Gyakorló feladatok: Formális modellek, temporális logikák, modellellenőrzés

Majzik István
BME Méréstechnika és Információs Rendszerek Tanszék

Alapszintű formális modellek

Elméleti kérdések

- Definiálja a következő formalizmusokat:
  - Kripke-struktúra (KS)
  - Címkézett tranzíciós rendszer (LTS)
  - Kripke tranzíciós rendszer (KTS)
- Döntse el, hogy igazak-e a következő állítások:
  - Egy KTS modell egy állapota legfeljebb egy atomi kijelentéssel címkézhető.
  - Az LTS modellek esetén egy tranzíció több akcióval is címkézhető.
  - LTS modellek esetén állapot címkék és tranzíció címkék is használhatók.

Elméleti kérdések – Megoldás 1/2

- Definiálja a következő formalizmusokat: KS, LTS, KTS

$KS = (S, R, L)$ és AP, ahol

$\mathrm{AP}=\{\mathrm{P,Q,R,...}\}$ atomi kijelentések halmaza (domén-specifikus)

$\mathrm{S}=\{\mathrm{s}_1,\mathrm{s}_2,\mathrm{s}_3,...\mathrm{s}_\mathrm{n}\}$ állapotok halmaza, s_1 kezdőállapot

$\mathrm{R} \subseteq \mathrm{S}\times\mathrm{S}$: állapotátmeneti reláció

$\mathrm{L}: \mathrm{S} \to 2^{AP}$ állapotok címkézése atomi kijelentésekkel

$LTS = (S, Act, \to)$, ahol

$\mathrm{S}=\{\mathrm{s}_1,\mathrm{s}_2,...\mathrm{s}_\mathrm{n}\}$ állapotok halmaza, s_1 kezdőállapot

$\mathrm{Act}=\{\mathrm{a,b,c,...}\}$ akciók halmaza (domén-specifikus)

$\to \subseteq \mathrm{S}\times\mathrm{Act}\times S$ címkézett állapotátmenetek, pl. $\mathrm{s}_1 \xrightarrow{\mathrm{a}} \mathrm{s}_2$

$KTS = (S, \to, L)$ és AP, Act, ahol

$AP = \{P, Q, R, ...\}$ atomi kijelentések és $Act = \{a, b, c, ...\}$ akciók halmaza

$S = \{s_1, s_2, s_3, ... s_n\}$ állapotok halmaza, s_1 kezdőállapot

$\to \subseteq S \times Act \times S$ állapotátmeneti reláció (akciókkal címkézve)

