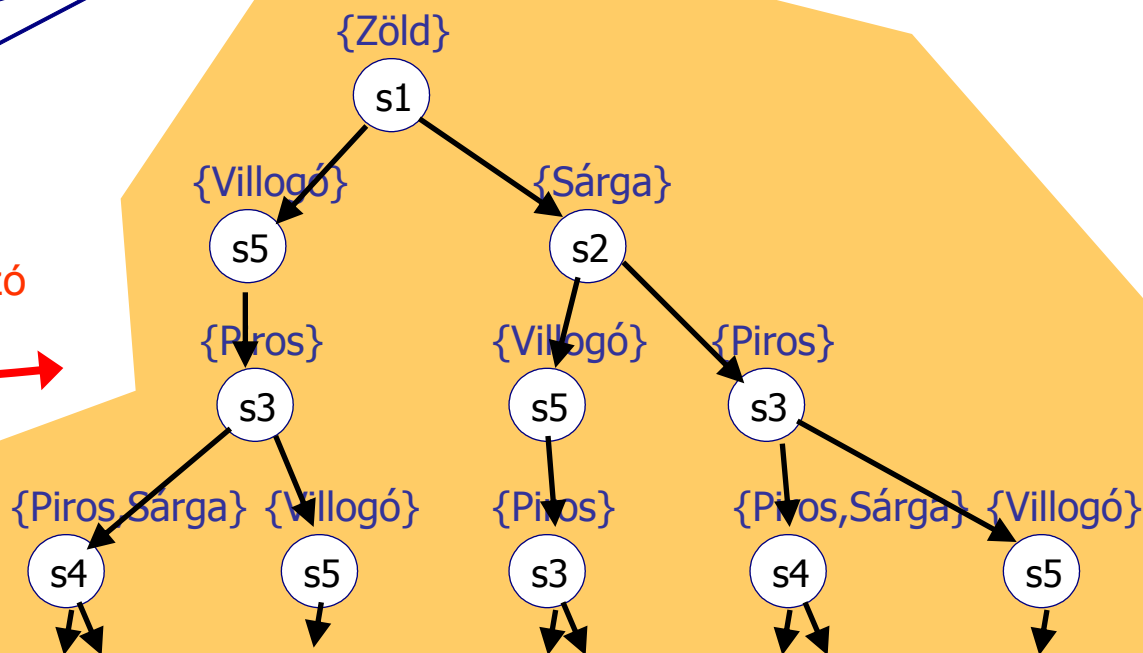
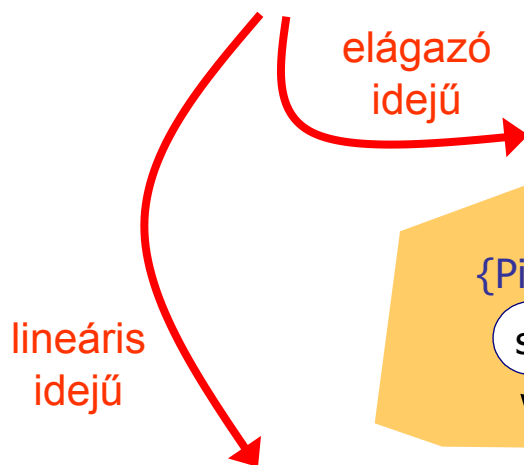
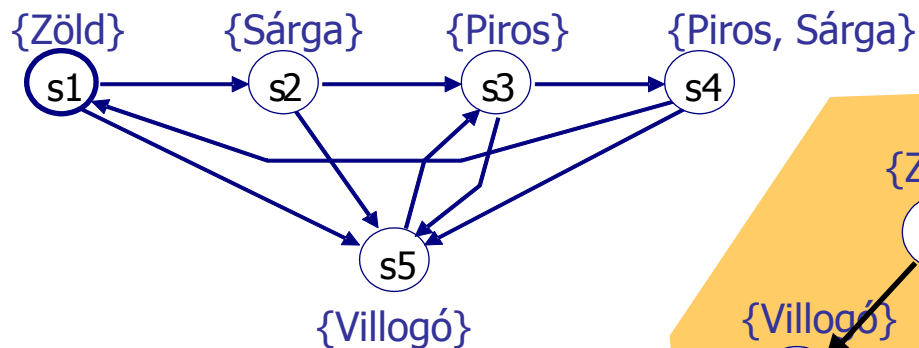


Elágazó idejű temporális logikák: Computational Tree Logic (CTL, CTL*)

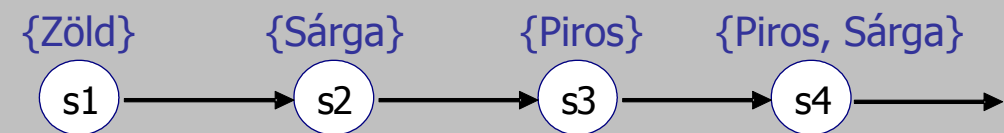
Majzik István

BME Méréstechnika és Információs Rendszerek Tanszék

Lineáris és elágazó idejű temporális logikák



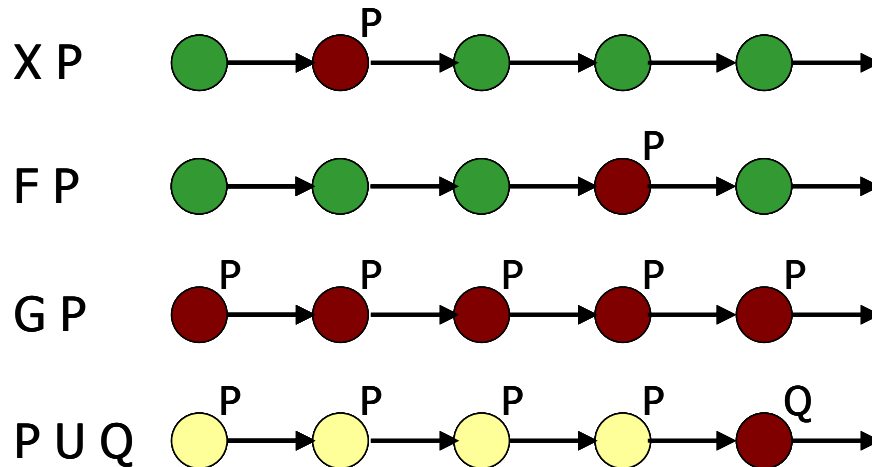
Logikai idő
elágazó idővonalak
mentén



Logikai idő mint idővonal (egy-egy lefutás)

Útvonalakra vonatkozó operátorok (lineáris TL)

