Tableaux semantici

Metodo di dimostrazione automatica per refutazione, definito per molte logiche diverse.

Verifica di $S \models A$: ricerca esaustiva di un modello di $S \cup \{\neg A\}$.

Se la ricerca fallisce, allora $S \models A$, altrimenti si trova un modello di $S \cup \{\neg A\}$.

In generale: metodo per determinare se un insieme S di formule ha un modello o è insoddisfacible. Permette di trovare un modello di S quando S è soddisfacibile.

Un tableau è un albero etichettato da insiemi di formule (rappresentato con la radice in alto).

Tableau iniziale per S: composto da un unico nodo, etichettato da S.

Il tableau viene espanso mediante **regole di espansione**

Regole di espansione per la logica proposizionale

Per formule in **forma normale negativa**

$$\frac{A \wedge B, S}{A, B, S} \qquad \frac{A \vee B, S}{A, S \mid B, S}$$

Espandere un nodo N etichettato da S: scegliere una formula di S che non sia un letterale, e aggiungere sotto il nodo uno o due figli, a seconda della regola applicata, etichettati come determinato dalla regola stessa.

Se un nodo contiene una formula e la sua negazione (oppure contiene $\bot$), non viene più espanso: il nodo è **chiuso**, così come il ramo che lo contiene.

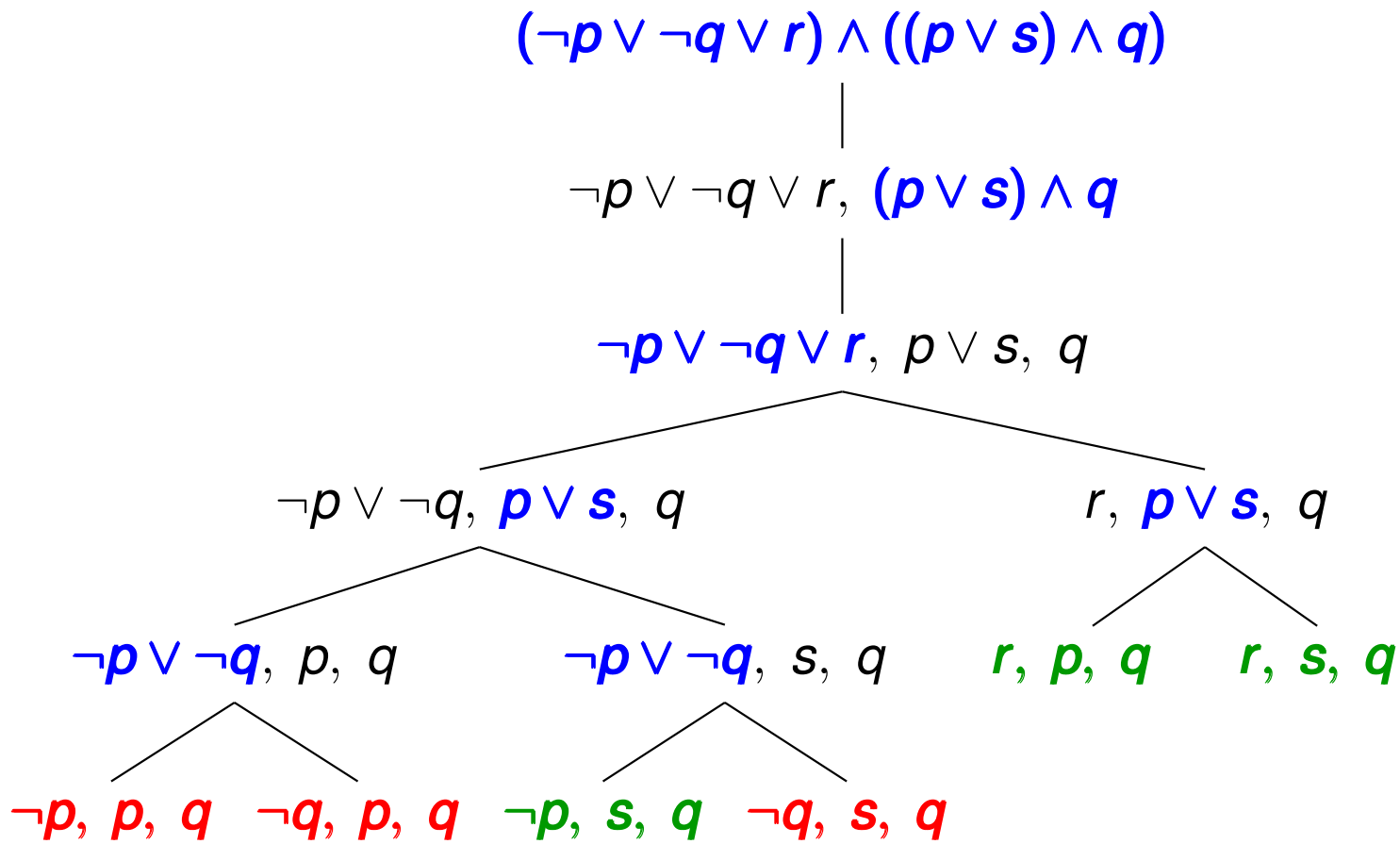
Tableau chiuso: tutti i rami sono chiusi.
Tableau aperto: con almeno un ramo non chiuso.
Tableau completo: non espandibile.

S insoddisfacibile $\iff$ esiste un tableau chiuso per S.

S soddisfacibile $\iff$ ogni foglia aperta di un tableau completo per S rappresenta (almeno) un modello di S.