$L: S \to 2^{AP}$ állapotok címkézése atomi kijelentésekkel

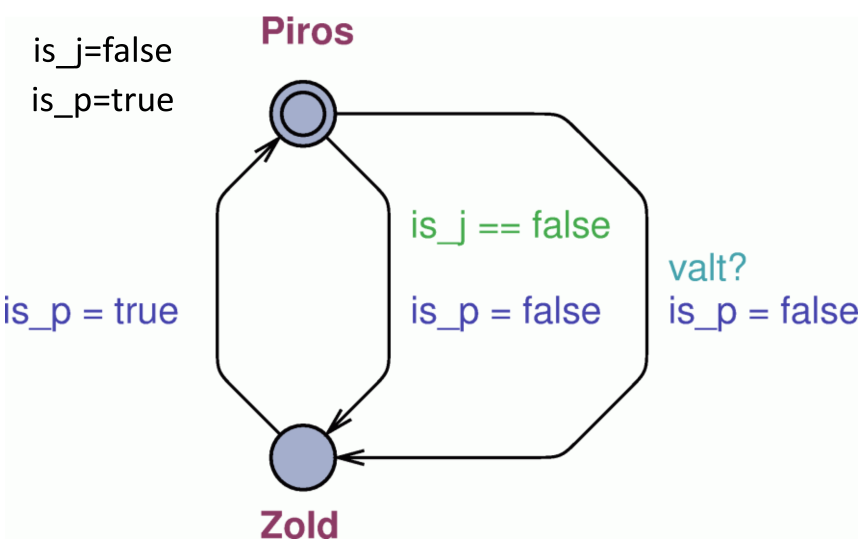
Elméleti kérdések – Megoldás 2/2

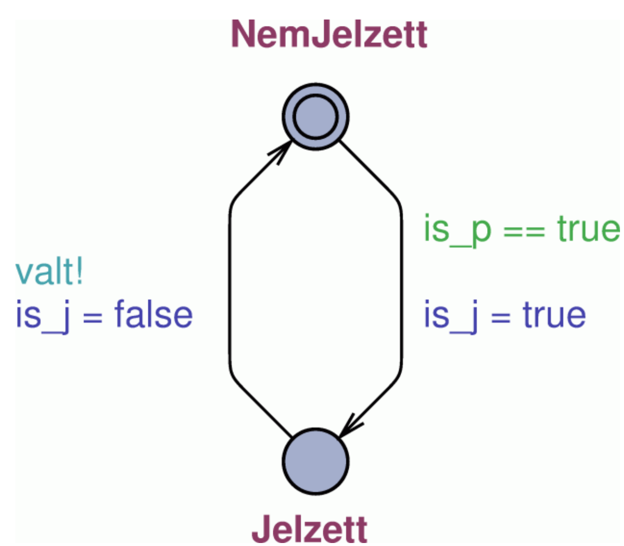
- Döntse el, hogy igazak-e a következő állítások:
  - Egy KTS modell egy állapota legfeljebb egy atomi kijelentéssel címkézhető.
    - Hamis.
  - Az LTS modellek esetén egy tranzíció több akcióval is címkézhető.
    - Hamis.
  - LTS modellek esetén állapot címkék és tranzíció címkék is használhatók.
    - Hamis.

Formális modellek értelmezése

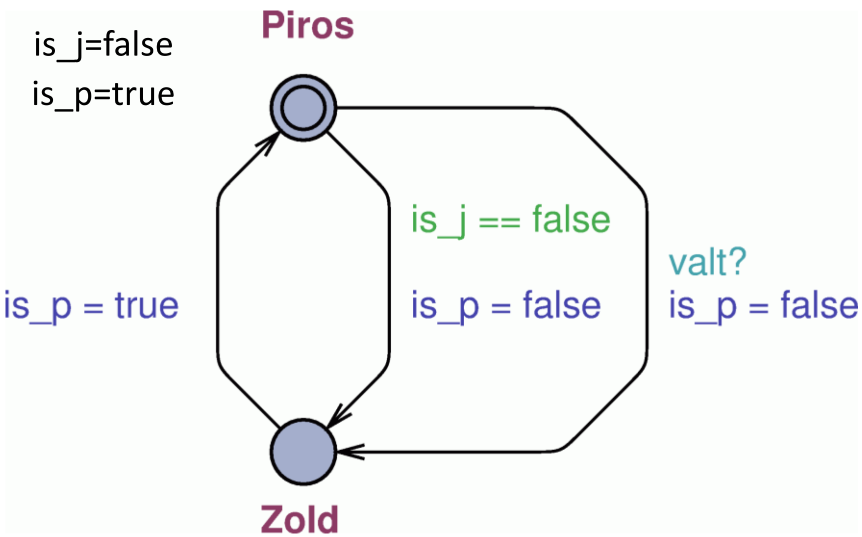
Az alábbi ábrákon látható két (az UPPAAL eszközben felvett) automata, ezek egy jelzőlámpa és egy gyalogos viselkedését modellezik. A kezdeti állapotban is_j=false, is_p=true.