- Egy-egy útvonal állapotain értelmezett operátorok
- F , G , X , U informálisan:
 - $F p$: „Valamikor p ”, egy elérhető állapotban igaz lesz p
 - $G p$: „Mindig p ”, minden elérhető állapotban igaz lesz p
 - $X p$: „Következő p ”, a következő állapotban igaz lesz p
 - $p U q$: „ p amíg q ”, egy elérhető állapotban igaz lesz q , és addig minden állapotban igaz p

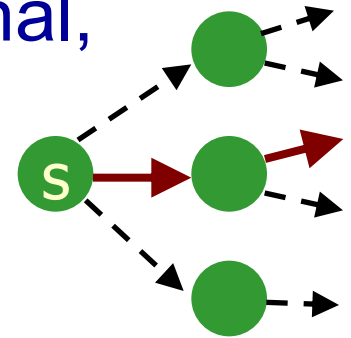


Elágazásokra vonatkozó operátorok (elágazó TL)

Egy-egy állapotban útvonal kvantorok:

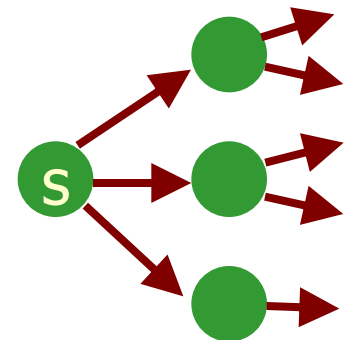
- $E\ p$ (Exists p): Létezik legalább egy útvonal, ahol a p követelmény teljesül

- Egy lehetséges elágazás mentén vizsgál
- Egzisztenciális operátor



- $A\ p$ (forAll p): Minden útvonalra fennáll, hogy a p követelmény teljesül

- Minden lehetséges elágazást magába foglal
- Univerzális operátor



Elágazó idejű temporális logikák

- **CTL***: Computational Tree Logic *

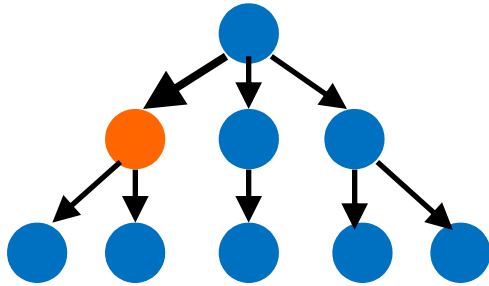
Tetszőleges kombinációja a következőknek:

- **Útvonal** kvantorok (E, A),
- Útvonalon értelmezett (sorrendi) operátorok (F, G, X, U)

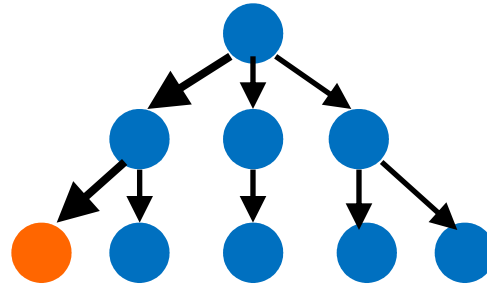
- **CTL**: Computational Tree Logic

- Útvonalon értelmezett operátort mindig közvetlenül meg kell előznie **útvonal kvantornak**
- Útvonalon értelmezett operátorok nem kombinálhatók

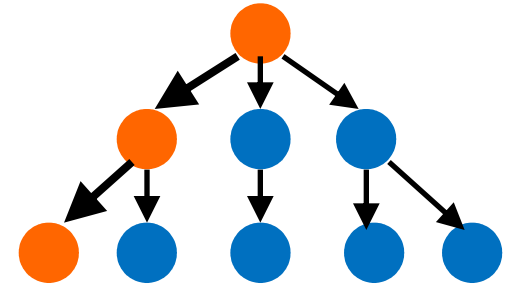
CTL operátorok (példák)



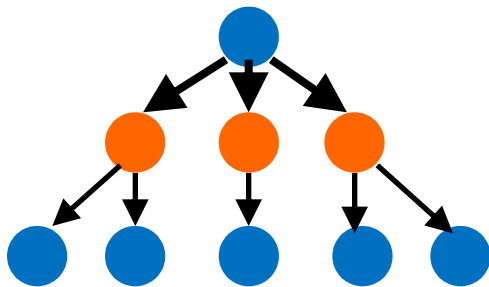
EX p



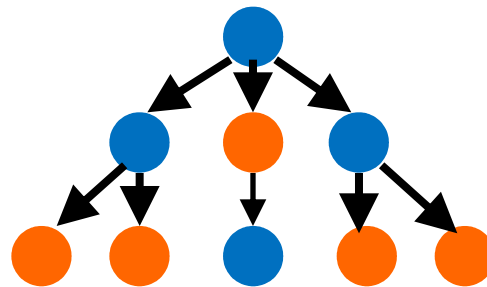
EF p



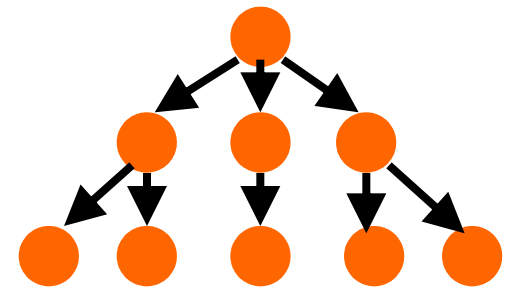
EG p



AX p



AF p



AG p

Példák CTL illetve CTL* kifejezésekre

- $A(p \Rightarrow F q)$:
 - Minden útvonalra igaz, hogy amennyiben p igaz az útvonal kezdetén, akkor ezt a jövőben olyan állapot fogja követni, ahol q igaz
- $E(p \wedge G q)$:
 - Létezik olyan útvonal, hogy ennek kezdetén p igaz és minden állapotán q is igaz
- $E(XXX p \vee F q)$:
 - Létezik olyan útvonal, hogy
 - vagy ennek negyedik állapotán igaz p ,
 - vagy valamikor a jövőben igaz q
- $AG EF p$:
 - A rendszer bármelyik elérhető állapotára igaz, hogy onnan indulva olyan állapotba vihető a rendszer, ahol p igaz
- $EF AG p$:
 - Lehetséges, hogy a rendszer olyan állapotba kerül, hogy utána p mindig igaz lesz
- $AG (p \Rightarrow AF q)$:
 - A rendszer bármelyik elérhető állapotára igaz, hogy ha ott p igaz, akkor bármely onnan induló úton valamikor q mindenképpen igaz lesz

