccio mediante equazioni differenziali. In questo Capitolo si analizzeranno due possibili applicazioni delle reti Booleane: l'analisi del ciclo cellulare del lievito a fissione [14] ed il controllo del ciclo cellulare dei mammiferi [17]. In questi due esempi gli stati della BN corrispondono a situazioni biologiche specifiche e dunque è possibile verificare quali tra queste siano stazionarie e quali no. Inoltre, una volta ricavati gli attrattori, lo studio dei rispettivi bacini di attrazione consente di trarre informazioni sulla robustezza della dinamica cellulare rispetto agli errori e alle mutazioni.

4.1 Analisi del ciclo cellulare nel lievito a fissione

In questo paragrafo si vuole descrivere il ciclo cellulare nel lievito a fissione (nel caso in esame, il *Schizosaccharomyces Pombe*) attraverso una Boolean network basandosi su quanto riportato in [14].

Il lievito *S. Pombe* è stato sequenziato nel 1999 ed i modelli matematici finora usati per la scissione cellulare [36] [37] sono basati sull'equazioni differenziali ordinarie delle concentrazioni biochimiche che prendono parte al processo e delle loro variazioni nel tempo. Da questi modelli matematici si può ricavare una BN, come ad esempio quella ottenuta in [30] che riguarda il ciclo del lievito a gemmazione

Saccharomyces cerevisiae, che è possibile studiare grazie agli strumenti forniti nel Capitolo 2.

4.1.1 Il ciclo cellulare nel lievito a fissione

Si consideri ora il ciclo cellulare nel *S. Pombe*; l'intero processo di fissione si divide in quattro fasi differenti: G_1 , S , G_2 e M . Durante la prima fase G_1 (in inglese “gap” ovvero “intervallo”), la cellula cresce e se si verificano alcune condizioni specifiche comincia la divisione. Nella seconda fase S (sintetica) viene sintetizzato il DNA e i cromosomi vengono duplicati. Dopo una seconda fase di gap (G_2), nell'ultima fase (M) avviene la mitosi in cui i cromosomi si separano e avviene la separazione della cellula nelle due cellule figlie. Durante la fase G_1 la cellula può entrare in una fase di quiescenza (G_0) in cui smette di dividersi in modo temporaneo o irreversibile. Una rappresentazione grafica del ciclo cellulare è riportata in Figura 4.1.

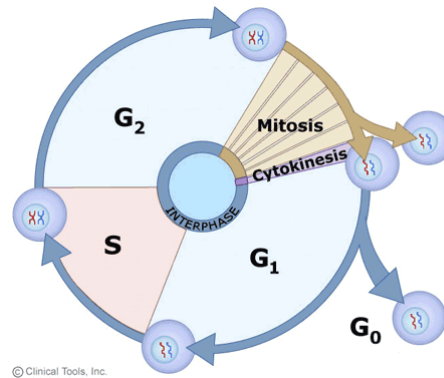


Figura 4.1: Rappresentazione del ciclo cellulare.

Le reazioni biochimiche che regolano il ciclo cellulare sono state ampiamente studiate ([36], [40]) e sono riportate in dettaglio nella Tabella 4.1.

4.1.2 Rappresentazione del ciclo cellulare tramite Boolean network

Si vogliono ora rappresentare le interazioni tra i regolatori del ciclo cellulare mediante una Boolean network.

Si considerino a questo proposito i nodi X_i , $i = 1, \dots, 10$, con $X_i \in \mathcal{B}$; il nodo X_i assume valore 1 o 0 a seconda della presenza o meno nella cellula della proteina associata, secondo quanto riportato in Tabella 4.2.

| Parent node | Daughter node | Rule of activation (comments) | Rule of inhibition (comments) |
|--------------------------|---|--|--|
| Start node | Starter Kinases (SK): Cdc2/Cig1, Cdc2/Cig2, Cdc2/Puc1 | Start node works as an indicator of mass of the cell and activates Start Kinases (SK) Cdc2/Cig1, Cdc2/Cig2, Cdc2/Puc1, +1[9] | |
| SK | Ste9, Rum1 | | Phosphorylate, thereby inactivate, -1 [9,25] |
| Cdc2/Cdc13 | Cdc25 | Cdc25 is phosphorylated thereby activated, +1 [9]. | |
| Wee1, Mik1 | Cdc2/Cdc13* | | Phosphorylate, inactivating, -1 [9] |
| Rum1 | Cdc2/Cdc13, Cdc2/Cdc13* | | Binds and inhibits activity, -1 [9]. |
| Cdc2/Cdc13 | Rum1 | | Phosphorylates and thereby targets Rum1 for degradation. -1 [9,25] |
| Ste9 | Cdc2/Cdc13, Cdc2/Cdc13* | | Labels Cdc13 for degradation [25,9], -1. |
| Cdc2/Cdc13* | Slp1 | Highly activated Cdc2/Cdc13* activates Slp1, [24,9] +1. | |
| Slp1 | Cdc2/Cdc13, Cdc2/Cdc13* | | Promotes degradation of Cdc13, thereby the activity of Cdc2/Cdc13 drops -1 [9] |
| Slp1 | PP | Activates, +1 [9] | |
| PP (Unknown phosphatase) | Ste9, Rum1, Wee1, Mik1 | Activates Rum1, Ste9, and the tyrosine-modifying enzymes (Wee1, Mik1) [9], +1 | |
| Cdc25 | Cdc2/Cdc13* | Cdc25 reverses phosphorylation of Cdc2, thereby Cdc2/Cdc13* becomes active, +1 [9,24] | |
| Cdc2/Cdc13 | Ste9 | | inhibits -1 [24] |
| PP | Cdc25 | | inhibits -1 [9] |
| Cdc2/Cdc13 | Wee1, Mik1 | | inhibits -1 [24] |
| Cdc2/Cdc13* | Rum1, Ste9 | | Inhibits -1 [24] |

doi:10.1371/journal.pone.0001672.t001

Tabella 4.1: Interazioni dei principali elementi coinvolti nella regolazione del ciclo cellulare del lievito a fissione da [14].

| nodo | proteina |
|----------|-------------|
| X_1 | Start |
| X_2 | SK |
| X_3 | Cdc2/Cdc13 |
| X_4 | Ste9 |
| X_5 | Rum1 |
| X_6 | Slp1 |
| X_7 | Cdc2/Cdc13* |
| X_8 | Wee1/Mik1 |
| X_9 | Cdc25 |
| X_{10} | PP |

Tabella 4.2: Relazione tra i nodi della BN e le proteine coinvolte nel ciclo cellulare.

Le interazioni tra i diversi nodi, in accordo con la Tabella 4.1 sono rappresentati dagli archi in Figura 4.2.