Esempio 1

$S = \{(p \wedge q \rightarrow r) \wedge ((\neg p \rightarrow s) \wedge q)\} \implies$ FNN



Modelli di S:

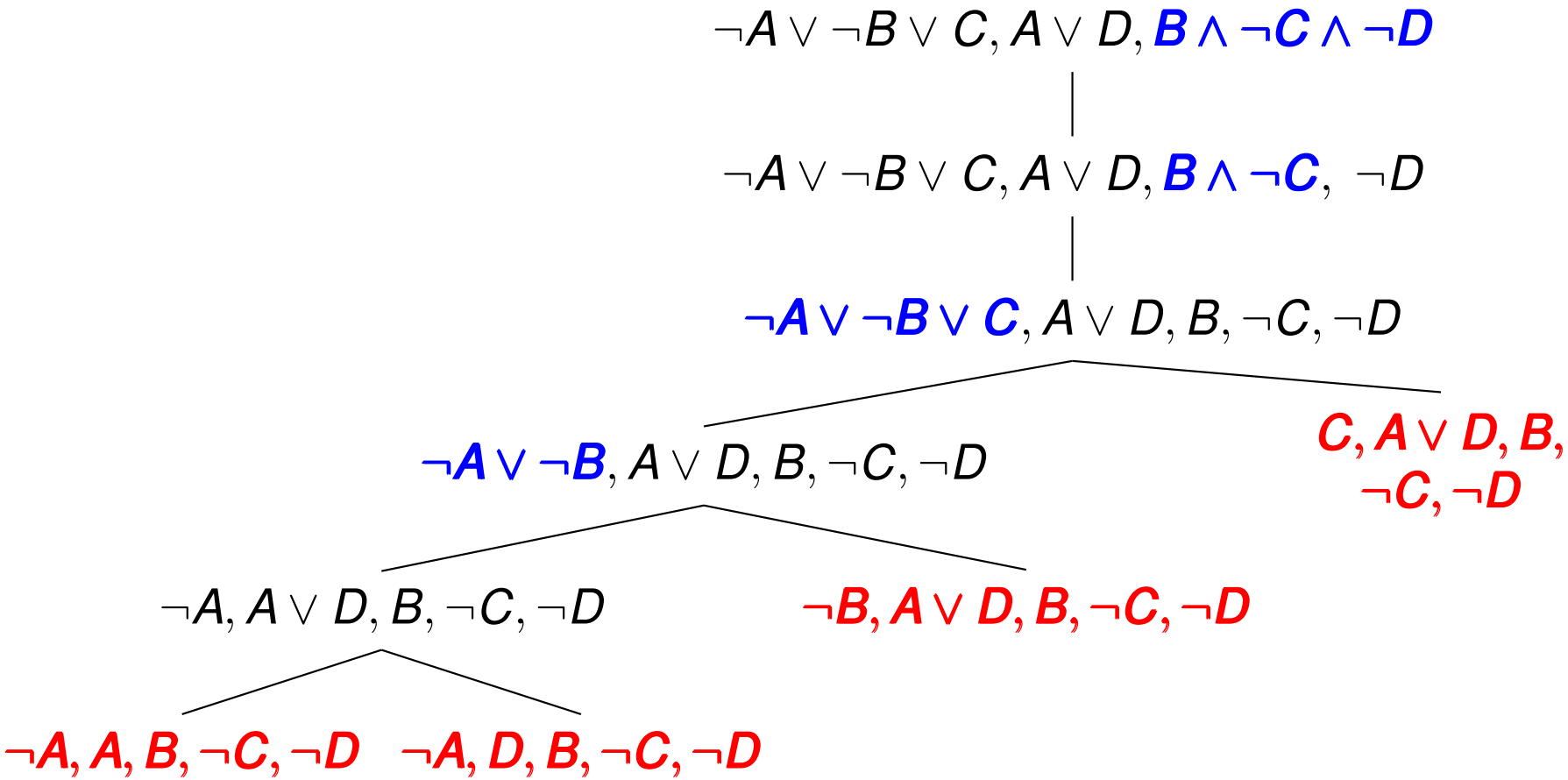
(1) ogni $\mathcal{M}$ tale che $\mathcal{M}(p) = \mathit{False}$, $\mathcal{M}(s) = \mathcal{M}(q) = \mathit{True}$

(2) ogni $\mathcal{M}$ tale che $\mathcal{M}(r) = \mathcal{M}(p) = \mathcal{M}(q) = \mathit{True}$

(3) ogni $\mathcal{M}$ tale che $\mathcal{M}(r) = \mathcal{M}(s) = \mathcal{M}(q) = \mathit{True}$

Esempio 2

Per mostrare $A \wedge B \rightarrow C,\ \neg A \rightarrow D \models B \rightarrow (C \vee D)$: costruire un tableau chiuso per $S = \{\neg A \vee \neg B \vee C,\ A \vee D,\ B \wedge \neg C \wedge \neg D\}$



Tableaux per LTL (Wolper 85)

La chiave del metodo dei tableau per LTL è la natura ricorsiva delle equivalenze:

$$\begin{aligned} \Box A &\leftrightarrow A \wedge \bigcirc \Box A \\ \Diamond A &\leftrightarrow A \vee \bigcirc \Diamond A \end{aligned}$$

Per semplicità, si trasformano inizialmente le formule in **forma normale negativa**, applicando le equivalenze:

$$\begin{aligned} \neg\neg A &\leftrightarrow A \\ A \rightarrow B &\leftrightarrow \neg A \vee B \\ \neg(A \rightarrow B) &\leftrightarrow A \wedge \neg B \\ \neg(A \wedge B) &\leftrightarrow \neg A \vee \neg B \\ \neg(A \vee B) &\leftrightarrow \neg A \wedge \neg B \\ \neg\Box A &\leftrightarrow \Diamond \neg A \\ \neg\Diamond A &\leftrightarrow \Box \neg A \\ \neg\bigcirc A &\leftrightarrow \bigcirc \neg A \end{aligned}$$

Tableaux per LTL vs tableaux per la logica classica

- Le interpretazioni sono sequenze infinite di stati: un tableau è un **grafo**
- Per verificare se un cammino nel grafo rappresenta un modello dell'insieme iniziale occorre "guardare avanti": per caratterizzare i cammini aperti è necessario **tenere traccia delle formule espanse** (marcandole per evitare di espanderle ancora)

Regole di espansione senza memoria delle formule espanse

Regole classiche

$$\frac{A \wedge B, S}{A, B, S} \qquad \frac{A \vee B, S}{A, S \mid B, S}$$

Regole temporali

$$\frac{\Box A, S}{A, \bigcirc \Box A, S} \qquad \frac{\Diamond A, S}{A, S \mid \bigcirc \Diamond A, S}$$

$$\frac{L_1, \ldots, L_m, \bigcirc A_1, \ldots, \bigcirc A_n,}{A_1, \ldots., A_n}$$

dove $L_1, \ldots, L_m$ sono letterali e $m, n \geq 0$

Le regole per $\wedge$, $\vee$, $\Box$ e $\Diamond$sono **statiche**: analizzano uno stesso stato.