CTL* formális szintaxis

- **Állapot-kifejezések:** Állapotokon kiértékelhető
 - **S1:** Minden P atomi kijelentés egy állapot-kifejezés
 - **S2:** Ha p és q állapot-kifejezések, $\neg p$ és $p \wedge q$ is
 - **S3:** Ha p útvonal-kifejezés, akkor $E p$ és $A p$ állapot-kifejezések.
- **Útvonal-kifejezések:** Útvonalakon kiértékelhető
 - **P1:** Minden állapot-kifejezés útvonal-kifejezés
 - **P2:** Ha p és q útvonal-kifejezések, akkor $\neg p$ és $p \wedge q$ is
 - **P3:** Ha p és q útvonal-kifejezések, akkor $X p$ és $p U q$ is
- **Érvényes CTL* kifejezések:** A fenti szabályok alapján generált állapot-kifejezések

CTL* szemantika: Jelölések

- $M=(S, R, L)$ Kripke-struktúra
- $\pi=(s_0, s_1, s_2, \dots)$ az M egy útvonala, ahol s_0 a kezdőállapot és $\forall i \geq 0: (s_i, s_{i+1}) \in R$.
- $\pi^i=(s_i, s_{i+1}, s_{i+2}, \dots)$ a π útvonal szuffixe i -től.
- $M, \pi \models p$ jelentése:
az M modellben a π útvonalon igaz p
- $M, s \models p$ jelentése:
az M modellben az s állapotban igaz p

CTL* szemantika: Állapot-kifejezések

- **S1:** $M, s \models P$ a.cs.a. $P \in L(s)$
- **S2:**
 - $M, s \models \neg p$ a.cs.a. $M, s \models p$ nem igaz
 - $M, s \models p \wedge q$ a.cs.a. $M, s \models p$ és $M, s \models q$
- **S3:**
 - $M, s \models E p$ (ahol p útvonal-kifejezés)
a.cs.a. létezik $\pi = (s_0, s_1, s_2, \dots)$ útvonal M -ben
 $s = s_0$ mellett, hogy $M, \pi \models p$.
 - $M, s \models A p$ (ahol p útvonal-kifejezés)
a.cs.a. minden $\pi = (s_0, s_1, s_2, \dots)$ útvonalra M -ben
ahol $s = s_0$ fennáll igaz, hogy $M, \pi \models p$.

CTL* szemantika: Útvonal-kifejezések

- **P1:**

- $M, \pi \models p$ (ahol p állapot-kifejezés) a.cs.a. $M, s_0 \models p$.

- **P2:**

- $M, \pi \models \neg p$ a.cs.a. $M, \pi \models p$ nem igaz
- $M, \pi \models p \wedge q$ a.cs.a. $M, \pi \models p$ és $M, \pi \models q$

- **P3:**

- $M, \pi \models X p$ a.cs.a. $M, \pi^1 \models p$
- $M, \pi \models p \cup q$ a.cs.a.
 $\exists j \geq 0 : (M, \pi^j \models q \text{ valamint } \forall 0 \leq k < j: M, \pi^k \models p)$

CTL formális szintaxis (CTL*-hoz képest)

- **Állapot-kifejezések:** Állapotokon kiértékelhető
 - **S1:** Minden P atomi kijelentés egy állapot-kifejezés
 - **S2:** Ha p és q állapot-kifejezések, $\neg p$ és $p \wedge q$ is
 - **S3:** Ha p útvonal-kifejezés, akkor $E p$ és $A p$ állapot-kifejezések.
- **Útvonal-kifejezések:** Útvonalakon kiértékelhető
 - **P1:** Minden állapot-kifejezés útvonal-kifejezés
 - **P2:** Ha p és q útvonal-kifejezések, akkor $\neg p$ és $p \wedge q$ is
 - **P3:** Ha p és q útvonal-kifejezések, akkor $X p$ és $p U q$ is
- Ezek helyett egy szabály:
 - **P0:** Ha p és q állapot-kifejezések, akkor $X p$ és $p U q$ útvonal-kifejezések.

Útvonal-kifejezések nem kombinálhatók,
rögtön hozzájuk „ragad” az útvonal kvantor

CTL formális szemantika

- Állapot-kifejezések:
 - **S1, S2, S3** szabályok (lásd CTL*) változatlanok.
- Útvonal-kifejezések:
 - **P1, P2, P3** helyébe lépő új **P0** szabályra:

P0:

- $M, \pi \models X p$ ahol p állapot-kifejezés
a.cs.a. $M, s_1 \models p$
- $M, \pi \models p \cup q$ ahol p, q állapot-kifejezés a.cs.a
 $\exists j \geq 0 : (M, s_j \models q \text{ valamint } \forall 0 \leq k < j : M, s_k \models p)$

Mintapélda: Kölcsönös kizárás

- Két processzből álló rendszer: P1 és P2
- Processz állapotok a követelmények szempontjából:
 - Kritikus szakaszban van: C1, C2
 - Nem-kritikus szakaszban van: N1, N2
 - Kritikus szakaszba belépni akar: W1, W2
- Atomi kijelentések:
 $AP = \{C1, C2, N1, N2, W1, W2\}$

Mintapélda (folytatás)

- Biztonság: Egyszerre csak egy processz lehet a kritikus szakaszban:

$AG (\neg(C1 \wedge C2))$

- Élőség: Ha egy processz várakozik, akkor előbb-utóbb mindig beléphet:

$AG (W1 \Rightarrow AF(C1))$ illetve $AG (W2 \Rightarrow AF(C2))$

- Egy processz előbb-utóbb megpróbál belépni

$AG (N1 \Rightarrow EF W1)$ illetve $AG (N2 \Rightarrow EF W2)$

- Nem csak ciklikusan (egymást váltva) léphetnek be a kritikus szakaszba:

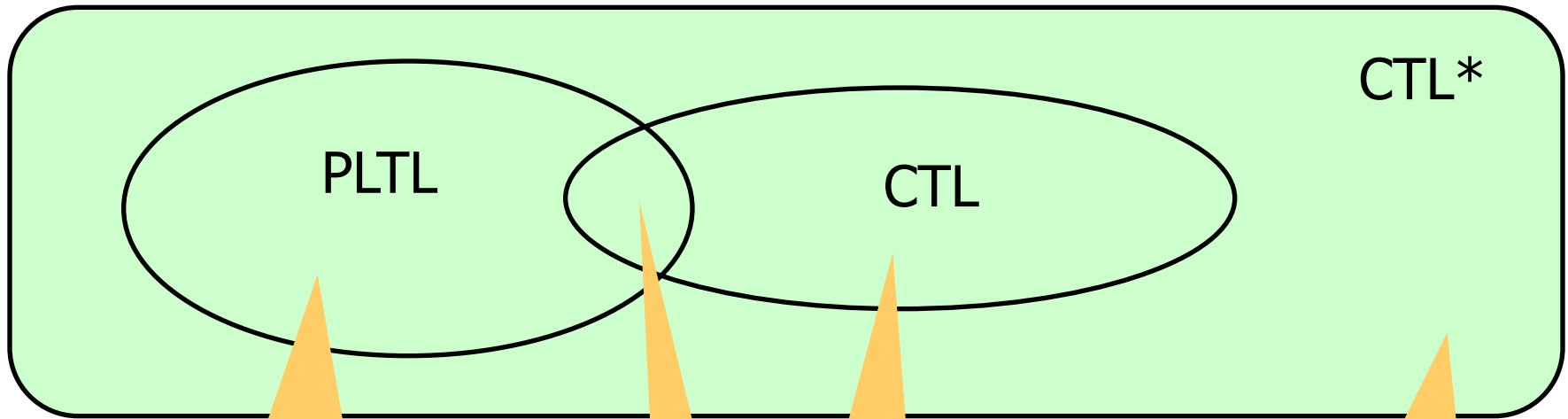
$EF(C1 \wedge E [C1 \cup (\neg C1 \wedge E[\neg C2 \cup C1])])$ illetve

