Il problema della soddisfacibilità per CTL

Davide Martincigh

Relatore: Angelo Montanari

25 luglio 2016

CTL*

Formule:

$$p, \neg\alpha, \alpha \wedge \beta, X\alpha, \alpha U \beta, A\alpha, E\alpha.$$

Siano $t : \mathrm{dom}(t) \to L$ un albero privo di foglie, $\sigma = \sigma_0, ..., \sigma_n, ...$ un cammino infinito di t e ψ una formula di CTL*. Si definisce induttivamente la relazione $t, \sigma \models \psi$ ponendo

$$
\begin{aligned}
t, \sigma \models p &\iff p \in t(\sigma_0) \quad \text{con } p \in L \\
t, \sigma \models \neg\alpha &\iff t, \sigma \not\models \alpha \\
t, \sigma \models \alpha \wedge \beta &\iff t, \sigma \models \alpha \text{ e } t, \sigma \models \beta \\
t, \sigma \models X\alpha &\iff t, \sigma_{\geq 1} \models \alpha \\
t, \sigma \models \alpha U \beta &\iff \exists i \geq 0 \text{ tale che } t, \sigma_{\geq i} \models \beta \text{ e } \forall 0 \leq j < i \; t, \sigma_{\geq j} \models \alpha \\
t, \sigma \models A\alpha &\iff \forall \tau \text{ cammino infinito tale che } \tau_0 = \sigma_0 \text{ vale } t, \tau \models \alpha \\
t, \sigma \models E\alpha &\iff t, \sigma \models \neg A(\neg\alpha).
\end{aligned}
$$

CTL*

Formule:

$$p, \neg\alpha, \alpha \wedge \beta, X\alpha, \alpha U \beta, A\alpha, E\alpha.$$

Siano $t : \mathrm{dom}(t) \to L$ un albero privo di foglie, $\sigma = \sigma_0, \ldots, \sigma_n, \ldots$ un cammino infinito di t e ψ una formula di CTL*. Si definisce induttivamente la relazione $t, \sigma \models \psi$ ponendo

$$
\begin{array}{lcl}
t, \sigma \models p & \iff & p \in t(\sigma_0) \quad \text{con } p \in L \\
t, \sigma \models \neg\alpha & \iff & t, \sigma \not\models \alpha \\
t, \sigma \models \alpha \wedge \beta & \iff & t, \sigma \models \alpha \text{ e } t, \sigma \models \beta \\
t, \sigma \models X\alpha & \iff & t, \sigma_{\geq 1} \models \alpha \\
t, \sigma \models \alpha U \beta & \iff & \exists i \geq 0 \text{ tale che } t, \sigma_{\geq i} \models \beta \text{ e } \forall 0 \leq j < i \; t, \sigma_{\geq j} \models \alpha \\
t, \sigma \models A\alpha & \iff & \forall \tau \text{ cammino infinito tale che } \tau_0 = \sigma_0 \text{ vale } t, \tau \models \alpha \\
t, \sigma \models E\alpha & \iff & t, \sigma \models \neg A(\neg\alpha).
\end{array}
$$

CTL

Definizione 1

CTL è il frammento di CTL* che contiene tutte e sole le formule del tipo

$$p, \neg\alpha, \alpha \wedge \beta, EX\alpha, AX\alpha, E(\alpha U\beta), A(\alpha U\beta).$$

Definizione 2

CTL^{ndn} è il frammento di CTL privo di doppie negazioni.

É possibile definire le formule di CTL^{ndn} nel seguente modo:

$$p, \neg p, \alpha \wedge \beta, \neg(\alpha \wedge \beta), EX\alpha, \neg EX\alpha, AX\alpha, \neg AX\alpha,$$
$$E(\alpha U\beta), \neg E(\alpha U\beta), A(\alpha U\beta), \neg A(\alpha U\beta).$$

CTL

Definizione 1

CTL è il frammento di CTL* che contiene tutte e sole le formule del tipo

$$p, \neg\alpha, \alpha \wedge \beta, EX\alpha, AX\alpha, E(\alpha U \beta), A(\alpha U \beta).$$

Definizione 2

CTL^{ndn} è il frammento di CTL privo di doppie negazioni.

É possibile definire le formule di CTL^{ndn} nel seguente modo:

$$p, \neg p, \alpha \wedge \beta, \neg(\alpha \wedge \beta), EX\alpha, \neg EX\alpha, AX\alpha, \neg AX\alpha,$$
$$E(\alpha U \beta), \neg E(\alpha U \beta), A(\alpha U \beta), \neg A(\alpha U \beta).$$

CTL

Definizione 1

CTL è il frammento di CTL* che contiene tutte e sole le formule del tipo

$$p, \neg\alpha, \alpha \wedge \beta, EX\alpha, AX\alpha, E(\alpha U \beta), A(\alpha U \beta).$$

Definizione 2

CTL^{ndn} è il frammento di CTL privo di doppie negazioni.

É possibile definire le formule di CTL^{ndn} nel seguente modo:

$$p, \neg p, \alpha \wedge \beta, \neg(\alpha \wedge \beta), EX\alpha, \neg EX\alpha, AX\alpha, \neg AX\alpha,$$
$$E(\alpha U \beta), \neg E(\alpha U \beta), A(\alpha U \beta), \neg A(\alpha U \beta).$$

Chiusura

Problema 1

Sia φ una formula di CTL^{ndn}, si vuole determinare se essa è soddisfacibile. In caso di risposta affermativa, si vuole inoltre produrre un modello finito che la soddisfi.

Definizione 3

Data una formula φ si definisce chiusura di φ l'insieme

$$C(\varphi) = \bigcup \left\{ \{\psi, \widetilde{\neg\psi}\} : \psi \text{ è una sottoformula di } \varphi \right\}$$

dove

$$\widetilde{\neg\psi} = \begin{cases} \psi' & \text{se } \psi = \neg\psi' \text{ per qualche } \psi' \\ \neg\psi & \text{altrimenti.} \end{cases}$$

Chiusura

Problema 1

Sia φ una formula di CTL^{ndn}, si vuole determinare se essa è soddisfacibile. In caso di risposta affermativa, si vuole inoltre produrre un modello finito che la soddisfi.