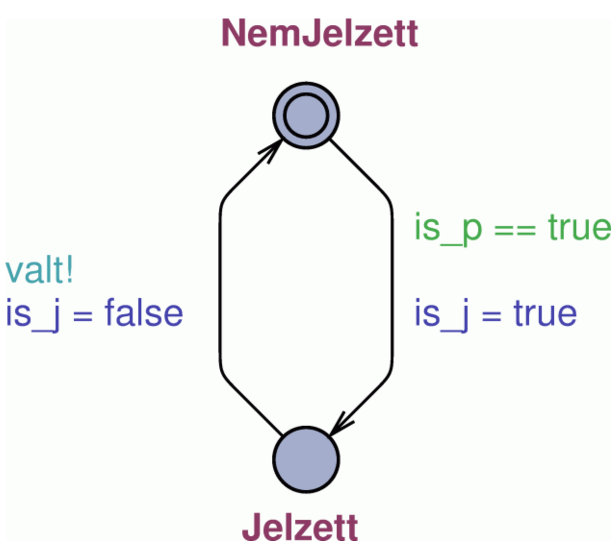
> Készítse el a két automata együtteseként tekintett teljes rendszer Kripke-struktúra modelljét, a jelzőlámpa és a gyalogos elérhető állapotkombinációit és a köztük lévő átmeneteket felvéve. A Kripke-struktúra minden állapotát jelölje meg azzal, hogy a jelzőlámpa és a gyalogos mely állapotait reprezentálja.



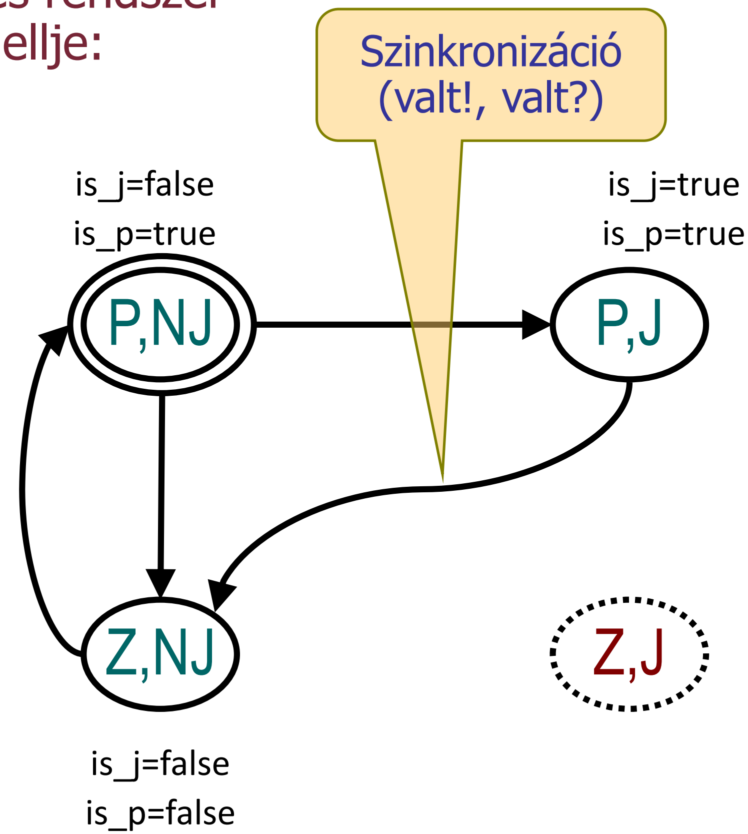


Formális modellek értelmezése – Megoldás





Teljes rendszer modellje:



Követelmények formalizálása temporális logikákkal

Temporális logikai kifejezések értelmezése

Indokolja meg, hogy következő LTL ekvivalenciák helyesek-e:

1. $F\ (Start \vee Stop) \equiv (F\ Start) \vee (F\ Stop)$
2. $G\ Normal \equiv not\ F\ (not\ Normal)$

Indokolja meg, hogy következő CTL ekvivalenciák helyesek-e:

1. $AF\ (Start \vee Stop) \equiv (AF\ Start) \vee (AF\ Stop)$
2. $AF\ (Start \wedge Stop) \equiv (AF\ Start) \wedge (AF\ Stop)$
3. $EF\ (Start \wedge Stop) \equiv (EF\ Start) \wedge (EF\ Stop)$

Indokolja meg, hogy az alábbi kifejezések szintaktikailag helyesek-e CTL illetve CTL* temporális logikában!

1. $A\ (X\ Stop \vee F\ Start)$
2. $A\ (Stop\ U\ (AX\ Start))$

Temporális logikai kifejezések – Megoldás 1/3

Két kifejezés ekvivalens, ha bármely modellen:

- ha a bal oldal teljesül, akkor a jobb oldal is teljesül, és
- ha a jobb oldal teljesül, akkor a bal oldal is teljesül

A következő LTL ekvivalenciák helyesek-e:

1. $F\,(Start \vee Stop) \equiv (F\,Start) \vee (F\,Stop)$

   Helyes: Az operátorok szemantikája alapján a bal oldal teljesítéséből következik a jobb oldal teljesülése, és viszont.