La regola per $\bigcirc$ è **dinamica**: è conclusa l'analisi di uno stato e si passa ad analizzare lo stato successivo.
È applicabile solo se le altre regole non sono applicabili.

La regola Next

$$\frac{L_1, \ldots, L_m, \bigcirc A_1, \ldots, \bigcirc A_n,}{A_1, ..., A_n}$$

dove $L_1, \ldots, L_m$ sono letterali e $m, n \geq 0$

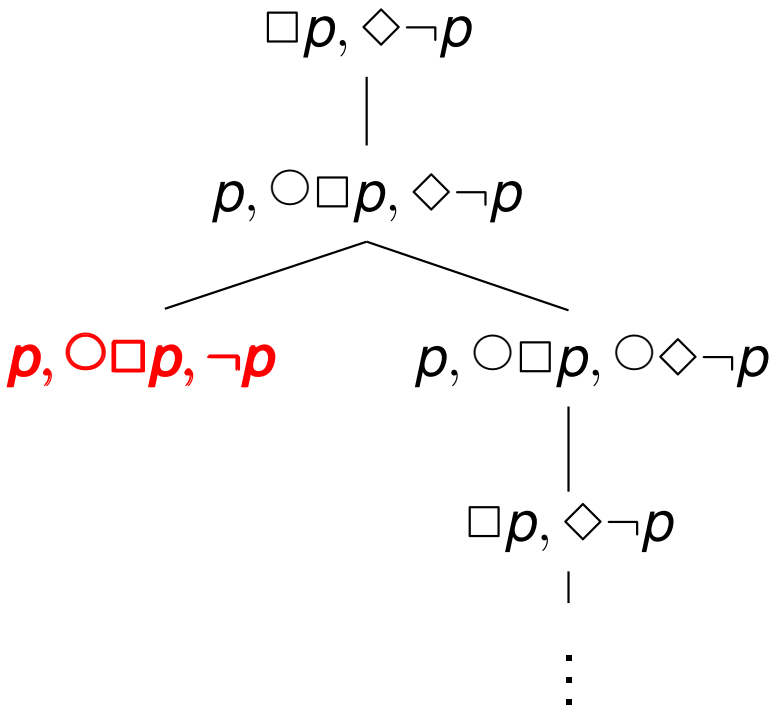
Un nodo a cui si applica questa regola rappresenta uno **stato** dell'eventuale modello temporale rappresentato dal ramo, uno stato in cui sono veri i letterali $L_1, \ldots, L_m$

Nel nodo figlio inizia l'analisi dello stato successivo.

Caso particolare in cui $n = 0$:

$$\frac{L_1, \ldots, L_m}{\top}$$

Loop checking



La terminazione della costruzione di un tableau completo è garantita dal **loop checking**: se un figlio di un nodo n avrebbe etichetta S e già esiste un nodo m con etichetta S, il nuovo nodo non viene creato, ma si aggiunge un arco da n a m (un tableau è un grafo).

Terminazione

Sia S l'insieme iniziale di formule e

$$subf(S) = \{A \mid A \text{ è una sottoformula di una formula in } S\}$$

Le etichette dei nodi in un tableau per S appartengono all'insieme finito

$$S^\star = subf(S) \cup \{\bigcirc A \mid A \in subf(S)\}$$

Cioè le etichette sono sottoinsiemi di $S^\star$: in un tableau per S ci possono essere al massimo $2^{|S^\star|}$ nodi.

Quindi la costruzione di un tableau per S termina sempre

Cammini e interpretazioni temporali

- **Stato:** nodo a cui è applicata la regola Next
- Se n è uno stato etichettato da S, n **rappresenta** tutte le interpretazioni classiche $\mathcal{I}$ tali che:

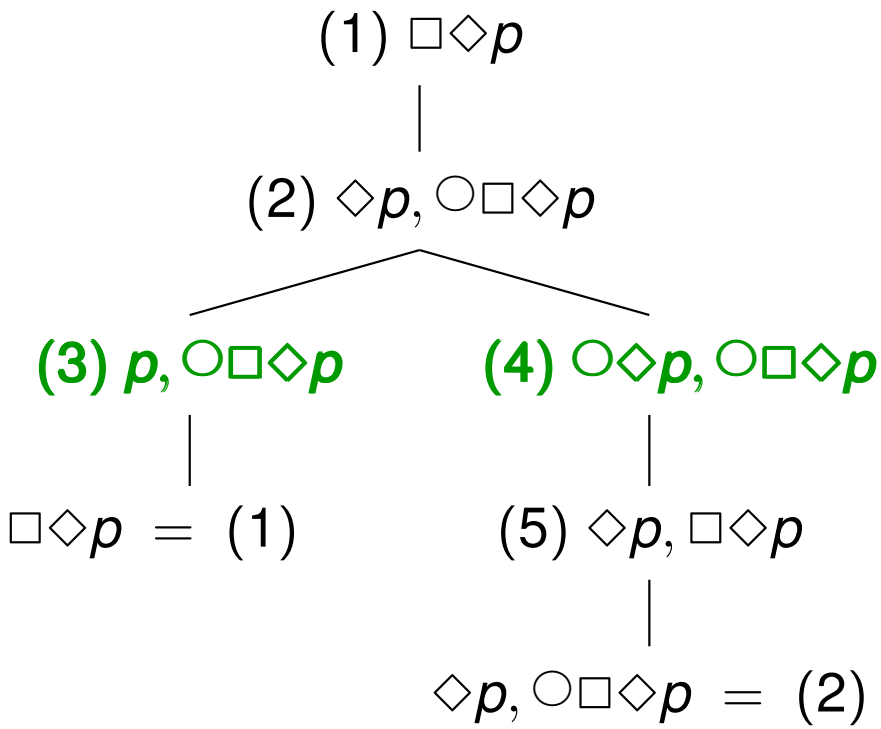
  $$\text{se } p \in S \text{ allora } \mathcal{I} \models p$$
  $$\text{se } \neg p \in S \text{ allora } \mathcal{I} \not\models p$$

  Cioè n rappresenta qualsiasi interpretazione classica in cui sia vera la congiunzione dei letterali in S.
- Se $\mathcal{C} = n_0, n_1, n_2, , ...$ un cammino infinito in un tableau, la **sequenza degli stati** del cammino $\mathcal{C}$ è la sottosequenza massimale di nodi del cammino costituita soltanto da stati (cioè la sequenza di nodi che si ottiene da $\mathcal{C}$ cancellando quelli che non sono stati).
- Un cammino $\mathcal{C}$ **rappresenta** tutte le interpretazioni temporali $\mathcal{M}$ tali che, se $s_0, s_1, s_2, ...$ è la sequenza degli stati di $\mathcal{C}$, per ogni $i \in \mathbb{N}$, s_i rappresenta $\mathcal{M}(i)$.