Definizione 3

Data una formula φ si definisce chiusura di φ l'insieme

$$C(\varphi) = \bigcup \left\{ \{\psi, \widetilde{\neg\psi}\} : \psi \text{ è una sottoformula di } \varphi \right\}$$

dove

$$\widetilde{\neg\psi} = \begin{cases} \psi' & \text{se } \psi = \neg\psi' \text{ per qualche } \psi' \\ \neg\psi & \text{altrimenti.} \end{cases}$$

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B$.

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B.$

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B$.

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B$.

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B$.

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B$.

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B$.

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B.$

T_0

Il *tableau* $T_0 = (S_0, R_0)$ è un grafo i cui stati sono gli elementi $B \in 2^{C(\varphi)}$ che soddisfano le seguenti condizioni:

- Per ogni $\alpha \in C(\varphi)$ esattamente uno tra α e $\widetilde{\neg\alpha}$ appartiene a B
- $\alpha \wedge \beta \in B \iff \alpha \in B$ e $\beta \in B$
- $A(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow A(\alpha U \beta) \in B$
- $A(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $E(\alpha U \beta) \in C(\varphi)$ e $\beta \in B \Longrightarrow E(\alpha U \beta) \in B$
- $E(\alpha U \beta) \in B$ e $\beta \notin B \Longrightarrow \alpha \in B$
- $\neg A(\alpha U \beta) \in B \Longrightarrow \beta \notin B$
- $\neg E(\alpha U \beta) \in B \Longrightarrow \beta \notin B$.

Due stati B e D di S_0 sono in relazione se e solo se per ogni formula α e β sono soddisfatte le seguenti condizioni:

- $AX\alpha \in B \Longrightarrow \alpha \in D$
- $\neg EX\alpha \in B \Longrightarrow \alpha \notin D$
- $A(\alpha U\beta) \in B$ e $\beta \notin B \Longrightarrow A(\alpha U\beta) \in D$
- $\neg E(\alpha U\beta) \in B$ e $\alpha \in B \Longrightarrow \neg E(\alpha U\beta) \in D$.

Due stati B e D di S_0 sono in relazione se e solo se per ogni formula α e β sono soddisfatte le seguenti condizioni:

- $AX\alpha \in B \Longrightarrow \alpha \in D$
- $\neg EX\alpha \in B \Longrightarrow \alpha \notin D$
- $A(\alpha U\beta) \in B$ e $\beta \notin B \Longrightarrow A(\alpha U\beta) \in D$
- $\neg E(\alpha U\beta) \in B$ e $\alpha \in B \Longrightarrow \neg E(\alpha U\beta) \in D$.

Due stati B e D di S_0 sono in relazione se e solo se per ogni formula α e β sono soddisfatte le seguenti condizioni:

- $AX\alpha \in B \Longrightarrow \alpha \in D$
- $\neg EX\alpha \in B \Longrightarrow \alpha \notin D$
- $A(\alpha U\beta) \in B$ e $\beta \notin B \Longrightarrow A(\alpha U\beta) \in D$
- $\neg E(\alpha U\beta) \in B$ e $\alpha \in B \Longrightarrow \neg E(\alpha U\beta) \in D$.

Due stati B e D di S_0 sono in relazione se e solo se per ogni formula α e β sono soddisfatte le seguenti condizioni:

- $AX\alpha \in B \Longrightarrow \alpha \in D$
- $\neg EX\alpha \in B \Longrightarrow \alpha \notin D$
- $A(\alpha U\beta) \in B$ e $\beta \notin B \Longrightarrow A(\alpha U\beta) \in D$
- $\neg E(\alpha U\beta) \in B$ e $\alpha \in B \Longrightarrow \neg E(\alpha U\beta) \in D$.

Due stati B e D di S_0 sono in relazione se e solo se per ogni formula α e β sono soddisfatte le seguenti condizioni:

- $AX\alpha \in B \Longrightarrow \alpha \in D$
- $\neg EX\alpha \in B \Longrightarrow \alpha \notin D$
- $A(\alpha U\beta) \in B$ e $\beta \notin B \Longrightarrow A(\alpha U\beta) \in D$
- $\neg E(\alpha U\beta) \in B$ e $\alpha \in B \Longrightarrow \neg E(\alpha U\beta) \in D$.

Culling

Si costruisce una successione di *tableau* $T_0, T_1, ..., T_n =: T$, dove T_{i+1} si ottiene da T_i tramite l'operazione di *culling*.

Si rimuove da T_i uno stato che non rispetta almeno una delle seguenti regole:

- Regola EXT: B deve avere almeno un R_i-successore. Equivalentemente, B deve soddisfare la formula EXT.
- Regola EX: Se $EX\alpha \in B$ allora esiste un R_i-successore di B che contiene α.
- Regola NAX: Se $\neg AX\alpha \in B$ allora esiste un R_i-successore di B che non contiene α.

Culling

Si costruisce una successione di *tableau* $T_0, T_1, ..., T_n =: T$, dove T_{i+1} si ottiene da T_i tramite l'operazione di *culling*.

Si rimuove da T_i uno stato che non rispetta almeno una delle seguenti regole:

- Regola EXT. B deve avere almeno un R_i-successore. Equivalentemente, B deve soddisfare la formula EXT.
- Regola EX. Se $EX\alpha \in B$ allora esiste un R_i-successore di B che contiene α.
- Regola NAX. Se $\neg AX\alpha \in B$ allora esiste un R_i-successore di B che non contiene α.

Culling

Si costruisce una successione di *tableau* $T_0, T_1, \ldots, T_n =: T$, dove T_{i+1} si ottiene da T_i tramite l'operazione di *culling*.

Si rimuove da T_i uno stato che non rispetta almeno una delle seguenti regole:

- Regola EXT. B deve avere almeno un R_i-successore. Equivalentemente, B deve soddisfare la formula EXT.
- Regola EX. Se $EX\alpha \in B$ allora esiste un R_i-successore di B che contiene α.
- Regola NAX. Se $\neg AX\alpha \in B$ allora esiste un R_i-successore di B che non contiene α.

Culling

Si costruisce una successione di *tableau* $T_0, T_1, ..., T_n =: T$, dove T_{i+1} si ottiene da T_i tramite l'operazione di *culling*.

Si rimuove da T_i uno stato che non rispetta almeno una delle seguenti regole:

- Regola EXT. B deve avere almeno un R_i-successore. Equivalentemente, B deve soddisfare la formula EXT.
- Regola EX. Se $EX\alpha \in B$ allora esiste un R_i-successore di B che contiene α.
- Regola NAX. Se $\neg AX\alpha \in B$ allora esiste un R_i-successore di B che non contiene α.