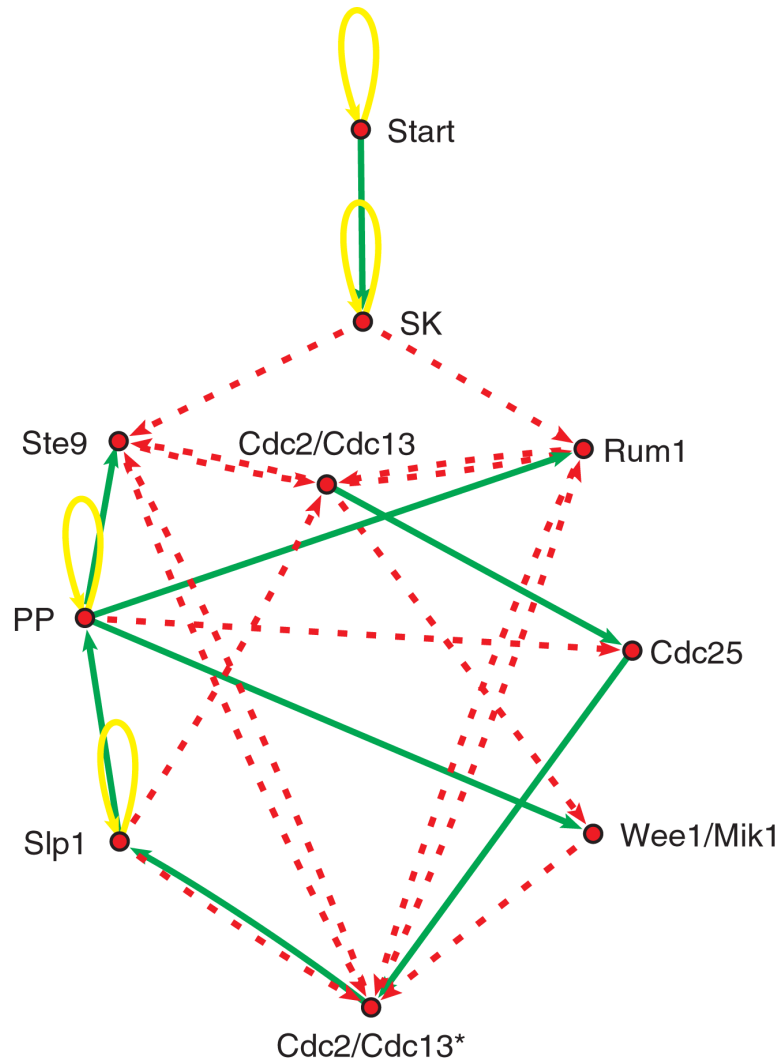


Figura 4.2: Rappresentazione delle interazioni tra le diverse proteine coinvolte nel ciclo cellulare del lievito a fissione. Le frecce verdi indicano le attivazioni, le frecce rosse le inibizioni e le frecce gialle le autoinibizioni.

L'intero modello può essere rappresentato dalla seguente Boolean network:

$$\left\{ \begin{array}{l} X_1(t+1) = f(X_1(t), \dots, X_n(t)) \\ X_2(t+1) = f(X_1(t), \dots, X_n(t)) \\ X_3(t+1) = g(X_1(t), \dots, X_n(t)) \\ X_4(t+1) = f(X_1(t), \dots, X_n(t)) \\ X_5(t+1) = f(X_1(t), \dots, X_n(t)) \\ X_6(t+1) = f(X_1(t), \dots, X_n(t)) \\ X_7(t+1) = h(X_1(t), \dots, X_n(t)) \\ X_8(t+1) = f(X_1(t), \dots, X_n(t)) \\ X_9(t+1) = f(X_1(t), \dots, X_n(t)) \\ X_{10}(t+1) = f(X_1(t), \dots, X_n(t)) \end{array} \right. ; \quad (4.1)$$

in cui le funzioni f, g e h sono date da:

$$f : X_i(t+1) = \begin{cases} 0 & \text{se } \sum_j a_{ij} N_j(t) < 0 \\ 1 & \text{se } \sum_j a_{ij} N_j(t) > 0, \\ X_i(t) & \text{se } \sum_j a_{ij} N_j(t) = 0 \end{cases}, \quad i \in \{1, 2, 4, 5, 6, 8, 9, 10\},$$

$$g : X_3(t+1) = \begin{cases} 0 & \text{se } \sum_j a_{ij} N_j(t) < 0 \\ 1 & \text{se } \sum_j a_{ij} N_j(t) \geq 0 \end{cases},$$

$$h : X_7(t+1) = \begin{cases} 0 & \text{se } \sum_j a_{ij} N_j(t) \leq 0 \\ 1 & \text{se } \sum_j a_{ij} N_j(t) > 0 \end{cases},$$

dove $a_{ij} = 1$ se esiste un arco di colore verde dal nodo N_j al nodo N_i (rappresenta l'attivazione del nodo N_i da parte dl nodo N_j), $a_{ij} = -1$ se esiste un arco di colore rosso dal nodo N_j al nodo N_i (rappresenta l'inibizione del nodo N_i da parte dl nodo N_j) e $a_{ij} = 0$ se non esistono archi da N_j a N_i (ovvero N_j non ha influenza su N_i). In accordo con [30] se il nodo N_i ha un self loop giallo si ha $a_{ii} = -1$ che rappresenta l'auto-inibizione per quelle proteine che non sono regolate negativamente dagli altri nodi. Dalla Figura 4.2 e dalle considerazioni appena riportate si può dunque ricavare la matrice:

$$A = [a_{ij}] = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix}.$$

Come si può notare due dei dieci nodi (N_3 e N_7) sono regolati da funzioni differenti rispetto agli altri; nel caso di $\text{Cdc2/Cdc13}^* \equiv N_7$ (funzione h) la somma degli elementi coinvolti deve essere positiva anche per mantenere N_7 attivo, mentre per la proteina $\text{Cdc2/Cdc13} \equiv N_3$, essendo presente un meccanismo di auto-attivazione, anche la somma nulla causa l'accensione del nodo N_3 (funzione g).

Una volta definito esaurientemente il modello, tramite una routine MATLAB `Fission_yeast.m` riportata in Appendice, si può ricavare la matrice di transizione $L \in \mathcal{L}^{1024 \times 1024}$ della BN in (4.1) in forma algebrica:

$$x(t+1) = Lx(t),$$

in cui $x(t) = N_1(t) \times \cdots \times N_{10}(t)$, con N_i versione vettoriale del nodo X_i .

4.1.3 Simulazione del ciclo cellulare del lievito a fissione

Si consideri ora la condizione iniziale seguente: assumiamo che siano inibiti tutti gli elementi coinvolti eccetto `Start`, `Ste9`, `Rum1`, e `Wee1/Mik1` [40] i.e. si consideri come stato iniziale $x(0) = x_{402} \approx (1, 0, 0, 1, 1, 0, 0, 1, 0, 0)$. La simulazione del comportamento della Boolean network è riportata sempre nel file `Fission_yeast.m` in cui si vede che la dinamica della BN si conclude nel punto fisso $x(9) = x_{924}$. L'intera traiettoria è riportata in Tabella 4.3 in cui a ciascuno stato è associata la corrispondente fase; i risultati ottenuti dalla routine definita coincidono con quelli esposti in [14].

Come risulta chiaro dalla Tabella 4.3 il ciclo cellulare coincide con quello descritto in precedenza e quindi si può ragionevolmente supporre che il modello utilizzato descriva in modo verosimile la reale dinamica.

Sempre nel file MATLAB `Fission_yeast.m` è riportata l'analisi completa della Boolean network da cui si possono ricavare: gli attrattori (in questo caso 13) divisi tra

| Istante | Stato | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | X_7 | X_8 | X_9 | X_{10} | Fase |
|---------|------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|----------|--------------|
| 0 | x_{412} | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | <i>Start</i> |
| 1 | x_{668} | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | G_1 |
| 2 | x_{1020} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | G_1/S |
| 3 | x_{892} | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | G_2 |
| 4 | x_{894} | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | G_2 |
| 5 | x_{886} | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | G_2/M |
| 6 | x_{870} | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | G_2/M |
| 7 | x_{1005} | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | M |
| 8 | x_{923} | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | M |
| 9 | x_{924} | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | G_1 |
| 10 | x_{924} | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | G_1 |

Tabella 4.3: Dinamica dello stato delle proteine nel ciclo cellulare.

punti fissi (12) e cicli limite (1 composto da tre stati), i propri bacini di attrazione e il tempo di transizione. Per questa analisi si è utilizzata la funzione `bn.m` fornita da Cheng e i suoi colleghi¹.