$EF(C2 \wedge E [C2 \cup (\neg C2 \wedge E[\neg C1 \cup C2])])$

Kifejezőképesség

- Egy temporális logika kifejezőképessége nagyobb egy másikénál, ha képes olyan tulajdonság megfogalmazására, amire a másik nem.
- Eddigi tapasztalatok:
 - Lineáris idejű logika nem tudja figyelembe venni a lehetséges elágazásokat
(implicit „minden útra” jellegű vizsgálat lehetséges)
 - CTL kötöttebb, mint a CTL*,
ezért kevesebb tulajdonság megfogalmazására képes
 - CTL* magába foglalja a lehetséges PLTL kifejezéseket

LTL, CTL, CTL* kifejezőképessége



$A(F(p \wedge X q))$
(implicit A
operátor)

$EG(EF p)$

$A(p U q)$
(implicit A
operátor)

$A(F(p \wedge X q)) \vee EG(EF p)$

Kifejezőképesség - formálisan

- TL1 és TL2 logikák kifejezőképessége azonos, ha minden M modellre és annak minden s állapotára:

$$\forall p \in \text{TL1}:$$

$$(\exists q \in \text{TL2} : (M, s \models p \Leftrightarrow M, s \models q)),$$

$$\forall q \in \text{TL2}:$$

$$(\exists p \in \text{TL1} : (M, s \models p \Leftrightarrow M, s \models q))$$

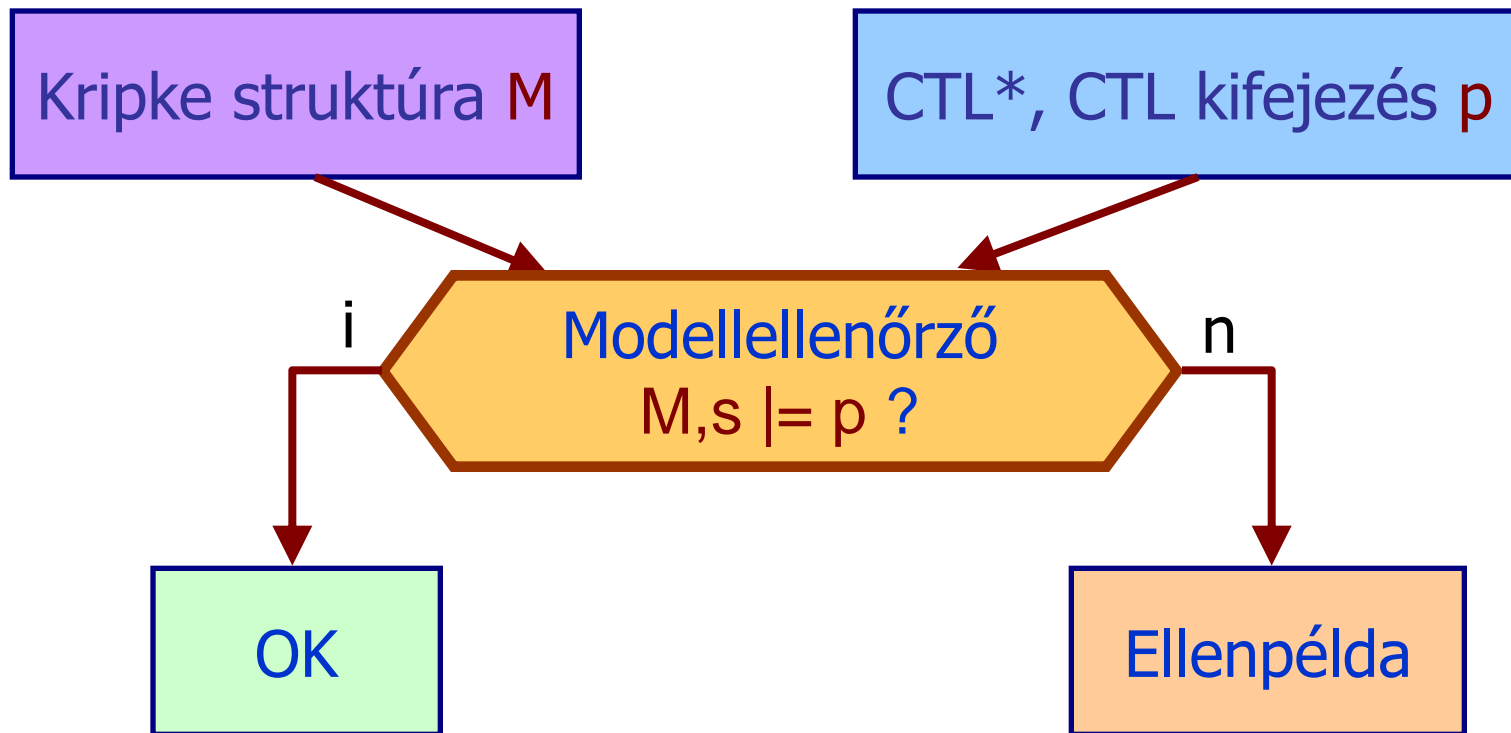
- Ha csak az első kifejezés igaz,
akkor TL1 kevésbé kifejező, mint TL2