Culling

Si costruisce una successione di *tableau* $T_0, T_1, \ldots, T_n =: T$, dove T_{i+1} si ottiene da T_i tramite l'operazione di *culling*.

Si rimuove da T_i uno stato che non rispetta almeno una delle seguenti regole:

- Regola EXT. B deve avere almeno un R_i-successore. Equivalentemente, B deve soddisfare la formula EXT.
- Regola EX. Se $EX\alpha \in B$ allora esiste un R_i-successore di B che contiene α.
- Regola NAX. Se $\neg AX\alpha \in B$ allora esiste un R_i-successore di B che non contiene α.

- Regola EU. Se $E(\alpha U \beta) \in B$ allora esiste un cammino in T_i che parte da B il cui ultimo stato contiene β e tale che tutti i suoi stati, ad eccezione eventualmente dell'ultimo, contengono α.
- Regola NAU. Se $\neg A(\alpha U \beta) \in B$ allora esiste un cammino P che parte da B tale che ogni suo stato non contiene β e tale che vale almeno una delle seguenti:
  - l'ultimo stato non contiene α;
  - P contiene un ciclo.

- Regola EU. Se $E(\alpha U \beta) \in B$ allora esiste un cammino in T_i che parte da B il cui ultimo stato contiene β e tale che tutti i suoi stati, ad eccezione eventualmente dell'ultimo, contengono α.
- Regola NAU. Se $\neg A(\alpha U \beta) \in B$ allora esiste un cammino P che parte da B tale che ogni suo stato non contiene β e tale che vale almeno una delle seguenti:
  - l'ultimo stato non contiene α;
  - P contiene un ciclo.

- Regola AU. Se $A(\alpha U \beta) \in B$ allora esiste una sottostruttura $T' = (S', R')$ di T_i con le seguenti caratteristiche:
  - T' è un DAG;
  - B è uno stato di S' privo di R'-predecessori;
  - ogni stato terminale, ovvero privo di R'-successori, contiene β;
  - ogni stato non terminale:
    - contiene α,
    - se contiene $EX\gamma$ allora ammette un R'-successore che contiene γ,
    - se contiene $\neg AX\gamma$ allora ammette un R'-successore che non contiene γ,
    - se contiene $E(\gamma U \delta)$ ma non contiene δ allora ammette un R'-successore che contiene $E(\gamma U \delta)$,
    - se contiene $\neg A(\gamma U \delta)$ e γ allora ammette un R'-successore che contiene $\neg A(\gamma U \delta)$.

- Regola AU. Se $A(\alpha U \beta) \in B$ allora esiste una sottostruttura $T' = (S', R')$ di T_i con le seguenti caratteristiche:
  - T' è un DAG;
  - B è uno stato di S' privo di R'-predecessori;
  - ogni stato terminale, ovvero privo di R'-successori, contiene β;
  - ogni stato non terminale:
    - contiene α,
    - se contiene $EX\gamma$ allora ammette un R'-successore che contiene γ,
    - se contiene $\neg AX\gamma$ allora ammette un R'-successore che non contiene γ,
    - se contiene $E(\gamma U \delta)$ ma non contiene δ allora ammette un R'-successore che contiene $E(\gamma U \delta)$,
    - se contiene $\neg A(\gamma U \delta)$ e γ allora ammette un R'-successore che contiene $\neg A(\gamma U \delta)$.

- Regola AU. Se $A(\alpha U \beta) \in B$ allora esiste una sottostruttura $T' = (S', R')$ di T_i con le seguenti caratteristiche:
  - T' è un DAG;
  - B è uno stato di S' privo di R'-predecessori;
  - ogni stato terminale, ovvero privo di R'-successori, contiene β;
  - ogni stato non terminale:
    - contiene α,
    - se contiene $EX\gamma$ allora ammette un R'-successore che contiene γ,
    - se contiene $\neg AX\gamma$ allora ammette un R'-successore che non contiene γ,
    - se contiene $E(\gamma U \delta)$ ma non contiene δ allora ammette un R'-successore che contiene $E(\gamma U \delta)$,
    - se contiene $\neg A(\gamma U \delta)$ e γ allora ammette un R'-successore che contiene $\neg A(\gamma U \delta)$.

- Regola AU. Se $A(\alpha U \beta) \in B$ allora esiste una sottostruttura $T' = (S', R')$ di T_i con le seguenti caratteristiche:
  - T' è un DAG;
  - B è uno stato di S' privo di R'-predecessori;
  - ogni stato terminale, ovvero privo di R'-successori, contiene β;
  - ogni stato non terminale:
    - contiene α,
    - se contiene $EX\gamma$ allora ammette un R'-successore che contiene γ,
    - se contiene $\neg AX\gamma$ allora ammette un R'-successore che non contiene γ,
    - se contiene $E(\gamma U \delta)$ ma non contiene δ allora ammette un R'-successore che contiene $E(\gamma U \delta)$,
    - se contiene $\neg A(\gamma U \delta)$ e γ allora ammette un R'-successore che contiene $\neg A(\gamma U \delta)$.

- Regola AU. Se $A(\alpha U \beta) \in B$ allora esiste una sottostruttura $T' = (S', R')$ di T_i con le seguenti caratteristiche:
  - T' è un DAG;
  - B è uno stato di S' privo di R'-predecessori;
  - ogni stato terminale, ovvero privo di R'-successori, contiene β;
  - ogni stato non terminale:
    - contiene α,
    - se contiene $EX\gamma$ allora ammette un R'-successore che contiene γ,
    - se contiene $\neg AX\gamma$ allora ammette un R'-successore che non contiene γ,
    - se contiene $E(\gamma U \delta)$ ma non contiene δ allora ammette un R'-successore che contiene $E(\gamma U \delta)$,
    - se contiene $\neg A(\gamma U \delta)$ e γ allora ammette un R'-successore che contiene $\neg A(\gamma U \delta)$.

Promising formula

Definizione 4

Una formula ψ è detta una *promising formula* se è della forma

$$EX\alpha, \neg AX\alpha, A(\alpha U\beta), E(\alpha U\beta), \neg A(\alpha U\beta).$$

Una *promising formula* del tipo $A(\alpha U\beta)$ è detta universale, mentre le *promising formula* che non sono universali vengono dette esistenziali. Infine le *promising formula* non della forma $\neg A(\alpha U\beta)$ sono dette ristrette.