Uno stato può rappresentare diverse interpretazioni classiche.

Quindi un cammino può rappresentare diverse interpretazioni temporali.

Esempio



Linguaggio: $P = \{p\} \Longrightarrow$ interpretazioni classiche: $\{p\}$ e $\emptyset$.

$\mathcal{C}_1 = 1, 2, \mathbf{3}, 1, (2, \mathbf{4}, 5)^\omega$. Sequenza di stati di $\mathcal{C}_1$: $3, 4^\omega$.

Lo stato 3 rappresenta solo l'interpretazione $\{p\}$,
lo stato 4 rappresenta sia $\emptyset$ sia $\{p\}$.

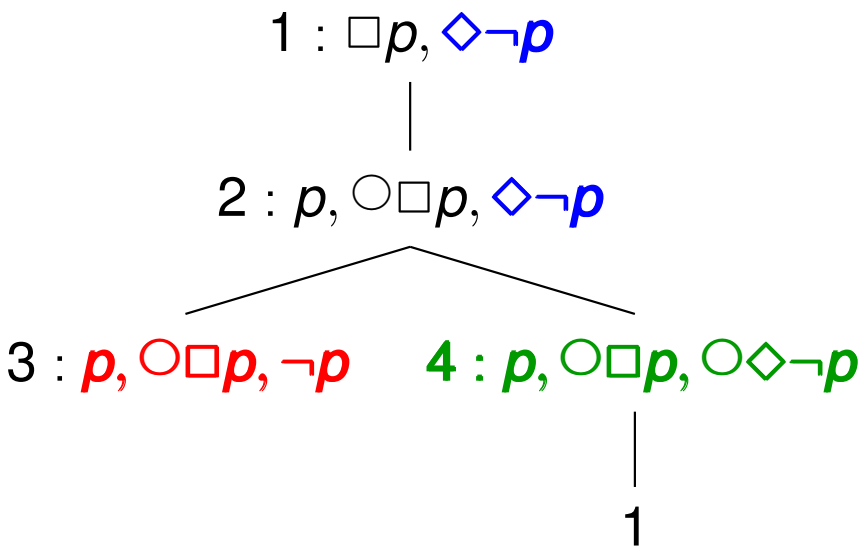
Interpretazioni $\mathcal{M}$ rappresentate da $\mathcal{C}_1$: tutte quelle tali che $p \in \mathcal{M}(0)$.

Interpretazioni rappresentate da $\mathcal{C}_2 = 1, (2, \mathbf{4}, 5)^\omega$: tutte.

Cammini aperti e chiusi

In logica classica, un ramo è chiuso se la sua foglia è contraddittoria (contiene una formula e la sua negazione) e **i rami aperti rappresentano modelli dell'insieme iniziale di formule.**

In LTL?



Il ramo $1, 2, 3$ è chiuso.

Il cammino $1, 2, \mathbf{4}, 1, 2, \mathbf{4}, \ldots$ dovrebbe rappresentare un modello di $\{\Box p, \Diamond \neg p\}$.

Quale? Quello in cui:

$\mathcal{M}_0 \models p$ (nodo 4)
$\mathcal{M}_1 \models p$ (nodo 4)
$\ldots$

Ma la **promessa** $\Diamond \neg p$ non è soddisfatta!

Regole di espansione con memoria delle formule espanse

Per caratterizzare i cammini aperti in un tableau completo, modifichiamo le regole di espansione.

Nell'applicazione delle regole *statiche*, teniamo traccia delle formule espanse, marcandole per evitare di espanderle di nuovo. Con l'applicazione della regola NEXT, le formule marcate scompaiono insieme ai letterali.

La memoria delle formule espanse serve per ricordare le "promesse" fatte in quel ramo e il loro eventuale soddisfacimento.

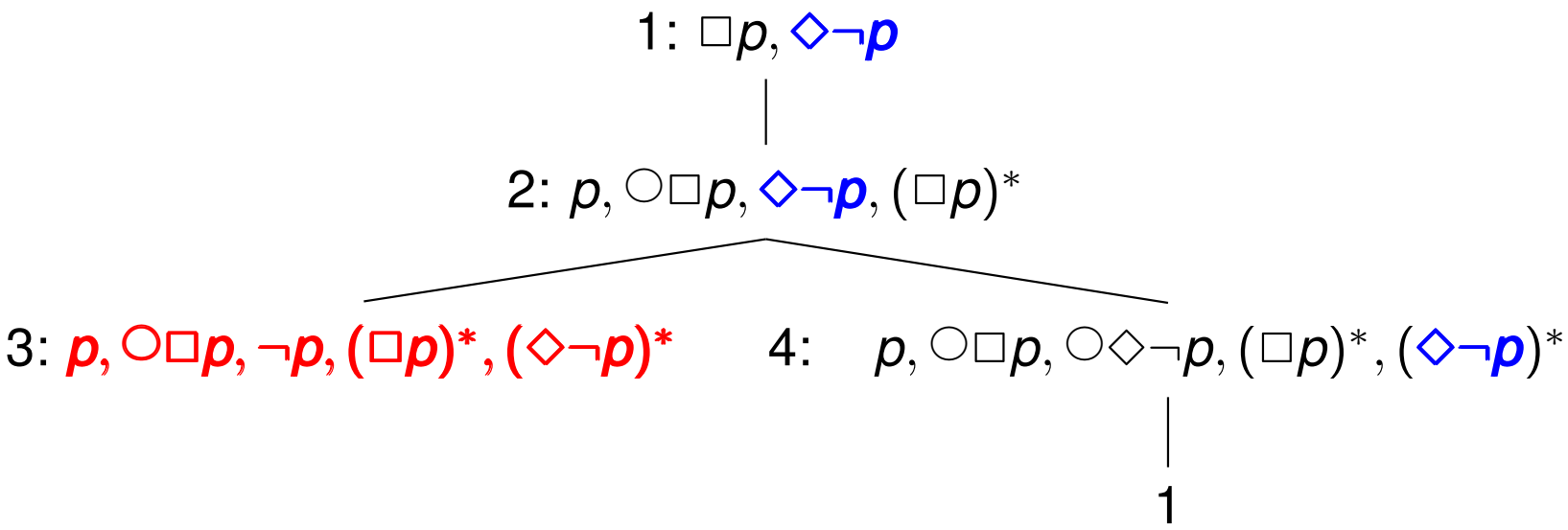
$$\frac{A \wedge B, S}{A, B, (\boldsymbol{A \wedge B})^*, S} \qquad \frac{A \vee B, S}{A, (\boldsymbol{A \vee B})^*, S \quad | \quad B, (\boldsymbol{A \vee B})^*, S}$$