FairCTL: A „fair” utak megadása

- A követelmények ellenőrzésének korlátozása a „fair” útvonalakra
 - Triviális ellenpéldák kihagyása (pl. mindig reset, minden üzenet elveszik)
- Módosított kvantorok:
 - A_q : minden „fair” útvonalon
 - E_q : létezik „fair” útvonal
- Az A_q és E_q kvantorokban szereplő q útvonal-kifejezés (fair útvonal) formái:
 - $GF\ r$: az r állapot-kifejezés végtelen sokszor előfordulhat a „fair” útvonal mentén („nincs kiéheztetés”)
 - $FG\ r$: az r állapot-kifejezés csaknem mindig igaz a „fair” útvonal mentén („üzemi állapot beáll”)
- Módosított operátorok jelentése:
 - $A_q F\ p$ jelentése az $A(q \Rightarrow F\ p)$ CTL* kifejezés
 - $E_q G\ p$ jelentése az $E(q \wedge G\ p)$ CTL* kifejezés
- FairCTL előnyei:
 - „Fair” útvonalakra korlátozható az ellenőrzés
 - A CTL modellellenőrzés egyszerűsége megmarad

CTL modellellenőrzés: Szemantika alapon

A modellellenőrzés feladata



Megközelítés

- Globális modellellenőrzés:
 - Adott p CTL kifejezés esetén $\text{Sat}(p)$ számítása:
Azok az állapotok, ahol igaz p
 - Ezeket az állapotokat p -vel címkézzük
 - Ezután $s \in \text{Sat}(p)$ egyszerűen vizsgálható egy adott s (kezdő)állapotra
- $\text{Sat}(p)$ számítása inkrementálisan történik
 - Címkézett állapothalmazok bővítése
 - A bővítés vége: Nem nő a címkézhető állapotok halmaza

CTL modellellenőrzés állapot címkézéssel

- Kifejezések felbontása azok struktúrája alapján, és „belülről kifelé” $\text{Sat}(\dots)$ halmaz számítások:

$\text{AF} (P \wedge E (Q \cup R))$



- Állapotok címkézése: Hol igaz egy adott kifejezés?
 - Kiindulás: KS címkézve van atomi kijelentésekkel
 - Tovább lépés: Címkézés az összetettebb kifejezésekkel
 - Ha p illetve q címkék már vannak, akkor megadható, hol lehet $\neg p$, $p \wedge q$, $\text{EX } p$, $\text{AX } p$, $E(p \cup q)$, $A(p \cup q)$ címke
 - Az eddigi címkéket alapul véve (adott szintaxis szabály szerint képzett) összetettebb kifejezésekkel való címkézés végezhető
 - Inkrementális címkézés a szemantika definíció alapján!

Hol igaz egy adott kifejezés?

- P (atomi kijelentés) azokban az s állapotokban igaz, ahol $P \in L(s)$
 - Itt P címkeként már szerepel a KS-ban
- $\neg P$ azokban az s állapotokban igaz, ahol $P \notin L(s)$
 - Ezek az állapotok $\neg P$ kifejezéssel címkézhetők
- $p \wedge q$ azokban az s állapotokban igaz, ahol p és q is igaz
 - Egy állapot címkézése lehet $p \wedge q$, ha címkéi között már van p és q

Temporális operátorok: EX , AX , $E(U)$, $A(U)$
esetére kell még megadni a címkézési eljárást

Az AX, EX alakú kifejezések

- $EX\ p$ azokban az s állapotokban igaz, amelyeknek van olyan rákövetkező állapota, ahol p igaz
 - Egy állapot címkézése lehet $EX\ p$, ha van olyan rákövetkező állapota, ami p -vel címkézett



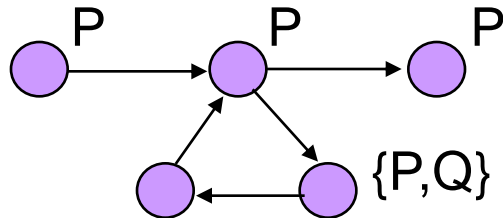
- $AX\ p$ azokban az s állapotokban igaz, amelyeknek minden rákövetkező állapotában p igaz
 - Egy állapot címkézése lehet $AX\ p$, ha minden rákövetkező állapota p -vel címkézett



Az $E(p \cup q)$ kifejezések

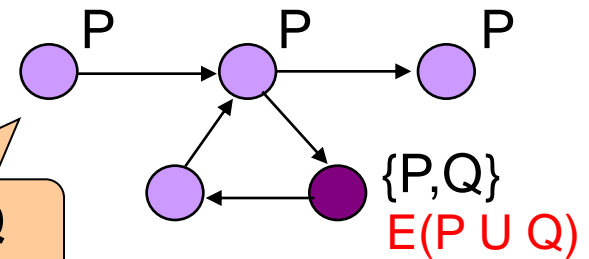
- Hol igaz $E(p \cup q)$?
 - Azonosság: $E(p \cup q) = q \vee (p \wedge EX E(p \cup q))$
 - „Rekurzív” kifejezés...
- Tehát mely állapotok címkézhetők $E(p \cup q)$ -val?
 - Ha s címkézett q -val, vagy
 - ha s címkézett p -vel, és legalább egy rákövetkezője már címkézett $E(p \cup q)$ -val
- Iteráció adódik:
 - q -val már címkézett állapotok adják azokat az állapotokat, ahol először megjelenik az $E(p \cup q)$ címke
 - Ezek megelőző állapotait kell végignézni:
Ha szerepel ott a p címke, akkor rátehető az $E(p \cup q)$ címke is
 - Így visszafelé járjuk be azokat az útvonalakat, amik p -vel címkézett állapotokon keresztül visznek q -val címkézett állapotba

Az $E(P \cup Q)$ címkézés

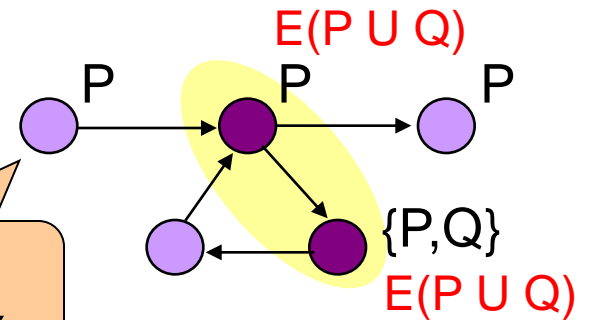


Kripke struktúra a kezdő címkézéssel

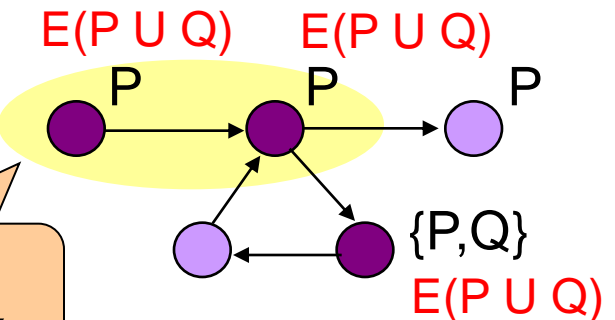
Első lépés: Q



Második lépés: $P \wedge EX$



Harmadik lépés: $P \wedge EX$



- Az iteráció addig tart, míg nő az állapothalmaz (fixpontot érünk el)