Quasi-soddisfazione

Definizione 5

Sia M un DAG i cui stati sono copie di stati di T_0, B uno stato di M e ψ una *promising formula* esistenziale. Si dice che M, B quasi-soddisfa al finito ψ (rispettivamente: che M, B propaga ψ) se esiste un cammino $P = D_0, ..., D_m$ in M con $D_0 = B$ che rispetta le condizioni riportate nella seguente tabella.

ψ	Quasi-soddisfazione al finito	Propagazione
$EX\alpha$	$m = 1$ ed $\alpha \in D_1$	non definita
$\neg AX\alpha$	$m = 1$ ed $\alpha \notin D_1$	non definita
$E(\alpha U \beta)$	$\beta \in D_m$ e per ogni $0 \leq i < m$ risulta che $\alpha \in D_i$	D_m è uno stato terminale di M e per ogni $0 \leq i \leq m$ risulta che $\alpha, \psi \in D_i$
$\neg A(\alpha U \beta)$	$\alpha \notin D_m$ e per ogni $0 \leq i \leq m$ risulta che $\beta \notin D_i$	D_m è uno stato terminale di M e per ogni $0 \leq i \leq m$ risulta che $\beta \notin D_i$ e $\psi \in D_i$

Definizione (Continua)

Si dice che M, B quasi-soddisfa ψ se M, B quasi-soddisfa al finito ψ oppure se M, B propaga ψ.
Ad ogni formula ψ universale, cioè del tipo $AX\alpha, \neg EX\alpha, A(\alpha U\beta)$ o $\neg E(\alpha U\beta)$, si può far corrispondere una *promising formula* esistenziale, ovvero la formula $EX\alpha, \neg AX\alpha, E(\alpha U\beta)$ o $\neg A(\alpha U\beta)$ rispettivamente. Si estende il concetto di quasi-soddisfazione al finito, propagazione e quasi-soddisfazione alle formule universali chiedendo che ogni cammino di M che inizia in B e finisce in uno stato terminale di M soddisfi la stessa condizione della corrispettiva *promising formula* esistenziale.

$DAG(B, \psi)$

Per ogni stato $B \in S$ e per ogni *promising formula* ristretta $\psi \in B$ si vuole definire una sottostruttura $DAG(B, \psi)$ di T tale che

- sia un DAG
- quasi-soddisfi al finito ψ

$DAG(B, \psi)$

Per ogni stato $B \in S$ e per ogni *promising formula* ristretta $\psi \in B$ si vuole definire una sottostruttura $DAG(B, \psi)$ di T tale che

- sia un DAG
- quasi-soddisfi al finito ψ
- ogni *promising formula* esistenziale presente negli stati non terminali di $DAG(B, \psi)$ sia quasi-soddisfatta.

Se ψ è del tipo $A(\alpha U \beta)$ la regola AU garantisce l'esistenza di almeno una sottostruttura con tali proprietà. Se invece ψ è una *promising formula* ristretta ed esistenziale è semplice determinare una sottostruttura con le proprietà richieste.

$DAG(B, \psi)$

Per ogni stato $B \in S$ e per ogni *promising formula* ristretta $\psi \in B$ si vuole definire una sottostruttura $DAG(B, \psi)$ di T tale che

- sia un DAG
- quasi-soddisfi al finito ψ
- ogni *promising formula* esistenziale presente negli stati non terminali di $DAG(B, \psi)$ sia quasi-soddisfatta.

Se ψ è del tipo $A(\alpha U \beta)$ la regola AU garantisce l'esistenza di almeno una sottostruttura con tali proprietà. Se invece ψ è una *promising formula* ristretta ed esistenziale è semplice determinare una sottostruttura con le proprietà richieste.

$DAG(B, \psi)$

Per ogni stato $B \in S$ e per ogni *promising formula* ristretta $\psi \in B$ si vuole definire una sottostruttura $DAG(B, \psi)$ di T tale che

- sia un DAG
- quasi-soddisfi al finito ψ
- ogni *promising formula* esistenziale presente negli stati non terminali di $DAG(B, \psi)$ sia quasi-soddisfatta.

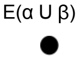
Se ψ è del tipo $A(\alpha U \beta)$ la regola AU garantisce l'esistenza di almeno una sottostruttura con tali proprietà. Se invece ψ è una *promising formula* ristretta ed esistenziale è semplice determinare una sottostruttura con le proprietà richieste.

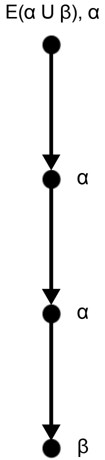
$DAG(B, \psi)$

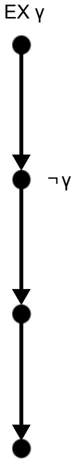
Per ogni stato $B \in S$ e per ogni *promising formula* ristretta $\psi \in B$ si vuole definire una sottostruttura $DAG(B, \psi)$ di T tale che

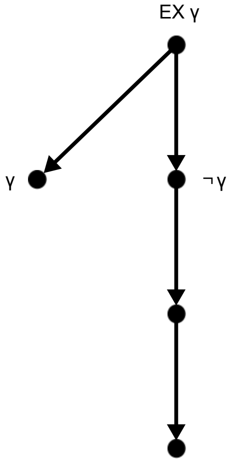
- sia un DAG
- quasi-soddisfi al finito ψ
- ogni *promising formula* esistenziale presente negli stati non terminali di $DAG(B, \psi)$ sia quasi-soddisfatta.

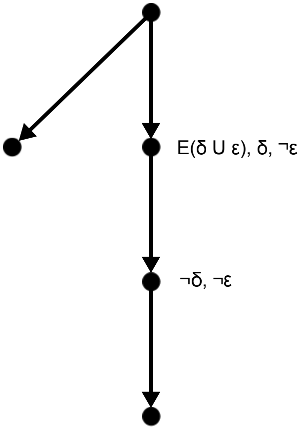
Se ψ è del tipo $A(\alpha U \beta)$ la regola AU garantisce l'esistenza di almeno una sottostruttura con tali proprietà. Se invece ψ è una *promising formula* ristretta ed esistenziale è semplice determinare una sottostruttura con le proprietà richieste.