Tutti gli attrattori trovati, coincidenti con quelli riportati in [14], sono riportati in Tabella 4.4.

L'attrattore principale della BN risulta x_{924} , che attira il 74,4% degli stati e coincide come si è visto poco sopra con lo stato biologico G_1 . Questo fatto è molto significativo per la robustezza del sistema; infatti anche in presenza di errori o di perturbazioni, il sistema con alta probabilità si riporterà nello stato G_1 . Dall'analisi si ricava un altro risultato interessante poiché il secondo attrattore (per dimensione del bacino di attrazione) è un ciclo limite e secondo gli autori in [14] questo potrebbe essere collegato ai mutanti doppi *Wee1-Cdc25* in cui si verificano piccole oscillazioni nell'attività della proteina *Cdc2/Cdc13*. Come si è potuto facilmente notare, la semplificazione maggiore apportata in questo modello è data dal considerare solo la presenza o meno di un certo elemento e non la sua concentrazione; grazie alla BN associata si sono potuti comunque ricavare dei risultati significativi. In Figura 4.3 è riportato l'intero grafo di stato della BN in cui spicca l'insieme di stati che portano all'attrattore principale; le frecce blu evidenziano la sequenza del ciclo cellulare privo di alterazioni riportata precedentemente in Tabella 4.3. Sempre dalla Figura 4.3 si ricava anche il tempo di assorbimento, $T_t = 9$, in accordo con quanto trovato dal programma MATLAB.

¹STP Toolbox disponibile in <http://lsc.amss.ac.cn/~dcheng/>.

| Attrattore | Tipo | Dim. del bacino di att. | Stato | (N, N_2, \dots, N_{10}) |
|------------|------|-------------------------|------------|----------------------------------|
| 1 | FP | 762 | x_{924} | $(0, 0, 0, 1, 1, 0, 0, 1, 0, 0)$ |
| 2 | LC | 208 | x_{788} | $(0, 0, 1, 1, 1, 0, 1, 1, 0, 0)$ |
| | LC | | x_{1006} | $(0, 0, 0, 0, 0, 1, 0, 0, 1, 0)$ |
| | LC | | x_{1021} | $(0, 0, 0, 0, 0, 0, 0, 0, 1, 1)$ |
| 3 | FP | 18 | x_{988} | $(0, 0, 0, 0, 1, 0, 0, 1, 0, 0)$ |
| 4 | FP | 18 | x_{956} | $(0, 0, 0, 1, 0, 0, 0, 1, 0, 0)$ |
| 5 | FP | 2 | x_{960} | $(0, 0, 0, 1, 0, 0, 0, 0, 0, 0)$ |
| 6 | FP | 2 | x_{958} | $(0, 0, 0, 1, 0, 0, 0, 0, 1, 0)$ |
| 7 | FP | 2 | x_{954} | $(0, 0, 0, 1, 0, 0, 0, 1, 1, 0)$ |
| 8 | FP | 2 | x_{992} | $(0, 0, 0, 0, 1, 0, 0, 0, 0, 0)$ |
| 9 | FP | 2 | x_{990} | $(0, 0, 0, 0, 1, 0, 0, 0, 1, 0)$ |
| 10 | FP | 2 | x_{986} | $(0, 0, 0, 0, 1, 0, 0, 1, 1, 0)$ |
| 11 | FP | 2 | x_{928} | $(0, 0, 0, 1, 1, 0, 0, 0, 0, 0)$ |
| 12 | FP | 2 | x_{926} | $(0, 0, 0, 1, 1, 0, 0, 0, 1, 0)$ |
| 13 | FP | 2 | x_{922} | $(0, 0, 0, 1, 1, 0, 0, 1, 1, 0)$ |

Tabella 4.4: Insieme degli attrattori, punti fissi (PF) e cicli limite (LM) ricavabili dalla matrice L .

4.2 Il controllo del ciclo cellulare dei mammiferi

In questa sezione come secondo esempio verrà analizzato il ciclo cellulare dei mammiferi seguendo il lavoro svolto da Fauré et al. [17]. Come nel caso precedente il controllo del ciclo cellulare dei mammiferi è stato studiato tramite modelli basati sulle equazioni differenziali [38]; accanto a questi modelli però hanno cominciato ad affermarsi i modelli logici, che adottano un punto di vista più qualitativo, da cui si possono trarre nuove ed importanti informazioni.

4.2.1 Modellizzazione del ciclo cellulare dei mammiferi tramite una BCN

Il ciclo cellulare dei mammiferi è composto dalle stesse fasi (G_1 , S , G_2 , M e G_0) descritte nell'applicazione precedente che sono state studiate approfonditamente e possono essere facilmente monitorate tramite un microscopio ottico. Nel caso del ciclo cellulare dei mammiferi le proteine e le loro relazioni (attivazione/inibizione) sono riportate nel grafico in Figura 4.4 tratto da [17].