Az $A(p \cup q)$ kifejezések

- Hol igaz $A(p \cup q)$?
 - Azonosság: $A(p \cup q) = q \vee (p \wedge AX A(p \cup q))$
 - Ez is „rekurzív” képlet...
- Tehát mely állapotok címkézhetők $A(p \cup q)$ -val?
 - Ha s címkézett q -val, vagy
 - ha s címkézett p -vel, és minden rákövetkezője már címkézett $A(p \cup q)$ -val
- Iteráció adódik:
 - q -val már címkézett állapotok adják azokat az állapotokat, ahol először megjelenik az $A(p \cup q)$ címke
 - Ezek megelőző állapotait kell végignézni:
Ha szerepel ott a p címke, és minden rákövetkező állapotukon szerepel az $A(p \cup q)$ címke, akkor ezekre is rátehető az $A(p \cup q)$ címke

Ezzel a formális szintaxisban használt operátorokat lefedtük!

Címkézés leírása halmazműveletekkel

- A címkézés **bővítése** halmazműveletekkel történik

- Z kezdőhalmaz adott (az előző lépések alapján)
- Az inkrementális bővítés a már címkézett állapotok (legyen Z halmaz) megelőző állapotaira hivatkozik:

$$\text{pre}_E(Z) = \{s \in S \mid \text{létezik olyan } s', \text{ hogy } (s, s') \in R \text{ és } s' \in Z\}$$

$$\text{pre}_A(Z) = \{s \in S \mid \text{minden } s'\text{-re, ahol } (s, s') \in R: s' \in Z\}$$

Kérdés: $\text{pre}_A(Z)$ felírható-e $\text{pre}_E()$ segítségével?

- Példa: $E(P \cup Q)$ esetén:

- Kezdőhalmaz: $X_0 = \{s \mid Q \in L(s)\}$
- Bővítés: $X_{i+1} = X_i \cup (\text{pre}_E(X_i) \cap \{s \mid P \in L(s)\})$

Eddig
címkézettek és ...

... ezek megelőző
állapotai közül amelyek ...

... P-vel
címkézettek

- Bővítés vége: Ha $X_{i+1} = X_i$, azaz nem bővül a halmaz

CTL modellellenőrzés: Összefoglalás

- Globális modellellenőrzés:
 - Állapotok címkézése azokkal a (rész)kifejezésekkel, amelyek igazak az adott állapotban
- Kifejezések felbontása (szintaxis szabályok)
 - Lépések: Atomi kijelentésekből indítva az összetettebb kifejezések felé („belülről kifelé”)
 - Az előző lépésben adott címkézés felhasználása (virtuális „atomi kijelentések”)
- $A(p \cup q)$, $E(p \cup q)$ esetén: Inkrementális állapotcímkézés
 - Kezdőhalmaz:
 - A belső kifejezések (p , q) által meghatározott állapothalmaz alapján
 - Iteráció: A szemantika alapján (megelőző állapotokra lépegetve)
 - Iteráció vége: Nem nő a címkézett állapotok halmaza
 - Precíz matematikai algoritmus: Fixpont számítás teljes hálóknban

CTL modellellenőrzés elméleti alapjai:

Fixpont iteráció

Teljes hálók elmélete

- $KS=(S,R,L)$ Kripke-struktúra
- $(2^S, \subseteq)$ teljes hálót képez, mivel
 - \subseteq reflexív, tranzitív, antiszimmetrikus reláció
 - 2^S tartalmaz felső (S) és alsó (\emptyset) határt
- Legyen: $\tau: 2^S \rightarrow 2^S$ leképezés $\tau(z)$ jelöléssel
 - τ monoton, ha $z_1 \subseteq z_2$ esetén $\tau(z_1) \subseteq \tau(z_2)$
 - τ \cup -folytonos, ha $z_1 \subseteq z_2 \subseteq z_3 \subseteq \dots$ esetén $\tau(\cup_i z_i) = \cup_i \tau(z_i)$
 - τ \cap -folytonos, ha $z_1 \supseteq z_2 \supseteq z_3 \supseteq \dots$ esetén $\tau(\cap_i z_i) = \cap_i \tau(z_i)$
 - Ha S véges, akkor a monotonitásból adódik a \cup -folytonosság és \cap -folytonosság

Fixpontok

- Fixpontok definíciója:
 - $\text{lfp } \tau(z)$ az a legkisebb $z \subseteq S$, amelyre $\tau(z)=z$
 - $\text{gfp } \tau(z)$ az a legnagyobb $z \subseteq S$, amelyre $\tau(z)=z$
- Tételek:
 - Tarski tétele (létezés):
Ha S véges és $\tau(z)$ monoton, akkor létezik legkisebb és legnagyobb fixpont
 - Kleene tételei (számítási mód):
Ha $\tau(z)$ \cup -folytonos, akkor $\text{lfp } \tau(z) = \bigcup_i \tau^i(\emptyset)$.
Ha $\tau(z)$ \cap -folytonos, akkor $\text{gfp } \tau(z) = \bigcap_i \tau^i(S)$.