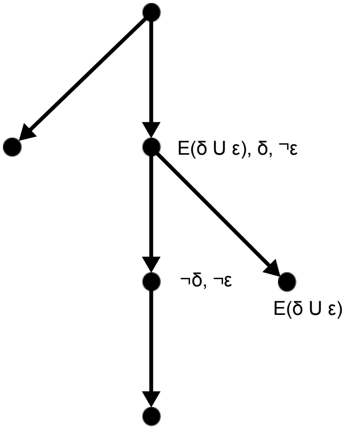

E(α U β)


E(α U β), α
α
α
β


EX γ
¬ γ


EX γ
γ
¬ γ


E(δ U ε), δ, ¬ε
¬δ, ¬ε


E(δ U ε), δ, ¬ε
¬δ, ¬ε
E(δ U ε)

Frammenti

Definizione 6

Sia B uno stato di T. Un DAG finito $F_B = (S_B, R_B, f_B)$, il cui unico stato privo di transizioni entranti è B, è detto un frammento di B se esiste una funzione di proiezione $\sim$: $S_B \to S$, con S insieme degli stati del *tableau* $T = (S, R)$, tale che

- per ogni $(C, D) \in R_B$ risulta $(\widetilde{C}, \widetilde{D}) \in R$;
- per ogni $C \in S_B$ risulta $f_B(C) = \widetilde{C} \cap L$;
- per ogni $\psi \in \widetilde{B}$ *promising formula* ristretta, la coppia F_B, B quasi-soddisfa al finito ψ;
- per ogni $C \in S_B$ non terminale e per ogni $\psi \in \widetilde{C}$ *promising formula* (esistenziale o universale), F_B, C quasi-soddisfa ψ.

Frammenti

Definizione 6

Sia B uno stato di T. Un DAG finito $F_B = (S_B, R_B, f_B)$, il cui unico stato privo di transizioni entranti è B, è detto un frammento di B se esiste una funzione di proiezione $\sim: S_B \to S$, con S insieme degli stati del *tableau* $T = (S, R)$, tale che

- per ogni $(C, D) \in R_b$ risulta $(\widetilde{C}, \widetilde{D}) \in R$;
- per ogni $C \in S_B$ risulta $f_B(C) = \widetilde{C} \cap L$;
- per ogni $\psi \in \widetilde{B}$ *promising formula* ristretta, la coppia F_B, B quasi-soddisfa al finito ψ;
- per ogni $C \in S_B$ non terminale e per ogni $\psi \in \widetilde{C}$ *promising formula* (esistenziale o universale), F_B, C quasi-soddisfa ψ.

Frammenti

Definizione 6

Sia B uno stato di T. Un DAG finito $F_B = (S_B, R_B, f_B)$, il cui unico stato privo di transizioni entranti è B, è detto un frammento di B se esiste una funzione di proiezione $\sim: S_B \to S$, con S insieme degli stati del *tableau* $T = (S, R)$, tale che

- per ogni $(C, D) \in R_b$ risulta $(\widetilde{C}, \widetilde{D}) \in R$;
- per ogni $C \in S_B$ risulta $f_B(C) = \widetilde{C} \cap L$;
- per ogni $\psi \in \widetilde{B}$ *promising formula* ristretta, la coppia F_B, B quasi-soddisfa al finito ψ;
- per ogni $C \in S_B$ non terminale e per ogni $\psi \in \widetilde{C}$ *promising formula* (esistenziale o universale), F_B, C quasi-soddisfa ψ.

Frammenti

Definizione 6

Sia B uno stato di T. Un DAG finito $F_B = (S_B, R_B, f_B)$, il cui unico stato privo di transizioni entranti è B, è detto un frammento di B se esiste una funzione di proiezione $\sim: S_B \to S$, con S insieme degli stati del *tableau* $T = (S, R)$, tale che

- per ogni $(C, D) \in R_b$ risulta $(\widetilde{C}, \widetilde{D}) \in R$;
- per ogni $C \in S_B$ risulta $f_B(C) = \widetilde{C} \cap L$;
- per ogni $\psi \in \widetilde{B}$ *promising formula* ristretta, la coppia F_B, B quasi-soddisfa al finito ψ;
- per ogni $C \in S_B$ non terminale e per ogni $\psi \in \widetilde{C}$ *promising formula* (esistenziale o universale), F_B, C quasi-soddisfa ψ.

Frammenti

Definizione 6

Sia B uno stato di T. Un DAG finito $F_B = (S_B, R_B, f_B)$, il cui unico stato privo di transizioni entranti è B, è detto un frammento di B se esiste una funzione di proiezione $\sim: S_B \to S$, con S insieme degli stati del *tableau* $T = (S, R)$, tale che

- per ogni $(C, D) \in R_b$ risulta $(\widetilde{C}, \widetilde{D}) \in R$;
- per ogni $C \in S_B$ risulta $f_B(C) = \widetilde{C} \cap L$;
- per ogni $\psi \in \widetilde{B}$ *promising formula* ristretta, la coppia F_B, B quasi-soddisfa al finito ψ;
- per ogni $C \in S_B$ non terminale e per ogni $\psi \in \widetilde{C}$ *promising formula* (esistenziale o universale), F_B, C quasi-soddisfa ψ.

Sia $B \in T$ e sia $H = \{\psi_1, ..., \psi_n\}$ l'insieme delle *promising formula* ristrette presenti in B.

- $F_1 := DAG(B, \psi_1)$.
- Se F_i quasi-soddisfa al finito ψ_{i+1} allora $F_{i+1} := F_i$. Altrimenti

Sia $B \in T$ e sia $H = \{\psi_1, ..., \psi_n\}$ l'insieme delle *promising formula* ristrette presenti in B.

- $F_1 := DAG(B, \psi_1)$.
- Se F_i quasi-soddisfa al finito ψ_{i+1} allora $F_{i+1} := F_i$. Altrimenti
  - Se ψ_{i+1} è esistenziale allora esiste un cammino lungo cui ψ_{i+1} si propaga. Detto D l'ultimo stato di tale cammino, F_{i+1} si ottiene da F_i attaccando a D una copia di $DAG(D, \psi_{i+1})$.
  - Se ψ_{i+1} è universale allora ad ogni stato terminale D [illegible] che esiste un cammino da B a D, che non quasi-soddisfa al finito ψ_{i+1} si attacca una copia di $DAG(D, \psi_{i+1})$.

Sia $B \in T$ e sia $H = \{\psi_1, ..., \psi_n\}$ l'insieme delle *promising formula* ristrette presenti in B.