$$\frac{\Box A, S}{A, \bigcirc \Box A, (\boldsymbol{\Box A})^*, S} \qquad \frac{\Diamond A, S}{A, (\boldsymbol{\Diamond A})^*, S \quad | \quad \bigcirc \Diamond A, (\boldsymbol{\Diamond A})^*, S}$$

$$\frac{L_1, \ldots, L_m, \bigcirc A_1, \ldots, \bigcirc A_n, B_1^*, \ldots, B_k^*}{A_1, \ldots., A_n}$$

dove $L_1, \ldots, L_m$ sono letterali e $m, n, k \geq 0$

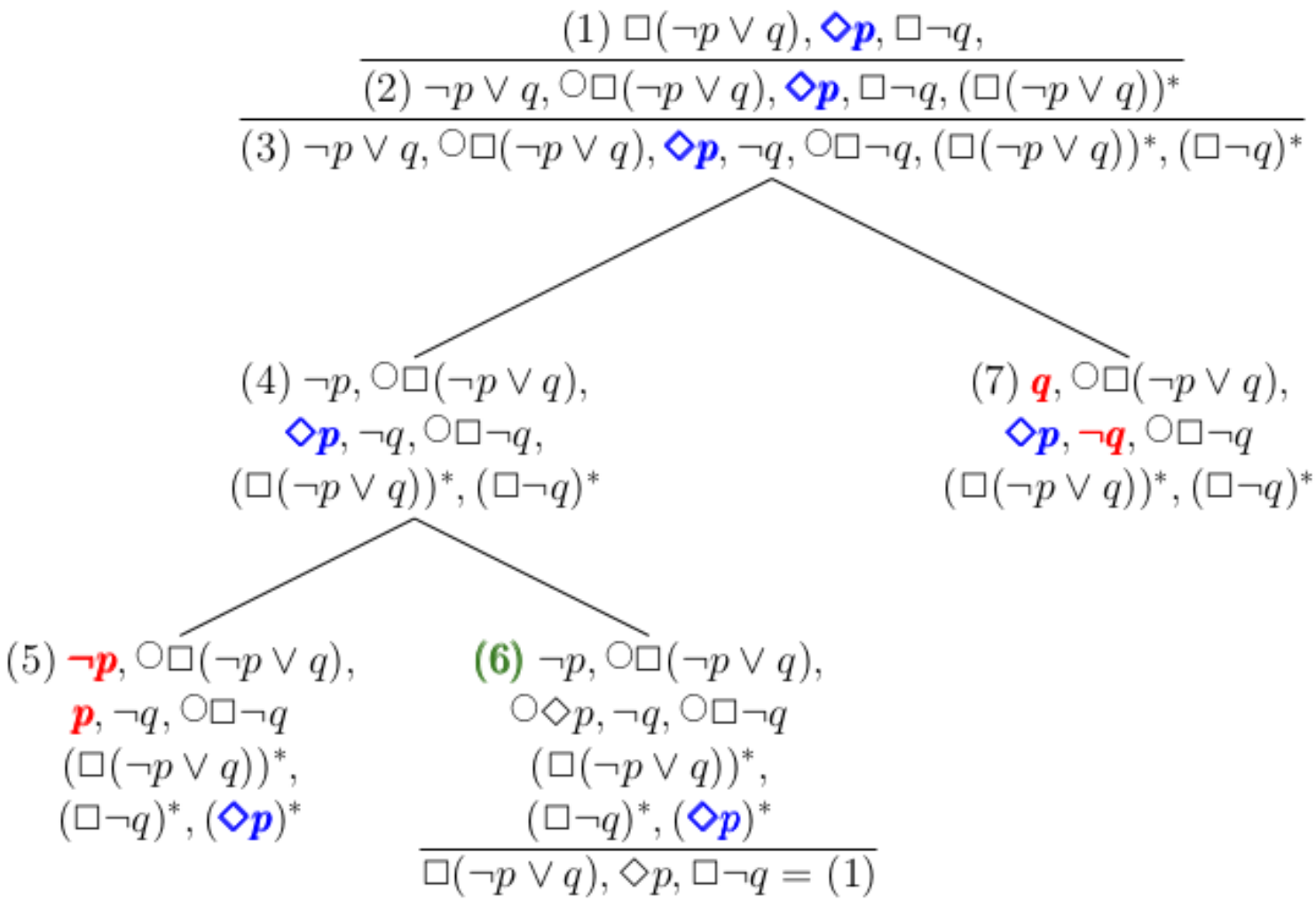
Esempio 1



Ogni nodo del cammino $(1, 2, 4)^\omega$ promette di soddifare $\Diamond \neg p$, ma nessuno contiene $\neg p$.