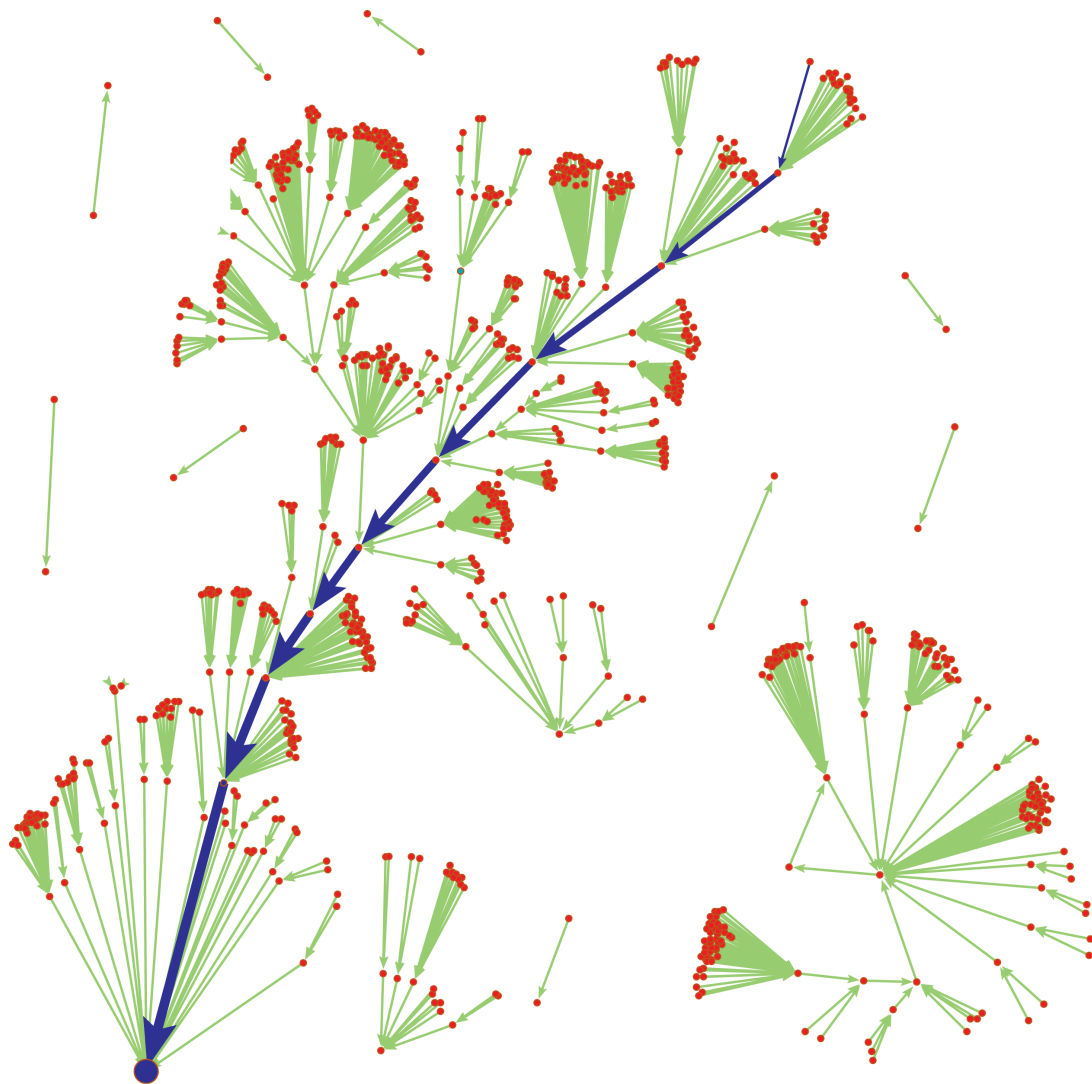


Figura 4.3: Grafo di stato completo dei 1024 possibili stati (rappresentati dai circoli rossi) riportato in [14]. Le frecce in blu evidenziano il ciclo cellulare privo di alterazioni. Sono stati esclusi per maggior leggibilità i self-loop dei punti fissi.

La divisione cellulare è strettamente regolata dalle proteine coinvolte nel processo, infatti deve essere coordinata con la crescita complessiva dell'organismo e inoltre deve rispondere a specifiche esigenze come ad esempio la guarigione delle ferite. Tale coordinamento si realizza attraverso segnali extra-cellulari, che possono essere positivi o negativi, la cui somma determina se la cellula si dividerà o rimarrà in uno stato di riposo (fase di quiescenza o G_0), che può essere raggiunto e lasciato dalla cellula durante la fase G_1 . Si ha quindi che i segnali positivi sollecitano

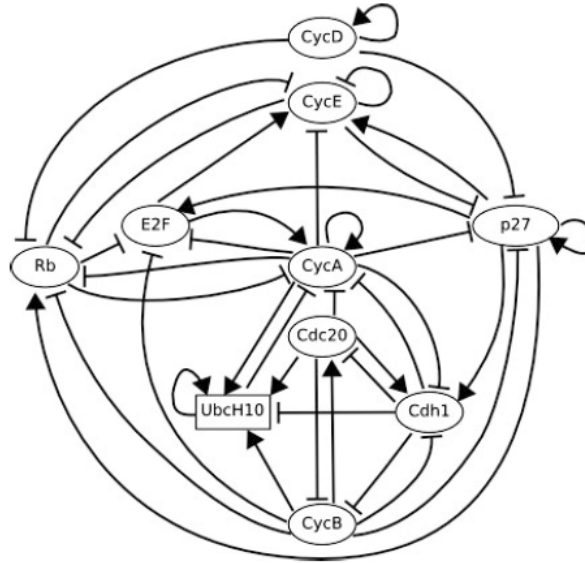


Figura 4.4: Relazioni tra i diversi componenti del ciclo cellulare dei mammiferi rappresentate dagli archi (attivazione negli archi con freccia, inibizione negli archi che terminano a “T”).

l’attivazione della Ciclina D (in inglese Cyclin D e abbreviato CycD) nella cellula che dà inizio al ciclo di divisione cellulare. In questa analisi, come nella precedente, si vuole studiare la presenza o meno degli elementi coinvolti nel ciclo cellulare e non la quantità presente; le interazioni logiche tra i diversi componenti riportate in [17] sono riassunte in Tabella 4.5.

Dalla Tabella 4.6 risulta evidente che l’elemento CycD è sempre costante; le relazioni tra gli elementi coinvolti possono essere quindi riscritti in forma di Boolean control network in cui l’elemento CycD risulta l’unico ingresso:

$$X_i(t+1) = F_i(X_1(t), X_2(t), \dots, X_9(t), U(t)) \quad i = 1, 2, \dots, 9, \quad (4.2)$$

in cui i nodi $X_i \in \mathcal{B}$ e l’ingresso $U \in \mathcal{B}$ sono definiti in Tabella 4.6.

La BCN riportata in (4.2) può essere portata in forma algebrica:

$$x(t+1) = L \times x(t) \times u(t),$$

in cui la matrice L può essere scritta:

| Elemento | Funzioni logiche che ne regolano il funzionamento |
|----------|--|
| Cycd | CycD |
| Rb | $((\neg\text{CycD}) \wedge (\neg\text{CycE}) \wedge (\neg\text{CycA}) \wedge (\neg\text{CycB}))$ $\vee (\text{p27} \wedge (\neg\text{CycD}) \wedge (\neg\text{CycB}))$ |
| E2F | $(\neg\text{Rb}) \wedge (\neg\text{CycA}) \wedge (\neg\text{CycB}) \vee (\text{p27} \wedge (\neg\text{Rb}) \wedge (\neg\text{CycB}))$ |
| CycE | $\text{E2F} \wedge (\neg\text{Rb})$ |
| CycA | $(\text{E2F} \wedge (\neg\text{Rb}) \wedge (\neg\text{Cdc20}) \wedge (\neg(\text{Cdh1} \wedge \text{Ubc})))$ $\vee (\text{CycA} \wedge (\neg\text{Rb}) \wedge (\neg\text{Cdc20}) \wedge (\neg(\text{Cdh1} \wedge \text{Ubc})))$ |
| p27 | $((\neg\text{CycD}) \wedge (\neg\text{CycE}) \wedge (\neg\text{CycA}) \wedge (\neg\text{CycB}))$ $\vee (\text{p27} \wedge (\neg(\text{CycE} \wedge \text{CycA})) \wedge (\neg\text{CycB}) \wedge (\neg\text{CycD}))$ |
| Cdc20 | CycB |
| Cdh1 | $(\neg(\text{CycA}) \wedge \neg(\text{CycB})) \vee (\text{Cdc20}) \vee (\text{p27} \wedge (\neg\text{CycB}))$ |
| UbcH10 | $(\neg\text{Cdh1}) \vee (\text{Cdh1} \wedge \text{Ubc} \wedge (\text{Cdc20}) \vee \text{CycA} \vee \text{CycB})$ |
| CycB | $(\neg\text{Cdc20}) \wedge (\neg\text{Cdh1})$ |

Tabella 4.5: Relazioni logiche tra i diversi componenti coinvolti nel ciclo cellulare.

$$L = [L_1 \mid L_2],$$

con $L_i \in \mathcal{L}^{512 \times 512}$ per $i = 1, 2$.