- $F_1 := DAG(B, \psi_1)$.
- Se F_i quasi-soddisfa al finito ψ_{i+1} allora $F_{i+1} := F_i$. Altrimenti
    - Se ψ_{i+1} è esistenziale allora esiste un cammino lungo cui ψ_{i+1} si propaga. Detto D l'ultimo stato di tale cammino, F_{i+1} si ottiene da F_i attaccando a D una copia di $DAG(D, \psi_{i+1})$.
    - Se ψ_{i+1} è universale allora ad ogni stato terminale D_j di F_i tale che esiste un cammino da B a D_j che non quasi-soddisfa al finito ψ_{i+1} si attacca una copia di $DAG(D_j, \psi_{i+1})$.

Sia $B \in T$ e sia $H = \{\psi_1, ..., \psi_n\}$ l'insieme delle *promising formula* ristrette presenti in B.

- $F_1 := DAG(B, \psi_1)$.
- Se F_i quasi-soddisfa al finito ψ_{i+1} allora $F_{i+1} := F_i$. Altrimenti
  - Se ψ_{i+1} è esistenziale allora esiste un cammino lungo cui ψ_{i+1} si propaga. Detto D l'ultimo stato di tale cammino, F_{i+1} si ottiene da F_i attaccando a D una copia di $DAG(D, \psi_{i+1})$.
  - Se ψ_{i+1} è universale allora ad ogni stato terminale D_j di F_i tale che esiste un cammino da B a D_j che non quasi-soddisfa al finito ψ_{i+1} si attacca una copia di $DAG(D_j, \psi_{i+1})$.

Sia $B \in T$ e sia $H = \{\psi_1, ..., \psi_n\}$ l'insieme delle *promising formula* ristrette presenti in B.

- $F_1 := DAG(B, \psi_1)$.
- Se F_i quasi-soddisfa al finito ψ_{i+1} allora $F_{i+1} := F_i$. Altrimenti
  - Se ψ_{i+1} è esistenziale allora esiste un cammino lungo cui ψ_{i+1} si propaga. Detto D l'ultimo stato di tale cammino, F_{i+1} si ottiene da F_i attaccando a D una copia di $DAG(D, \psi_{i+1})$.
  - Se ψ_{i+1} è universale allora ad ogni stato terminale D_j di F_i tale che esiste un cammino da B a D_j che non quasi-soddisfa al finito ψ_{i+1} si attacca una copia di $DAG(D_j, \psi_{i+1})$.

Modello

A partire dai frammenti è possibile costruire un modello M che contiene uno stato B tale che M, B soddisfa la formula iniziale φ.
Sia B uno stato B che contiene φ .

- $M_0 := (\{B\}, \emptyset)$ e $V_0 := \emptyset$ e $T_0 := \{B\}$.
- Fintanto che l'insieme degli stati terminali è non vuoto:

Modello

A partire dai frammenti è possibile costruire un modello M che contiene uno stato B tale che M, B soddisfa la formula iniziale φ. Sia B uno stato B che contiene φ .

- $M_0 := \big(\{B\}, \emptyset\big)$ e $V_0 := \emptyset$ e $T_0 := \{B\}$.
- Fintanto che l'insieme degli stati terminali è non vuoto:
  - Si sceglie un $C \in T_i$ e lo si espande, ovvero si attacca a C una copia di $Frag(C)$. Si pone $V_{i+1} := V_i \cup \{C\}$.

Modello

A partire dai frammenti è possibile costruire un modello M che contiene uno stato B tale che M, B soddisfa la formula iniziale φ. Sia B uno stato B che contiene φ .

- $M_0 := (\{B\}, \emptyset)$ e $V_0 := \emptyset$ e $T_0 := \{B\}$.
- Fintanto che l'insieme degli stati terminali è non vuoto:
  - Si sceglie un $C \in T_i$ e lo si espande, ovvero si attacca a C una copia di $Frag(C)$. Si pone $V_{i+1} := V_i \cup \{C\}$.
  - Se uno stato terminale di $Frag(C)$ ha la stessa $\sim$ immagine di uno stato precedentemente espanso, i due stati vengono identificati.
  - Detto T l'insieme di tutti gli stati terminali di $Frag(C)$ sopravvissuti, si pone $T_{i+1} = T_i \cup T$.

Modello

A partire dai frammenti è possibile costruire un modello M che contiene uno stato B tale che M, B soddisfa la formula iniziale φ. Sia B uno stato B che contiene φ .

- $M_0 := (\{B\}, \emptyset)$ e $V_0 := \emptyset$ e $T_0 := \{B\}$.
- Fintanto che l'insieme degli stati terminali è non vuoto:
  - Si sceglie un $C \in T_i$ e lo si espande, ovvero si attacca a C una copia di $Frag(C)$. Si pone $V_{i+1} := V_i \cup \{C\}$.
  - Se uno stato terminale di $Frag(C)$ ha la stessa $\sim$ immagine di uno stato precedentemente espanso, i due stati vengono identificati.
  - Detto T l'insieme di tutti gli stati terminali di $Frag(C)$ sopravvissuti, si pone $T_{i+1} = T_i \cup T$.

Modello

A partire dai frammenti è possibile costruire un modello M che contiene uno stato B tale che M, B soddisfa la formula iniziale φ.
Sia B uno stato B che contiene φ .

- $M_0 := \big(\{B\}, \emptyset\big)$ e $V_0 := \emptyset$ e $T_0 := \{B\}$.
- Fintanto che l'insieme degli stati terminali è non vuoto:
  - Si sceglie un $C \in T_i$ e lo si espande, ovvero si attacca a C una copia di $Frag(C)$. Si pone $V_{i+1} := V_i \cup \{C\}$.
  - Se uno stato terminale di $Frag(C)$ ha la stessa $\sim$ immagine di uno stato precedentemente espanso, i due stati vengono identificati.
  - Detto T l'insieme di tutti gli stati terminali di $Frag(C)$ sopravvissuti, si pone $T_{i+1} = T_i \cup T$.

Modello

A partire dai frammenti è possibile costruire un modello M che contiene uno stato B tale che M, B soddisfa la formula iniziale φ. Sia B uno stato B che contiene φ .

- $M_0 := (\{B\}, \emptyset)$ e $V_0 := \emptyset$ e $T_0 := \{B\}$.
- Fintanto che l'insieme degli stati terminali è non vuoto:
  - Si sceglie un $C \in T_i$ e lo si espande, ovvero si attacca a C una copia di $Frag(C)$. Si pone $V_{i+1} := V_i \cup \{C\}$.
  - Se uno stato terminale di $Frag(C)$ ha la stessa $\sim$ immagine di uno stato precedentemente espanso, i due stati vengono identificati.
  - Detto T l'insieme di tutti gli stati terminali di $Frag(C)$ sopravvissuti, si pone $T_{i+1} = T_i \cup T$.