Quindi il cammino è **chiuso**

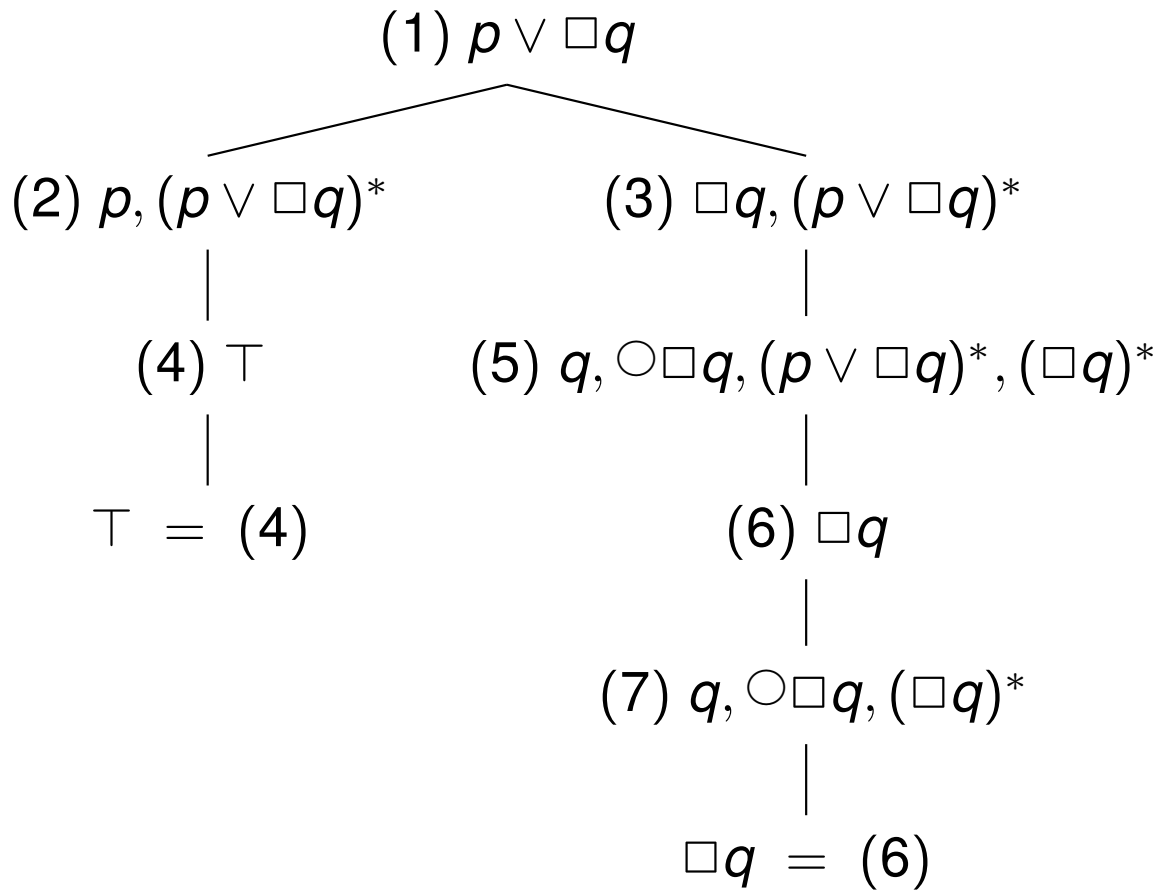
Esempio 2: un altro tableau chiuso



Cammini finiti e infiniti

Caso particolare della regola Next: $\dfrac{L_1, \ldots, L_m, B_1^*, \ldots, B_k^*}{\top}$

Quindi in un tableau completo gli unici cammini finiti sono quelli che terminano con un nodo contraddittorio.



Nota sul loop checking

Il loop checking è costoso: in una procedura automatizzata non viene eseguito su ogni nodo.

Inoltre le formule marcate non possono essere ignorate.

Negli algoritmi proposti in letteratura **il loop checking avviene soltanto su stati** (nodi a cui si può applicare la regola Next): uno stato contiene soltanto letterali (non marcati), formule Next (della forma $\bigcirc A$, non marcate) e formule marcate.

Perché due stati siano considerati uguali, devono avere gli stessi letterali, le stesse formule Next e le stesse formule marcate.

In alternativa: il loop checking può avvenire sui nodi generati dalla regola Next (che non hanno formule marcate) – come nell'esempio precedente.

Nodi di accettazione di una eventuality

Concetto chiave per caratterizzare i cammini aperti.

Eventualities: formule della forma $\Diamond A$ (o $B \mathcal{U} A$).

Un nodo n etichettato da S è un **nodo di accettazione di $\Diamond A$** sse

$$\Diamond A \notin S \text{ oppure } A \in S$$

(se S contiene $\Diamond A$, allora contiene anche A)

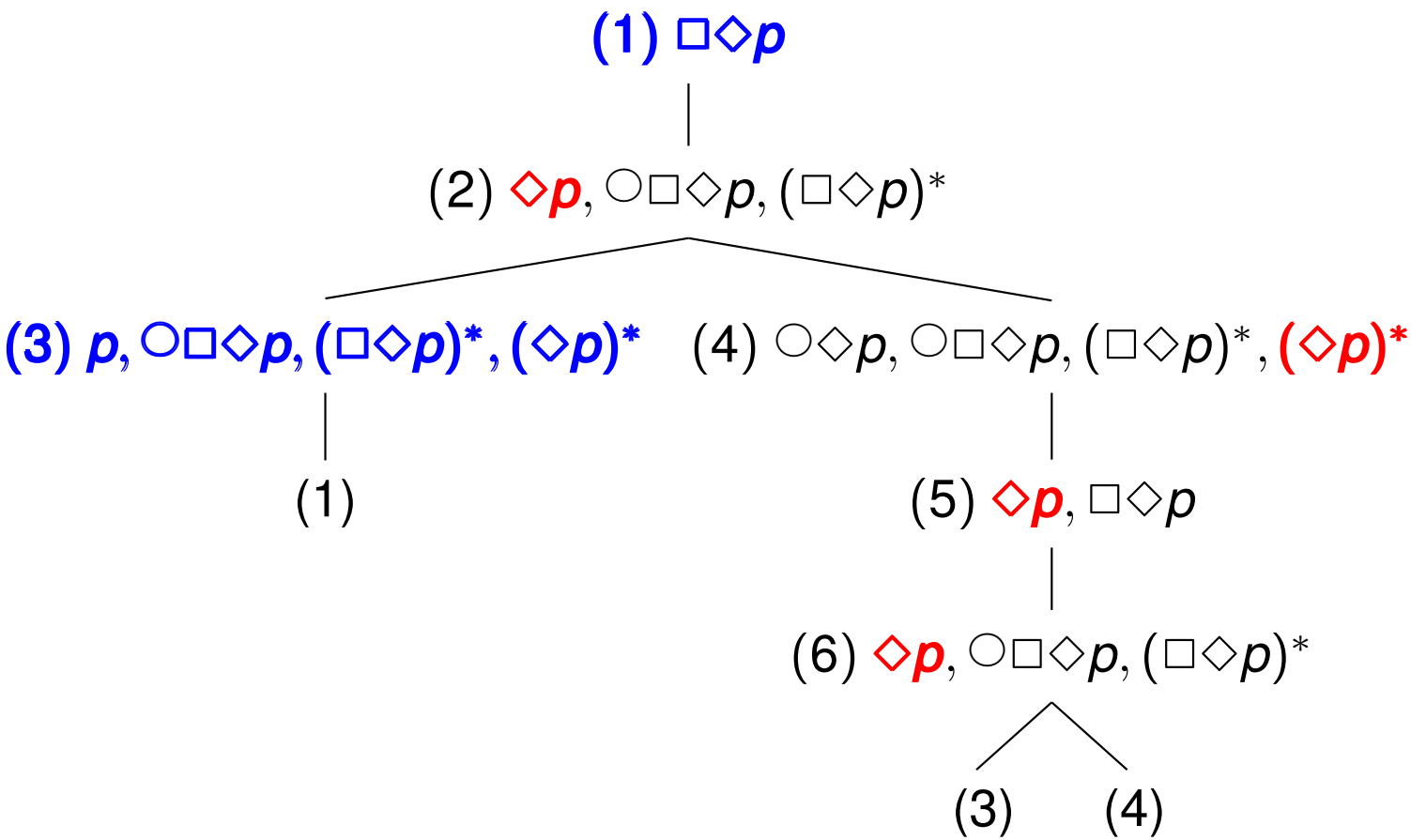
Insieme dei nodi di accettazione di $\Diamond A$:

$$f_{\Diamond A} = \{S \mid \Diamond A \notin S \text{ oppure } A \in S\}$$

Attenzione: $A \in S$ significa A è un elemento dell'insieme S (e non "A occorre come sottoformula"), e le eventuali marcature vengono ignorate.

Nota: un nodo etichettato da $\{\top\}$, o comunque da un insieme di letterali, è un nodo di accettazione di qualsiasi eventuality

Esempio



I nodi in $f_{\Diamond p}$ (i nodi di accettazione di $\Diamond p$) sono in blu
Gli altri nodi contengono l'eventuality $\Diamond p$ (in rosso) ma non contengono p
Il nodo 6, anche se ha la stessa etichetta del nodo 2, viene ancora espanso perché non è uno stato (se il controllo di cicli avviene solo su stati)

Caratterizzazione dei cammini aperti

Un cammino di un tableau soddisfa un'eventuality $\Diamond A$ sse contiene nodi di $f_{\Diamond A}$ infinitamente spesso (almeno un nodo di $f_{\Diamond A}$ occorre infinite volte nel cammino).

Se un cammino massimale non soddisfa $\Diamond A$: da un certo punto in poi contiene solo nodi che non sono in $f_{\Diamond A}$, cioè contengono $\Diamond A$ ma non A ($\Diamond A$ è una promessa che non è mai soddisfatta).

Un cammino è aperto sse soddisfa tutte le eventualities.

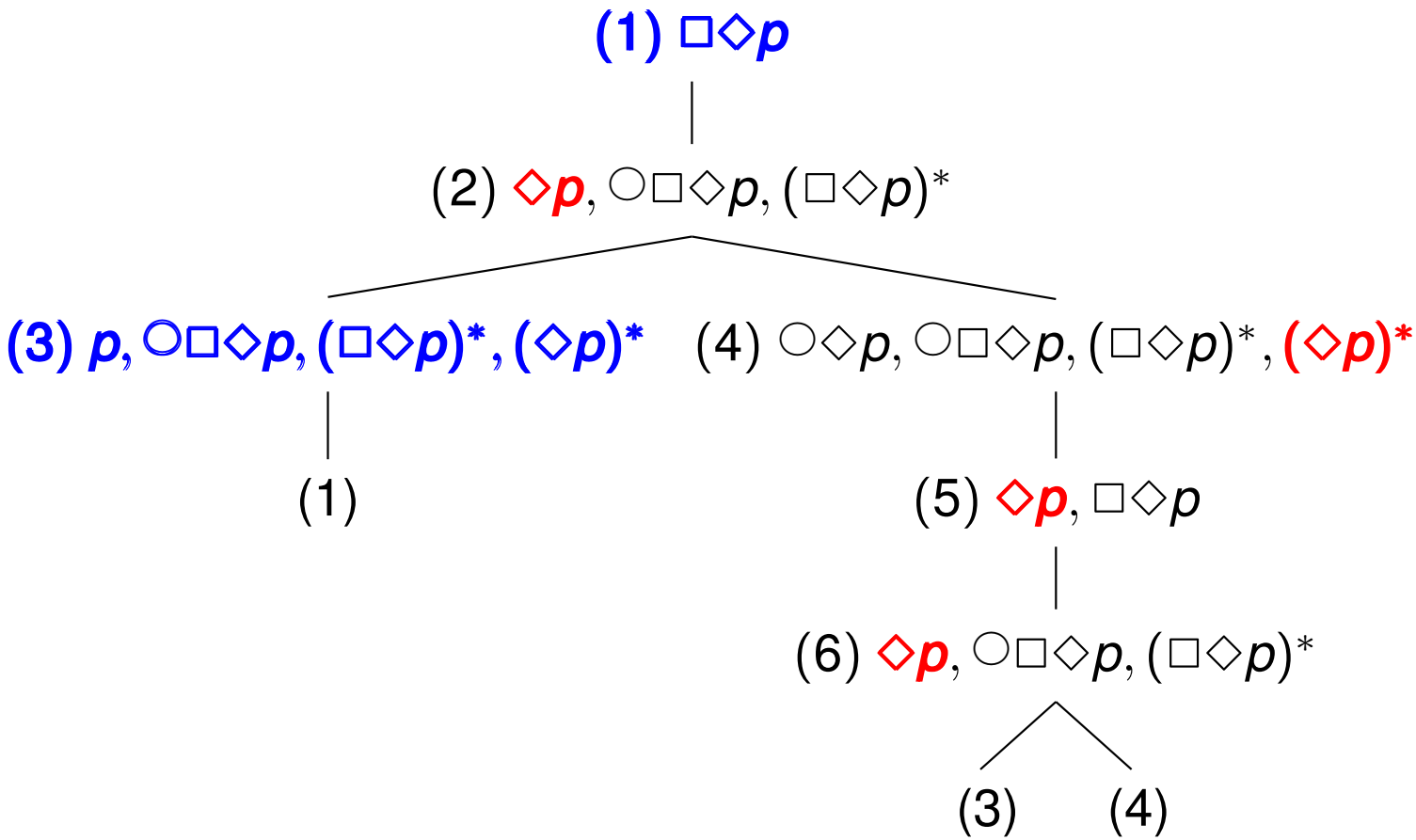
Se $\mathcal{C}$ è un cammino aperto in un tableau completo, allora ogni interpretazione rappresentata da $\mathcal{C}$ è un modello dell'insieme iniziale di formule.