Dato che l'ingresso u deve essere considerato costante si hanno due tipologie di controllo disponibili: $u = u_2 = \delta_2^2$ e $u = u_1 = \delta_2^1$. Nel primo caso si ottiene la seguente BN:

$$x(t+1) = L_2 x(t); \quad (4.3)$$

mentre nel secondo caso si ottiene:

$$x(t+1) = L_1 x(t). \quad (4.4)$$

4.2.2 Ciclo cellulare in assenza e in presenza di CycD

Iniziamo ora a studiare il ciclo cellulare in assenza di CycD, i.e. supponendo $u(t) = u_2, \forall t$. In questo caso il modello si riporta alla BN senza ingresso scritta in (4.3). La matrice di transizione L_2 si può ricavare dalle relazioni riportate in Tabella 4.5 in cui si è posto $\text{CycD} = 0$ e quindi conseguentemente $\neg\text{CycD} = 1$. Il calcolo

| nodo/ingresso | elemento |
|---------------|----------|
| X_1 | Rb |
| X_2 | E2F |
| X_3 | CycE |
| X_4 | CycA |
| X_5 | p27 |
| X_6 | Cdc20 |
| X_7 | Cdh1 |
| X_8 | UbcH10 |
| X_9 | Cycb |
| U | CycD |

Tabella 4.6: Relazione tra i nodi/ingresso della BCN e gli elementi coinvolti nel ciclo cellulare.

di L_2 si ottiene mediante il file MATLAB `Mammalian_cell_cycle_cycd0.m` (riportato sempre in Appendice) in cui viene calcolato lo stato successivo applicando in ingresso tutti gli stati possibili (da x_1 a x_{512}). Una volta ottenuta la matrice di transizione utilizzando la funzione `bn.m`, già citata nell'esempio precedente, risulta che la rete presenta un unico punto fisso: $x_{236} \approx (1, 0, 0, 0, 1, 0, 1, 0, 0)$, mentre il tempo di assorbimento è $T_t = 9$. I valori dei nodi che compongono lo stato x_{236} corrispondono biologicamente alla fase G_0 o di quiescenza. Questo risultato è in accordo dunque con i risultati sperimentali dato che, come già precedentemente accennato, l'assenza della Ciclina D porta la cellula in uno stato di riposo. Dato che il punto fisso x_{236} è unico, la BN in (4.3) è globalmente stabile; partendo da qualunque stato iniziale dopo un massimo di T_t passi, la dinamica resterà indefinitamente nello stato x_{236} .

Si consideri ora la rete in presenza di CycD i.e. per $u(t) = u_1, \forall t$; in questo caso la BN da studiare risulta quella in (4.4). Ponendo `CycD=1` si ottiene, consultando la Tabella 4.6, che gli elementi Rb e p27 vengono resi nulli e rimangono spenti per tutta la durata della dinamica. La matrice di transizione L_1 si può calcolare tramite la routine in `Mammalian_cell_cycle_cycd1.m`, analoga a quella usata in precedenza. Anche in questo caso la BN ha un unico attrattore, un ciclo limite composto da 7 stati riportati in Tabella 4.7.

Anche questo ciclo limite coincide con quello ottenuto in [17]; la presenza di Ciclina D mette in moto l'intero ciclo cellulare che si ripete periodicamente.

Si è dunque visto come questo modello nonostante la sua semplicità rappresenti bene, almeno dal punto di vista qualitativo, il reale ciclo cellulare nei mammiferi. Essendo comunque un'approssimazione, alcuni comportamenti sono in contrasto

| Stato x | Configurazione dei nodi (N_1, \dots, N_9) |
|-----------|---|
| x_{284} | $(0, 1, 1, 1, 0, 0, 1, 0, 0)$ |
| x_{416} | $(0, 0, 1, 1, 0, 0, 0, 0, 0)$ |
| x_{477} | $(0, 0, 0, 1, 0, 0, 0, 1, 1)$ |
| x_{469} | $(0, 0, 0, 1, 0, 1, 0, 1, 1)$ |
| x_{498} | $(0, 1, 0, 0, 0, 0, 1, 1, 0)$ |
| x_{378} | $(0, 1, 1, 0, 0, 0, 1, 0, 0)$ |
| x_{316} | $(0, 1, 1, 1, 0, 0, 1, 0, 0)$ |

Tabella 4.7: Ciclo limite ottenuto in presenza di Ciclina D.

con il comportamento reale; gli attrattori trovati dal modello Booleano sono globali cioè attraggono la dinamica a prescindere dallo stato di partenza mentre sono stati evidenziati casi particolari che portano ad oscillazioni non previste in questa analisi. La modellizzazione tramite BN è comunque molto utile per avere un semplice e valido punto di partenza a cui aggiungere, se necessario, alcune modifiche come aggiornamento asincrono, diversi livelli di concentrazione ecc. per ottenere i risultati desiderati.

Appendice A

Codici MATLAB

In questa appendice vengono riportati tutti i codici MATLAB utilizzati per le analisi e le simulazioni svolte nel Capitolo 4.

A.1 Codici per il ciclo cellulare nel lievito a fissione

A.1.1 Fission_yeast.m

In questo file MATLAB è riportata l'analisi del ciclo cellulare del lievito a fissione. Nella prima parte viene ricavata la matrice di transizione L della BN mediante un ciclo for, in cui viene calcolato lo stato successivo applicando in ingresso tutti gli stati possibili (da x_1 a x_{512}). La matrice viene definita grazie alla funzione `lm.m` disponibile nell'STP Toolbox reperibile nel <http://lsc.amss.ac.cn/~dcheng/>; la matrice $L = \delta_n[v_1 \dots v_m]$ è definita dal comando $A = lm(v, n)$ in cui $v = v_1 \dots v_m$. Nella seconda parte viene calcolato e riportato il ciclo cellulare senza alterazioni ponendo come stato iniziale $x(0) = x_{412}$. Nella terza parte si effettua l'analisi della BN grazie alla funzione `bn.m` fornita sempre in <http://lsc.amss.ac.cn/~dcheng/>; vengono calcolati tutti gli attrattori distinti (punti fissi e cicli limite) e il tempo di assorbimento. Nella quarta ed ultima parte viene ricavata le dimensione di ciascun bacino di attrazione.

```
%Fission_yeast.m
```

```
clc  
clear all
```

```

%numero dei nodi nella BN
nod=10;
%numero degli stati nella BN
num=2^(nod);

v1=[-1 1 0 0 0 0 0 0 0 0]';
v2=[ 0 -1 0 -1 -1 0 0 0 0 0]';
v3=[0 0 0 -1 -1 0 0 -1 1 0]';
v4=[0 0 -1 0 0 0 -1 0 0 0]';
v5=[0 0 -1 0 0 0 -1 0 0 0]';
v6=[0 0 -1 0 0 -1 -1 0 0 1]';
v7=[0 0 0 -1 -1 1 0 0 0 0]';
v8=[0 0 0 0 0 0 -1 0 0 0]';
v9=[0 0 0 0 0 0 1 0 0 0]';
v10=[0 0 0 1 1 0 0 1 -1 -1]';

%definizione della matrice A
A=[v1 v2 v3 v4 v5 v6 v7 v8 v9 v10];

%per ogni stato di partenza si trova lo stato
%successivo

for i=1:num
    J=tobin(num-i,nod+1);
    X_1=J(1,1);
    X_2=J(1,2);
    X_3=J(1,3);
    X_4=J(1,4);
    X_5=J(1,5);
    X_6=J(1,6);
    X_7=J(1,7);
    X_8=J(1,8);
    X_9=J(1,9);
    X_10=J(1,10);

    Iniz= [X_1;X_2;X_3;X_4;X_5;X_6;X_7;X_8;X_9;X_10];

    vettf=A*Iniz;

%valori successivi dei nodi
X_1n(1,i)=magg(vettf(1),0,X_1);