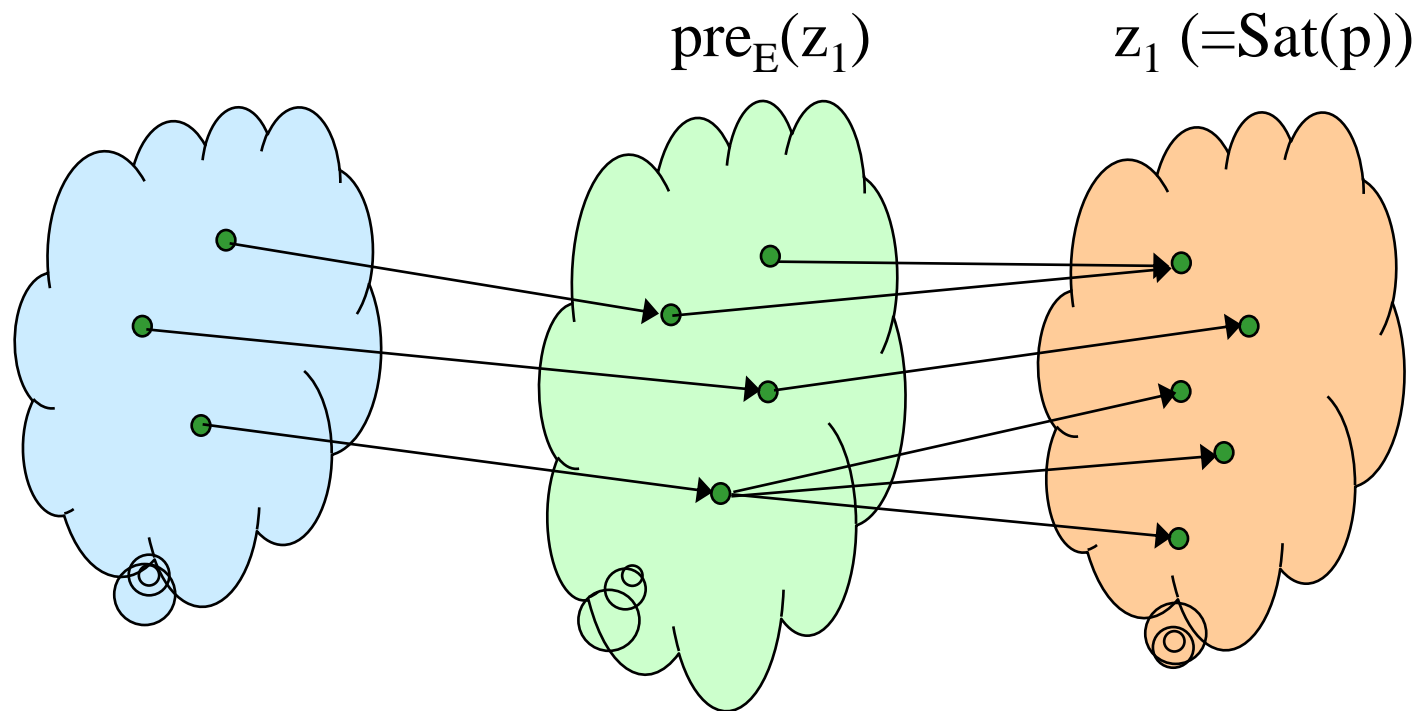
Fixpontok számítása

- A tételek alapján belátható következmény:
 - Véges S és monoton $\tau(z)$ esetén:
 - $\exists i_0: \text{lfp } \tau(z) = \tau^{i_0}(\emptyset),$ Id. $\text{lfp } \tau(z) = \bigcup_i \tau^i(\emptyset)$
 - $\exists j_0: \text{gfp } \tau(z) = \tau^{j_0}(S),$ Id. $\text{gfp } \tau(z) = \bigcap_i \tau^i(S)$
- Iterációval való számítás (korlát az S mérete):
 - $\text{lfp } \tau(z): \quad \emptyset, \tau(\emptyset), \tau^2(\emptyset), \dots$ amíg $\tau^i(\emptyset) = \tau^{i+1}(\emptyset)$
 - Monoton nő a halmaz mérete a fixpontig
 - $\text{gfp } \tau(z): \quad S, \tau(S), \tau^2(S), \dots$ amíg $\tau^j(S) = \tau^{j+1}(S)$
 - Monoton csökken a halmaz mérete a fixpontig

CTL operátorok és fixpont műveletek

- Pongyola megfogalmazásban:
 - **lfp**: eshetőségek (pl. EF); adott tulajdonságot teljesítő állapotokhoz vezető utak kiinduló állapotai
 - **gfp**: folyamatos igazságok (pl. EG): adott tulajdonságot folyamatosan teljesítő utak kiinduló állapotai
- Bizonyítható tétel: $\text{Sat}(\text{EF } p) = \text{lfp } \tau(z)$,
 - ahol $\tau(z) = \text{Sat}(p) \cup \text{pre}_E(z)$ itt analógia: $\text{EF}(p) = p \vee \text{EX EF}(p)$
 - ahol $\text{pre}_E(z) = \{s \mid \exists t: (s,t) \in R \text{ és } t \in z\}$, azaz azoknak az állapotoknak a halmaza, ahonnan van átmenet z-be
 - Itt fixpont számítás: „visszafelé lépked az utakon”; kezdőállapotokat keres a **p**-t kielégítő állapotba vezető utakhoz
 - Kezdetben \emptyset , ebből $\text{Sat}(p)$ lesz az első bővítés
 - Visszalépés az odavezető állapotokon: $\text{pre}_E(z)$ lépked visszafelé

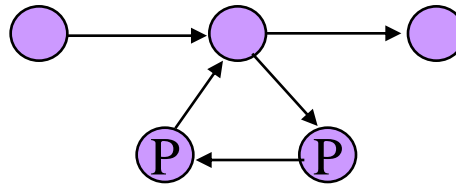
$\text{pre}_E(z)$ számítása



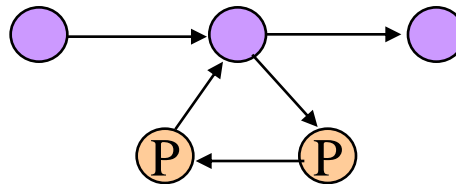
- $\text{Sat}(p)$ lesz az első lépés eredménye
- $\text{pre}_E(z)$ az iteráció során visszafelé lépked az egyes utakon, kezdőállapotokat keresve a $\text{Sat}(p)$ -be vezető utakhoz

EF(P) iterációs példa

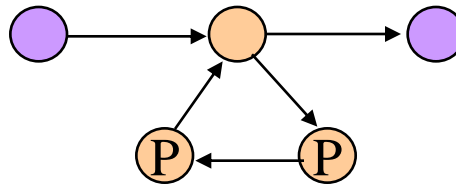
$$z_0 = \emptyset$$



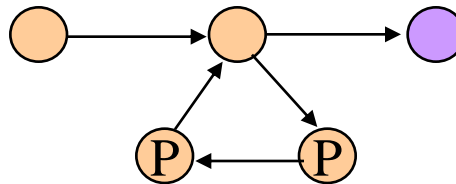
$$z_1 = \text{Sat}(P) \cup \text{pre}_E(z_0) = \text{Sat}(P)$$



$$z_2 = \text{Sat}(P) \cup \text{pre}_E(z_1)$$



$$z_3 = \text{Sat}(P) \cup \text{pre}_E(z_2)$$



CTL operátorok és fixpont műveletek

Tételek (folytatás):

- $\text{Sat}(\text{EG } p) = \text{gfp } \tau(z)$,
 - ahol $\tau(z) = \text{Sat}(p) \cap \text{pre}_E(z)$
analógia: $\text{EG}(p) = p \wedge \text{EX EG}(p)$
 - Itt fixpont számítás: csak az olyan utak kellenek, ahol p minden állapotban igaz (ld. metszet $\text{Sat}(p)$ -vel)
 - Kezdetben S , ebből az elérhető $\text{Sat}(p)$
 - Visszalépés az olyan odavezető állapotokon, ahol igaz p
- $\text{Sat}(E(p \cup q)) = \text{lfp } \tau(z)$,
 - ahol $\tau(z) = \text{Sat}(q) \cup (\text{Sat}(p) \cap \text{pre}_E(z))$
analógia: $E(p \cup q) = q \vee (p \wedge \text{EX } E(p \cup q))$