Viceversa, ogni modello di un insieme di formule S è rappresentato da qualche cammino aperto in un tableau completo per S.

Un tableau è chiuso se tutti i suoi rami sono chiusi.

Teorema. Se $\mathcal{C}$ è un cammino aperto in un tableau completo, i cui stati sono $s_0, s_1, s_2, ...$, e $\mathcal{M}$ è rappresentata da $\mathcal{C}$ allora per ogni $i \in \mathbb{N}$: $\mathcal{M}_i \models A$ per ogni formula A che appartiene all'etichetta di s_i.

Esempio

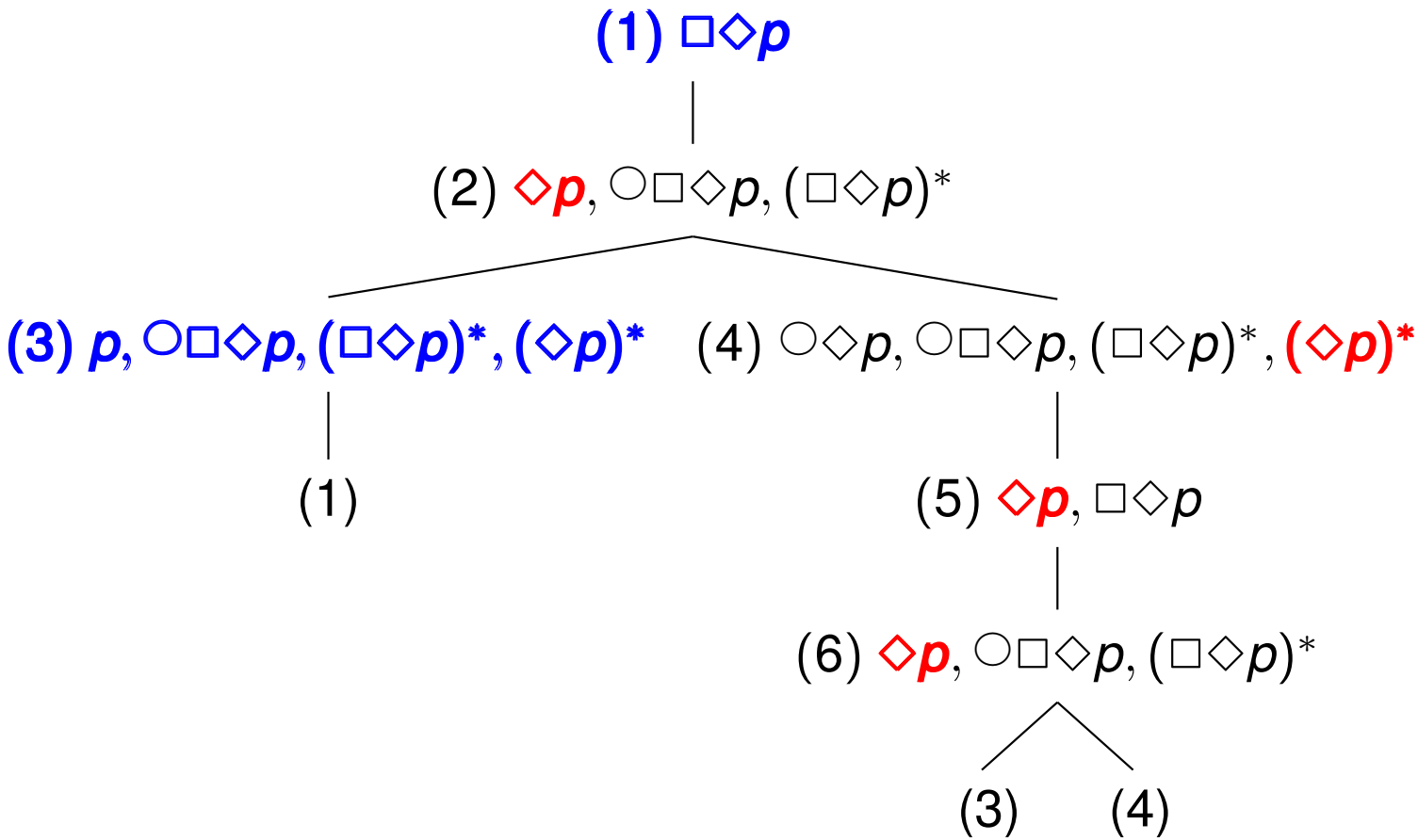


I cammini aperti sono tutti e solo quelli che contengono infinite volte almeno un nodo in $\{1, 3\}$: $(1\ 2\ (4\ 5\ 6)^*\ 3)^\omega$.

Quanti sono?

Cammini chiusi: tutti quelli della forma $1\ 2 \ldots (4\ 5\ 6)^\omega$.

Esempio



Sia $\mathcal{C} = (1\ 2\ 4\ 5\ 6\ 3)^\omega$

La sequenza di stati in $\mathcal{C}$ è $(4\ 3)^\omega$.

Interpretazioni $\mathcal{M}$ rappresentate da $\mathcal{C}$: tutte quelle in cui, per ogni i dispari, $\mathcal{M}_i \models p$ (per i pari si può avere sia $\mathcal{M}_i \models p$ che $\mathcal{M}_i \not\models p$).