```

```

X_2n(1,i)=magg(vettf(2),0,X_2);
X_3n(1,i)=maggtre(vettf(3));
X_4n(1,i)=magg(vettf(4),0,X_4);
X_5n(1,i)=magg(vettf(5),0,X_5);
X_6n(1,i)=magg(vettf(6),0,X_6);
X_7n(1,i)=maggset(vettf(7));
X_8n(1,i)=magg(vettf(8),0,X_8);
X_9n(1,i)=magg(vettf(9),0,X_9);
X_10n(1,i)=magg(vettf(10),0,X_10);

end

Tot=[X_1n;X_2n;X_3n;X_4n;X_5n;X_6n;X_7n;X_8n;X_9n;X_10n];

for j=1 : num
    alpha=Tot(1:nod,j);
    [Vett(j),x]=ptox(alpha');
end

%costruzione della matrice di transizione L
%in forma di oggetto lm
L=lm(Vett,num);

k=2;

%%
% Ciclo cellulare privo di alterazioni

time=0;
fprintf('Istante_\%d',time);
p=[1 0 0 1 1 0 0 1 0 0];
x=lm([ptox(p)],num)
p
for j=1:11
    time=j;
    fprintf('Istante_\%d',time);
    x=L*x
    app=x.v;
    p=xtop(app,nod)
end

%%

```

```

% Analisi della dinamica della Boolean network
[n,l,c,r0,T] = bn(L,k);

%restituisce il numero degli attrattori (n);
%la lunghezza degli attrattori (l);
%
%gli stati che formano gli attrattori (c)
%il tempo di assorbimento (r0);

fprintf('Numero degli attrattori: %d\n\n',n);
fprintf('Lunghezza degli attrattori:\n');
disp(l);
fprintf('\nTutti gli attrattori sono ora riportati:\n\n');
for i=1:length(c)
    fprintf('No. %d (lunghezza %d)\n\n',i,l(i));
    disp(c{i});
end
fprintf('Tempo di assorbimento: r0 = %d\n\n',r0);

%%
% Analisi della dimensione dei bacini di attrazione

%trasforma la matrice L da oggetto L in matrice normale
Ln=invlm(L);

Q=Ln^r0;
QQ=lm(Q);

for k=1:15
    cont(k)=0;
end

vv=QQ.v;

%attrattori
numero_att=13;
dacont(1)=922;
dacont(2)=924;
dacont(3)=926;
dacont(4)=928;
dacont(5)=954;

```

```

    dacont(6)=956;
    dacont(7)=958;
    dacont(8)=960;
    dacont(9)=986;
    dacont(10)=988;
    dacont(11)=990;
    dacont(12)=992;
    dacont(13)=788;
    dacont(14)=1006;
    dacont(15)=1021;

for j=1:15
    for i=1:num
        if(vv(i)==dacont(j))

            cont(j)=cont(j)+1;
        end
    end
end

contciclo=cont(13)+cont(14)+cont(15);

for i=1:numero_att-1
    fprintf('Dim. del bacino di att. per lo stato
%d: %d\n\n', dacont(i), cont(i));

end

fprintf('Dim. del bacino di att. per il ciclo limite:
%d\n\n', contciclo);

```

A.1.2 Altre funzioni utilizzate

All'interno del file `Fission_yeast.m` vengono utilizzate diverse funzioni che vengono ora riportate.

Le funzioni `magg.m`, `maggtre.m` e `maggset.m` definiscono l'aggiornamento dei nodi (la prima per i nodi X_i con $i = 1, 2, 4, 5, 6, 8, 9, 10$; la seconda per X_3 e la terza per X_7) e sono composte semplicemente una clausola `if-else`.

```
function M = magg(y, theta, A)
```

```

if(y > theta)
    M=1;
elseif (y < theta)
    M=0;

else
    M=A;

end

function M = maggtre(y)

if(y >= 0)
    M=1;
else
    M=0;
end

end

function M = meggset(y)

if(y > 0)
    M=1;
else
    M=0;
end

end

```

Le funzioni `ptox.m` e `xtop.m` servono per passare rispettivamente dal valore dei nodi X_i al valore dello stato x e viceversa; sono state ottenute implementando il criterio a pag. (ref) del Capitolo 1.

```

function [i,x] = ptox(p)

% PTOX  Riceve in input il vettore p =[p1 p2 ... pn]
% e restitisce due output:
% x -> è un oggetto lm che indica lo stato
% corrispondente;

```

```

% i -> numero intero con  $0 < i \leq 2^n$  tale che  $x_i = x$ 

n=length(p);
i=0;
for j=1:n
    temp=(1-p(1,j))*2^(n-j);
    i=i+temp;

end
i=i+1;
x=zeros(2^n,1);
x(i,1)=1;
x=lm(x);
return

function p = xtop(i,n)

% XTOP Riceve in input due valori:
% i -> intero tale che  $x = x_i$ ;
% n -> numero dei nodi;
% e restituisce in output:
% p -> vettore booleano che riporta i valori dei nodi;

p=zeros(1,n);
q=zeros(1,n+1);
q(1,1)=2^n-i;
for j=1:n
    p(1,j)=floor(q(1,j)/(2^(n-j)));
    q(1,j+1)=q(1,j)-p(1,j)*2^(n-j);
end
return

```

A.2 Codici per il ciclo cellulare dei mammiferi

Per la seconda applicazione si sono utilizzati due codici distinti: il primo in assenza ed il secondo in presenza di CycD. In entrambi i casi viene calcolata e poi analizzata la matrice di transizione in modo del tutto analogo a quanto fatto per il ciclo cellulare del lievito a fissione sempre utilizzando la funzione bn.m.

A.2.1 Mammalian_cell_cycle_cycd0.m

```

% Mammalian_cell_cycle_cycd0.m

clc
clear all

%numero di nodi
nod=9;

%numero di stati
num=2^(nod);

for i=1:num;
    J=tobin(num-i,nod+1);
    X_1=J(1,1);
    X_2=J(1,2);
    X_3=J(1,3);
    X_4=J(1,4);
    X_5=J(1,5);
    X_6=J(1,6);
    X_7=J(1,7);
    X_8=J(1,8);
    X_9=J(1,9);

    X_1n(i)=((~X_3)&(~X_4)&(~X_9))|(X_5&(~X_9));
    X_2n(i)= ((~X_1)&(~X_4)&(~X_9))|(X_5&(~X_1)&(~X_9));
    X_3n(i)=X_2&(~X_1);
    X_4n(i)=((X_2)&(~X_1)&(~X_6)&(~(X_7&X_8)))|((X_4)&(~X_1)
    &(~X_6)&(~(X_7&X_8)));
    X_5n(i)=((~X_3)&(~X_4)&(~X_9))|(X_5)&(~(X_3&X_4)&(~X_9));
    X_6n(i)=X_9;
    X_7n(i)=((~X_4)&(~X_9))|(X_6)|(X_5&(~X_9));
    X_8n(i)= (~X_7)|(X_7&X_8&(X_6|X_4|X_9));
    X_9n(i)=(~X_6)&(~X_7);

end

Tot= [X_1n;X_2n;X_3n;X_4n;X_5n;X_6n;X_7n;X_8n;X_9n];
for j=1 : num
    [Vett(j),x]=ptox(Tot(1:nod,j)');