Terminazione

Proposizione 1

L'algoritmo Frag termina sempre.

Dimostrazione.

Ad ogni iterazione viene aggiunto n nuovo stato all'insieme V_i degli stati espansi, dunque la successione degli insiemi $\widetilde{V}_i$, ovvero delle $\sim$ immagini dei V_i, è non decrescente. Ma i $\widetilde{V}_i$ sono contenuti nell'insieme degli stati di T, che è finito, dunque deve esistere un j tale che gli stati terminali dei frammenti relativi agli stati di T_j potranno essere identificati con stati di V_j. $\square$

Terminazione

Proposizione 1

L'algoritmo Frag termina sempre.

Dimostrazione.

Ad ogni iterazione viene aggiunto n nuovo stato all'insieme V_i degli stati espansi, dunque la successione degli insiemi $\widetilde{V}_i$, ovvero delle $\sim$ immagini dei V_i, è non decrescente. Ma i $\widetilde{V}_i$ sono contenuti nell'insieme degli stati di T, che è finito, dunque deve esistere un j tale che gli stati terminali dei frammenti relativi agli stati di T_j potranno essere identificati con stati di V_j. $\square$

Sia $M = M_n = Mod(T, \varphi)$ il modello restituito dall'algoritmo Mod e sia $V := V_n$ l'insieme degli stati espansi nel corso dell'intero algoritmo.

Proposizione 2

Sia A un ciclo in M. Allora esistono un sottocammino P di A ed un $C_i \in V$ tali che P è un cammino in $Frag(C_i)$ da C_i ad uno stato terminale di $Frag(C_i)$.

Proposizione 3

Sia D uno stato di M e sia ψ una promising formula contenuta in D tale che la struttura $(\{D\}, \emptyset)$ non quasi-soddisfa al finito ψ. Allora esiste un $C_k \in V$ tale che D è uno stato non terminale di $Frag(C_k)$.

Sia $M = M_n = Mod(T, \varphi)$ il modello restituito dall'algoritmo Mod e sia $V := V_n$ l'insieme degli stati espansi nel corso dell'intero algoritmo.

Proposizione 2

Sia A un ciclo in M. Allora esistono un sottocammino P di A ed un $C_i \in V$ tali che P è un cammino in $Frag(C_i)$ da C_i ad uno stato terminale di $Frag(C_i)$.

Proposizione 3

Sia D uno stato di M e sia ψ una promising formula contenuta in D tale che la struttura $(\{D\}, \emptyset)$ non quasi-soddisfa al finito ψ. Allora esiste un $C_k \in V$ tale che D è uno stato non terminale di $Frag(C_k)$.

Sia $M = M_n = Mod(T, \varphi)$ il modello restituito dall'algoritmo Mod e sia $V := V_n$ l'insieme degli stati espansi nel corso dell'intero algoritmo.

Proposizione 2

Sia A un ciclo in M. Allora esistono un sottocammino P di A ed un $C_i \in V$ tali che P è un cammino in $Frag(C_i)$ da C_i ad uno stato terminale di $Frag(C_i)$.

Proposizione 3

Sia D uno stato di M e sia ψ una promising formula contenuta in D tale che la struttura $(\{D\}, \emptyset)$ non quasi-soddisfa al finito ψ. Allora esiste un $C_k \in V$ tale che D è uno stato non terminale di $Frag(C_k)$.

Correttezza

Teorema 1

Se il tableau T contiene almeno uno stato contenete φ allora, detto B lo stato selezionato al primo passo dall'algoritmo Mod, risulta $Mod(T, \varphi), B \vDash \varphi$.

Dimostrazione.

Si dimostra per induzione sulla struttura di ψ che per ogni $D \in M = Mod(T, \varphi)$ e per ogni $\psi \in D$ risulta $M, D \vDash \psi$. In particolare risulta $Mod(T, \varphi), B \vDash \varphi$. $\square$

Correttezza

Teorema 1

Se il tableau T contiene almeno uno stato contenete φ allora, detto B lo stato selezionato al primo passo dall'algoritmo Mod, risulta $Mod(T, \varphi), B \models \varphi$.

Dimostrazione.

Si dimostra per induzione sulla struttura di ψ che per ogni $D \in M = Mod(T, \varphi)$ e per ogni $\psi \in D$ risulta $M, D \models \psi$. In particolare risulta $Mod(T, \varphi), B \models \varphi$. $\square$

Sia φ una formula di CTL^{ndn} soddisfacibile e sia t un albero con radice r tale che $t, r \models \varphi$.

Proposizione 4

Si può supporre senza perdita di generalità che l'albero t sia finitamente generato.

Lemma 1

Sia v un nodo di t. Allora $\widetilde{v}^t$, visto come stato di T_0, soddisfa tutte le regole descritte nell'operazione di culling. Inoltre ogni qual volta una di tali regole richiede, per essere soddisfatta, l'esistenza di una sottostruttura di T_0 con determinate proprietà (che dipendono dalla regola considerata), esiste un albero t' finitamente generato che continua a soddisfare φ e che ammette un sottoalbero s' la cui $\sim_{t'}$ immagine è una delle sottostrutture di T_0 cercate.

Sia φ una formula di CTL^{ndn} soddisfacibile e sia t un albero con radice r tale che $t, r \models \varphi$.

Proposizione 4

Si può supporre senza perdita di generalità che l'albero t sia finitamente generato.

Lemma 1

Sia v un nodo di t. Allora $\widetilde{v}^t$, *visto come stato di* T_0, *soddisfa tutte le regole descritte nell'operazione di culling. Inoltre ogni qual volta una di tali regole richiede, per essere soddisfatta, l'esistenza di una sottostruttura di* T_0 *con determinate proprietà (che dipendono dalla regola considerata), esiste un albero* t' *finitamente generato che continua a soddisfare* φ *e che ammette un sottoalbero* s' *la cui* $\sim_{t'}$ *immagine è una delle sottostrutture di* T_0 *cercate.*

Sia φ una formula di CTL^{ndn} soddisfacibile e sia t un albero con radice r tale che $t, r \models \varphi$.