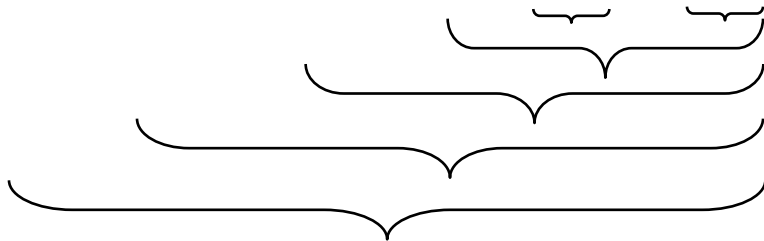
Termináló utakon
sántít az analógia

Összefoglalás: CTL modellellenőrzés iteratívan

- Alapelv: Kifejezések felbontása és „belülről kifelé”

Sat(..) számítások:

$AX\ AG\ (p \wedge E\ (Q\ U\ R))$



- Szabályok: A szintaxis definíció alapján

- $Sat(P) = \{s \mid P \in L(s)\}$
- $Sat(\neg p) = S \setminus Sat(p)$
- $Sat(p \wedge q) = Sat(p) \cap Sat(q)$
 $Sat(p \vee q) = Sat(p) \cup Sat(q)$

- Szabályok (folytatás):

- $\text{Sat}(\text{EF } p)$ számítás: $\text{lfp } \tau(z)$

- ahol $\tau(z) = \text{Sat}(p) \cup \text{pre}_E(z)$
 - ahol $\text{pre}_E(z) = \{s \mid \exists t: (s,t) \in R \text{ és } t \in z\}$

azaz az iteráció:

- $z_0 = \emptyset$
 - $z_1 = \tau(z_0) = \tau(\emptyset)$
 - $z_{i+1} = \tau(z_i) = \text{Sat}(p) \cup \text{pre}_E(z_i) = \text{Sat}(p) \cup \{s \mid \exists t: (s,t) \in R \text{ és } t \in z_i\}$
 - amíg $z_{i+1} = z_i$ és itt $z_i = \text{lfp } \tau(z) = \text{Sat}(\text{EF } p)$

- Szabályok (folytatás):

- $\text{Sat}(\text{EG } p)$ számítás: gfp $\tau(z)$

- ahol $\tau(z) = \text{Sat}(p) \cap \text{pre}_E(z)$
 - ahol $\text{pre}_E(z) = \{s \mid \exists t: (s,t) \in R \text{ és } t \in z\}$

azaz az iteráció:

- $z_0 = S$
 - $z_1 = \tau(z_0) = \tau(S)$
 - $z_{i+1} = \tau(z_i) = \text{Sat}(p) \cap \{s \mid \exists t: (s,t) \in R \text{ és } t \in z_i\}$
 - amíg $z_{i+1} = z_i$ és itt $z_i = \text{gfp } \tau(z) = \text{Sat}(\text{EG } p)$

- $\text{Sat}(E(p \cup q))$ számítás hasonlóan

- További CTL operátorok mint helyettesítések kezelhetők

- A modellellenőrzés halmazműveleteket jelent!

CTL modellellenőrzés komplexitása

- Worst case időkomplexitás: $O(|S|^2 \times |p|)$
 - $|S|^2$ az átmenetek száma worst case esetben (ld. $\text{pre}(z)$)
 - $|p|$ az operátorok száma;
itt a szintaxis szabályainak megfelelően bontjuk fel a kifejezést és külön-külön számítjuk a $\text{Sat}(\cdot)$ halmazt
 - PLTL-nél kedvezőbb (pedig elágazó idejű logika);
 - De itt a PLTL kifejezés soha nem hosszabb, mint az ugyanazt a tulajdonságot leíró CTL kifejezés (ha van ilyen)
- FairCTL esetén: $O(|S|^2 \times |p| \times |q|)$
 - Itt q a fair útvonalak megadására szolgáló kifejezés
- CTL* esetén: $O(|S|^2 \times 2^{|p|})$
 - Itt a beágyazott PLTL kifejezések kötetlensége miatt jön be komplexitásnövekedés
 - Id. PLTL modellellenőrzés komplexitása: $O(|S|^2 \times 2^{|p|})$

Modális mu-kalkulus

Modális mu-kalkulus

- Szintaxis KTS-en:

$$p ::= P \mid Z \mid \neg p \mid p \wedge p \mid [a]p \mid \langle a \rangle p \mid \mu Z.p \mid \nu Z.p$$

- Közvetlenül a fixpont operátorokat tartalmazza
 - $\nu Z.p$ legnagyobb fixpont (Z változó, ahol p nyitott kifejezés Z -re)
 - az a legnagyobb $S^* \subseteq S$ halmaz, melyet újra megkapunk, ha a $p(Z)$ formulát azzal az interpretációval értékeljük ki, hogy Z az S^* -on igaz
 - $\mu Z.p$ legkisebb fixpont (ahol p nyitott kifejezés Z -re)
- Előírás: Z páros számú negáció hatókörében fordulhat elő
 - Így kapunk monoton kifejezéseket, $\text{Sat}(p)$ iterációval számítható
- CTL* is lefedhető vele (kifejezőképessége nagyobb)
- Worst case időkomplexitás: $O(|S|^2 \times |p|^a)$
 - Itt a az egymásba ágyazott **váltakozó** fixpont operátorok száma („alternation depth”)
 - CTL esetén az alternation depth értéke 1 (független fixpont műveletek), általános esetben ennél nagyobb is lehet

A modális mu-kalkulus használata

- CTL: egymásba ágyazott fixpont kifejezések között nincs függőség:
pl. $AG\ EF\ p = \nu Z.(\mu Y.(p \vee EX(Y)) \wedge AX(Z))$
 - Egy belső fixpont kifejezés nem függ egy külső fixpont kifejezés változójától
 - Egy-egy kifejezés kiértékelhető belülről kifelé haladva, egyenként feloldva az egyes operátorokat
- Általános eset:
pl. $\nu Z.\mu Y.(Z \vee <a>Y)$, azaz van **a** és **b** lépésekből álló út, végtelen sokszor előforduló **b**-vel
 - Kölcsönös függőség van a legkisebb és legnagyobb fixpont kifejezések között
 - Az iteráció "ellenkező irányba tart", újraszámolt belső iteráció szükséges a külső iterációs lépésekben