```



```

end
%costruzione della matrice L

L=lm(Vett, num);
k=2;

%%
% Analisi della dinamica della Boolean network

%restituisce numero degli attrattori (n);
%lunghezza degli attrattori (l);
%gli stati che formano gli attrattori (c);
%il tempo di assorbimento (r0);

[n,l,c,r0,T] = bn(L,k);

fprintf('Numero degli attrattori: %d\n\n',n);
fprintf('Lunghezza degli attrattori:\n');
disp(l);
fprintf('\nTutti gli attrattori sono ora riportati:\n\n');
for i=1:length(c)
    fprintf('No. %d (lunghezza %d)\n\n',i,l(i));
    disp(c{i});
end
fprintf('Tempo di assorbimento: r0=%d\n\n',r0);

```

A.2.2 Mammalian_cell_cycle_cycd1.m

```

% Mammalian_cell_cycle_cycd1.m

clc
clear all

%numero di nodi
nod=9;

%numero di stati
num=2^(nod);

for i=1:num;
    J=tobin(num-i,nod+1);
    X_1=J(1,1);

```

```

X_2=J(1,2);
X_3=J(1,3);
X_4=J(1,4);
X_5=J(1,5);
X_6=J(1,6);
X_7=J(1,7);
X_8=J(1,8);
X_9=J(1,9);

X_1n(i)=0;
X_2n(i) = ((~X_1)&(~X_4)&(~X_9))|(X_5&(~X_1)&(~X_9));
X_3n(i)=X_2&(~X_1);
X_4n(i)=((X_2)&(~X_1)&(~X_6)&(~(X_7&X_8)))|((X_4)&(~X_1)
&(~X_6)&(~(X_7&X_8)));
X_5n(i)=0;
X_6n(i)=X_9;
X_7n(i)=((~X_4)&(~X_9))|(X_6)|(X_5&(~X_9));
X_8n(i) = (~X_7)|(X_7&X_8&(X_6|X_4|X_9));
X_9n(i)=(~X_6)&(~X_7);

end

Tot= [X_1n;X_2n;X_3n;X_4n;X_5n;X_6n;X_7n;X_8n;X_9n];
for j=1 : num
    [Vett(j),x]=ptox(Tot(1:nod,j)');
end

L=lm(Vett, num);
k=2;

%%
% Analisi della dinamica della Boolean network

% restituisce numero degli attrattori (n);
% lunghezza degli attrattori (l);
% gli stati che formano gli attrattori (c);
% il tempo di assorbimento (r0);

[n,l,c,r0,T] = bn(L,k);

fprintf('Numero degli attrattori: %d\n\n',n);
fprintf('Lunghezza degli attrattori:\n');
disp(l);

```

```
fprintf('\nTutti gli attrattori sono ora riportati:\n\n');
for i=1:length(c)
    fprintf('No. %d (lunghezza %d)\n\n', i, l(i));
    disp(c{i});
end
fprintf('Tempo di assorbimento: r0=%d\n\n', r0);
```


Conclusioni

In questo lavoro di tesi sono state esposte ed analizzate le proprietà principali delle Boolean networks e delle Boolean control networks. L'analisi è stata condotta seguendo l'approccio algebrico in spazio di stato; approccio pressoché nuovo dato che è stato introdotto e sviluppato in [9]. Si è quindi cercato di riportare i risultati ottenuti in [9] cercando di approfondire alcune tematiche come stabilità, stabilizzabilità e controllabilità. Per quanto riguarda le Boolean networks sono stati riportati i criteri per ottenere gli attrattori (sia punti fissi che cicli limite) con i rispettivi domini di attrazione e il tempo di assorbimento. Si sono inoltre fornite le condizioni necessarie e sufficienti affinché una rete Booleana sia globalmente stabile. Infine si è introdotta una nuova forma, detta forma normale per classi, per la matrice di transizione dalla quale è possibile ritrovare le caratteristiche principali della rete senza doverla rianalizzare.

Con l'introduzione degli ingressi, e quindi passando alle Boolean control networks sorgono problemi riguardanti la controllabilità del sistema. Si sono fornite quindi condizioni necessarie e sufficienti per la controllabilità delle BCN. Un altro problema importante riguarda la globale stabilizzazione della rete; anche in questo caso si sono fornite le condizioni che devono verificarsi sia in caso di ingresso booleano libero e sia in caso di ingresso costante. L'ultimo problema affrontato riguarda la determinazione della sequenza di ingresso affinché la dinamica del sistema vada da uno stato ad un altro raggiungibile dal primo; in questa parte si è pure dimostrato che un sistema globalmente stabilizzabile può sempre essere stabilizzato mediante un controllo in retroazione $u(t) = Kx(t)$ e si è fornito il criterio per poter ricavare la matrice K stabilizzante.

Una volta terminata l'analisi delle proprietà delle Boolean control networks sono state riportate due possibili applicazioni nell'ambito biochimico disponibili in letteratura (citare); i risultati riportati, ottenuti mediante lo studio delle reti in spazio di stato, coincidono con quelli trovati dagli autori.

Alcune tematiche riguardanti le Boolean networks rimangono parzialmente o totalmente inesplorate in letteratura; tra queste ad esempio l'analisi dell'osservabilità e il controllo ottimo. Inoltre anche il legame tra Boolean control networks e switched

systems, in quest'ambito solo accennato, potrebbe essere ulteriormente approfondito e sviluppato. Per quanto riguarda le applicazioni pratiche, ulteriori utilizzi delle Boolean networks rispetto a quelle già accennate potrebbero riguardare la teoria dei codici, argomento di grande interesse nell'ingegneria dell'Informazione.

Ringraziamenti

In questa pagina vorrei cogliere l'occasione di ringraziare tutti quelli che mi hanno supportato (e sopportato) in questo periodo di tesi.

Ringrazio in primo luogo la Prof. Valcher che è stata sempre disponibilissima nei miei confronti sia dal punto di vista accademico, sia dal punto di vista umano evitando che cadessi in preda al panico.

Ringrazio la mia famiglia per essermi stata sempre accanto e per avermi aiutato a prendere ogni decisione. In particolar modo ringrazio mia mamma e mia sorella Elisa, che hanno letto (saltando le formule) innumerevoli volte la mia tesi.

Ringrazio Irene, perché con lei ho condiviso questo periodo di stress ma nonostante questo mi è sempre rimasta accanto facendomi sentire tutta la sua vicinanza e il suo amore.

Ringrazio tutti i miei amici che in qualche modo hanno fatto sentire la loro vicinanza e che si sforzavano di apparire interessati quando cominciavo a parlare della mia tesi.

Infine, ma non per questo meno importante, ringrazio Ginevra (o meglio Ginevrina) perché ogni suo sorriso mi ricorda costantemente le cose importanti della vita.

Grazie a tutti.