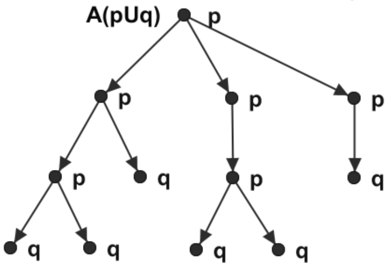
Proposizione 4

Si può supporre senza perdita di generalità che l'albero t sia finitamente generato.

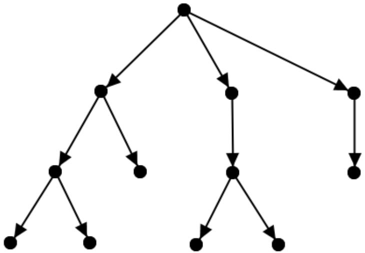
Lemma 1

Sia v un nodo di t. Allora $\widetilde{v}^t$, visto come stato di T_0, soddisfa tutte le regole descritte nell'operazione di culling. Inoltre ogni qual volta una di tali regole richiede, per essere soddisfatta, l'esistenza di una sottostruttura di T_0 con determinate proprietà (che dipendono dalla regola considerata), esiste un albero t' finitamente generato che continua a soddisfare φ e che ammette un sottoalbero s' la cui $\sim_{t'}$ immagine è una delle sottostrutture di T_0 cercate.

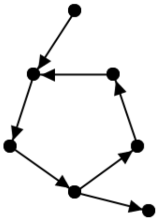
$t|_v$



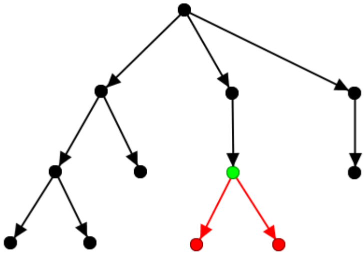
$t|_v$



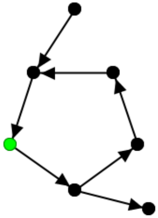
T_0



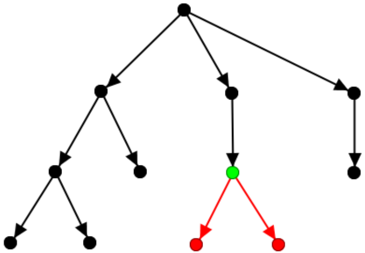
$t|_v$

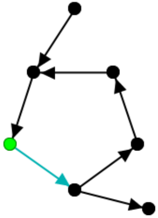


T_0

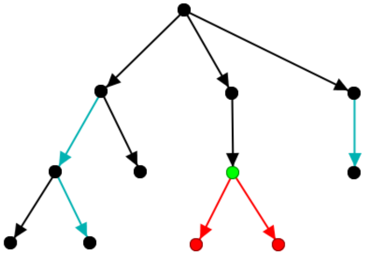


$t|_v$ T_0

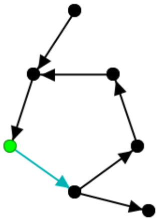




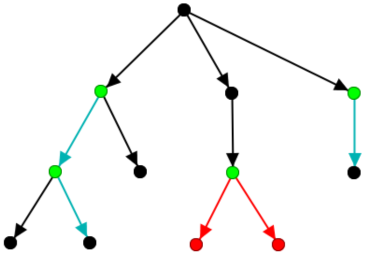
$t|_v$



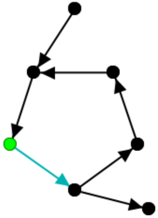
T_0



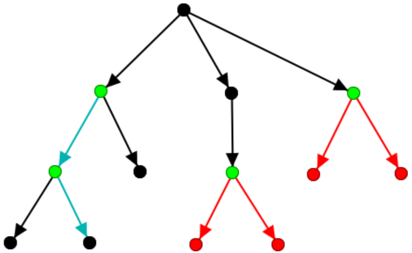
$t|_v$

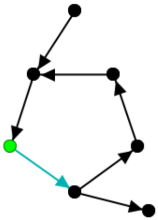


T_0

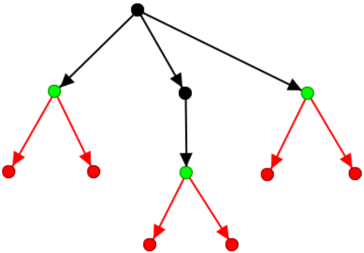


$t|_v$ $\quad$ T_0

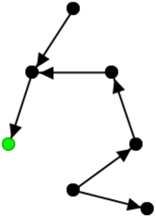




$\mathbf{t}|_{\mathbf{v}}$



$\mathbf{T}_0$



Teorema 2

Sia φ una formula di CTL^{ndn} soddisfacibile. Allora il tableau $T = (S, R)$ ottenuto dopo l'operazione di culling contiene almeno uno stato B tale $\varphi \in B$, dunque l'algoritmo fornisce come risposta "sì".

Dimostrazione.

Si considera il sottoinsieme di T_0 definito da $G := \{\widetilde{v}^t : t \text{ è un albero finitamente generato la cui radice soddisfa } \varphi \text{ e } v \in \mathrm{dom}(t)\}$. Per il Lemma 1 nessuno degli stati di G può essere rimosso durante l'operazione di *culling* se tutti gli altri stati di G sono ancora presenti, dunque per ogni i risulta $G \subseteq T_i$. Dato un albero t tale che $t, r \models \varphi$, si ha che $\widetilde{r}^t \in G \subseteq T_n = T$ e basta prendere $B := \widetilde{r}^t$. □

Teorema 2

Sia φ una formula di CTL^{ndn} soddisfacibile. Allora il tableau $T = (S, R)$ ottenuto dopo l'operazione di culling contiene almeno uno stato B tale $\varphi \in B$, dunque l'algoritmo fornisce come risposta "sì".

Dimostrazione.

Si considera il sottoinsieme di T_0 definito da $G := \{\widetilde{v}^t : t$ è un albero finitamente generato la cui radice soddisfa φ e $v \in \text{dom}(t)\}$. Per il Lemma 1 nessuno degli stati di G può essere rimosso durante l'operazione di *culling* se tutti gli altri stati di G sono ancora presenti, dunque per ogni i risulta $G \subseteq T_i$. Dato un albero t tale che $t, r \vDash \varphi$, si ha che $\widetilde{r}^t \in G \subseteq T_n = T$ e basta prendere $B := \widetilde{r}^t$. □