Bibliografia

- [1] Akutsu T., Hayashida M., Ching W.-K. and Ng M. K., “Control of Boolean Networks: Hardness Results and Algorithms for Tree Structured Networks”, *J. Theoretical Biology* vol. 244, pp. 670-679, 2007.
- [2] Albert R. and Othmer H. G., “The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in *Drosophila melanogaster*”, *J. Theoretical Biology* vol. 223, pp. 1-18, 2003.
- [3] Aldana M., “Boolean dynamics of networks with scale-free topology”, *Physica D* vol. 185, pp. 45-66, 2003.
- [4] Bornholdt, S., “Boolean network models of cellular regulation: prospects and limitations”, *J. R. Soc. Interface* vol. 5, pp. 585-594, 2008.
- [5] Chaves M., Albert R., and Sontag E. D., “Robustness and fragility of Boolean models for genetic regulatory networks”, *J. Theoretical Biology* vol. 235, pp. 431-449, 2005.
- [6] Cheng D., “Input-state approach to Boolean networks”, *IEEE Trans. Neural Networks*, vol. 20, pp. 512-521, 2009.
- [7] Cheng D., “Disturbance decoupling of Boolean control networks”, *IEEE Trans. Automatic Control*, vol. 56, pp. 2-10, 2011.
- [8] Cheng D., Li Z. and Qi H., “Realization of Boolean control networks”, *Automatica*, vol. 46, pp. 62-69, 2010.
- [9] Cheng D., Li Z. and Qi H., *Analysis and Control of Boolean Networks*. Springer-Verlag, 2011.
- [10] Cheng D. and Qi H., “Controllability and observability of Boolean control networks”, *Automatica*, vol. 45, pp. 1659-1667, 2009.

- [11] Cheng D. and Qi H., “A linear representation of dynamic of Boolean networks”, *IEEE Trans. Automatic Control*, vol. 55, pp. 2251-2258, 2010.
- [12] Cheng D. and Qi H., “State-space analysis of Boolean networks”, *IEEE Trans. Neural Networks*, vol. 21, pp. 584-594, 2010.
- [13] Datta A., Pal R., Choudhary A. and Dougherty E. R., “Control approaches for probabilistic gene regulatory networks”, *IEEE Signal Processing Magazine*, vol. 24, pp. 54-63, 2010.
- [14] Davidich M. I. and Bornholdt S., “Boolean Network Model Predicts Cell Cycle Sequence of Fission Yeast”, *PLoS One*, 3(2) e1672, 2008.
- [15] Espinosa-Soto C., Padilla-Longoria P., and Alvarez-Buylla E. R., “A gene regulatory network model for cell-fate determination during Arabidopsis thaliana flower development that is robust and recovers experimental gene expression profiles.”, *Plant Cell*, vol. 16, pp. 2923-2939, 2004.
- [16] Farrow C., Heidel J., Maloney J. and Rogers J., “Scalar equations for synchronous Boolean networks with biological applications”, *IEEE Trans. Neural Networks*, vol. 15, pp. 348-354, 2004.
- [17] Fauré A., Naldi A., Chaouiya C. and Thieffry D., “Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle”, *Bioinformatics*, vol. 22, pp. 124-131, 2006.
- [18] Fornasini E. and Marchesini G., *Appunti di Teoria dei Sistemi*. Ed. Libreria Progetto, 2003.
- [19] Fornasini E., *Appunti di Teoria dei Sistemi*. Ed. Libreria Progetto, 2011.
- [20] Green D. G., Leishman T. G., and Sadedin S., “The emergence of social consensus in Boolean networks”, *In IEEE Symp. Artificial Life (ALIFE'07), Honolulu, HI.*, pp. 402-408, 2007.
- [21] Gupta S., Bisht S. S., Kukreti R., Jain S., and Brahmachari S. K., “Boolean network analysis of a neurotransmitter signaling pathway”, *J. Theoretical Biology*, vol. 244, pp. 423-469, 2007.
- [22] Heidel J., Maloney J., Farrow C. and Rogers J., *Fundamentals of Artificial Neural Networks*, MIT Press 1995.
- [23] Heidel J., Maloney J., Farrow C. and Rogers J., “Finding cycles in synchronous Boolean networks with applications to biochemical systems”, *Int. J. Bifurc. Chaos*, vol. 13, pp. 535-552, 2003.

- [24] Jacob F. and Monod J., “Genetic regulatory mechanisms in the synthesis of proteins”, *Journal of molecular biology*, vol. 3, pp. 318-386, 1961.
- [25] Kauffman S. A., “Metabolic Stability and Epigenesis in Randomly Constructed Genetic Nets”, *J. Theoretical Biology*, vol. 22, pp. 437-467, 1969.
- [26] Kauffman S. A., Peterson C., Samuelsson B., and Troein C., “Random Boolean network models and the yeast transcriptional network”, *Proc. Natl. Acad. Sci. U.S.A.*, vol. 100, pp. 14796-14799, 2003.
- [27] Klemm K. e Bornholdt S., “Topology of Biological Networks and Reliability of Information Processing”, *Proceedings of the National Academy of Sciences*, vol. 102, pp. 8414-18419, 2005.
- [28] Klemm K. e Bornholdt S., “Stable and unstable attractors in Boolean networks”, *Physical Review E*, vol. 72, 2005.
- [29] Laschov D. and Margaliot M., “Controllability of Boolean control networks via Perron-Frobenius Theory”, *Automatica* to appear.
- [30] Li F., Long T., Lu, Y., Ouyang Q., and Tang C., “The yeast cell-cycle network is robustly designed”, *Proc. Natl. Acad. Sci. U.S.A.*, vol. 101, pp. 4781-4786, 2004.
- [31] Li S., Assmann S. M., and Albert R., “Predicting essential components of signal transduction networks: a dynamic model of guard cell abscisic acid signaling”, *PLoS Biol.*, vol. 4, pp. 1732-1748, 2006.
- [32] Liu Q., Guo X. and Zhou T., “Optimal control for probabilistic boolean networks”, *IET Systems Biology*, vol. 4, pp. 99-107, 2010.
- [33] Luque B. and Sole R. W., “Lyapunov exponents in random Boolean networks”, *Physica A*, vol. 284, pp. 33-45, 2000.
- [34] Marr C., and Hütt M-T., “Similar Impact of Topological and Dynamic Noise on Complex Patterns”, *Physics Letters A*, vol. 349, pp. 302-305, 2006.
- [35] Morgan D. O., *The cell cycle: principles of control*, New Science Press, 2007.
- [36] Novak B., Pataki Z., Ciliberto A. and Tyson J. J., “Mathematical model of the cell division cycle of fission yeast”, *Chaos*, vol. 11, pp. 277-286, 2001.
- [37] Novak B. and Tyson J. J., “Modeling the control of DNA replication in fission yeast”, *Cell biology Proc. Natl. Acad. Sci. USA*, vol. 94, pp. 9147-9152, 1997.

- [38] Novak B. and Tyson J. J., “A model for restriction point control of the mammalian cell cycle”, *J Theor. Biol.*, vol. 230, pp. 563-579, 2004.
- [39] Szallasi Z. and Liang S., “Modeling the normal and neoplastic cell cycle with realistic Boolean genetic networks”, *Pac. Symp. Biocomput.*, vol. 3, pp. 66-76, 1998.
- [40] Tyson J. J., Csikasz-Nagy A. and Novak B., “The dynamics of cell cycle regulation”, *BioEssays*, vol. 24, pp. 1095-1109, 2002.
- [41] Zhao Q., “A remark on ‘Scalar equations for synchronous Boolean networks with biological applications’ by C. Farrow, J. Heidel, J. Maloney and J. Rogers”, *IEEE Trans. Neural Networks*, vol. 16, pp. 1715-1716, 2005.