2. $G\,Normal \equiv not\;F\,(not\;Normal)$

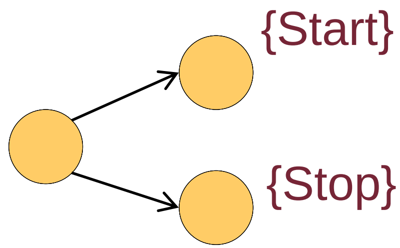
   Helyes: Az operátorok szemantikája alapján a bal oldal teljesítéséből következik a jobb oldal teljesülése, és viszont.

Temporális logikai kifejezések – Megoldás 2/3

Indokolja meg, hogy következő CTL ekvivalenciák helyesek-e:

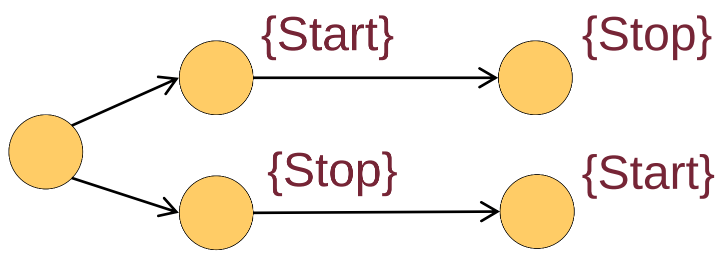
1. AF (Start ∨ Stop) ≡ (AF Start) ∨ (AF Stop)

   Példa modell #1: A bal oldal teljesül, de a jobb oldal nem.



2. AF (Start ∧ Stop) ≡ (AF Start) ∧ (AF Stop)

   Példa modell #2: A jobb oldal teljesül, de a bal oldal nem.



3. EF (Start ∧ Stop) ≡ (EF Start) ∧ (EF Stop)

   Példa modell #1 fentebb: Jobb oldal teljesül, de a bal oldal nem.

Temporális logikai kifejezések – Megoldás 3/3

Indokolja meg, hogy az alábbi kifejezés szintaktikailag helyes-e CTL illetve CTL* temporális logikában!

1. A (X Stop ∨ F Start)

   Szintaktikailag nem helyes CTL-ben, mert a ∨ Boole operátor található az X Stop és az F Start útvonal-kifejezések között
   (ez pedig nem megengedett CTL esetén).

2. A (Stop U (AX Start))

   Szintaktikailag helyes CTL-ben, mert az AU operátor két állapot-kifejezésre van alkalmazva, ezek a Stop és az AX Start.

Követelményformalizálás: Vasúti átjáró

- Egy vasúti átjárót biztosító fénysorompó viselkedését az állapotaihoz rendelt következő atomi kijelentésekkel jellemezzük: {kikapcsolt, fehér, piros}
- Az átjáróhoz érkező autós viselkedését az állapotaihoz rendelt következő atomi kijelentésekkel jellemezzük: {érkezik, körülnéz, megáll, áthalad}
- Formalizálja LTL kifejezések segítségével az alábbi követelményeket, amelyek az autós viselkedésére minden esetben (folyamatosan) vonatkoznak:
  1. Kikapcsolt állapotú fénysorompó esetén az autós körülnéz és a következő időpillanatban vagy áthalad, vagy megáll.
  2. Az autós előbb-utóbb át fog haladni a vasúti átjárón.
  3. Ha egy autós érkezésekor a fénysorompó piros, akkor az autós addig nem halad át, amíg fehérre nem vált a fénysorompó.

Követelményformalizálás: Vasúti átjáró – Megoldás

- A fénysorompó címkéi:

  {kikapcsolt, fehér, piros}
- Az autós címkéi:

  {érkezik, körülnéz, megáll, áthalad}
- A követelményeket formalizáló LTL kifejezések:

  „minden esetben (folyamatosan) vonatkoznak”: kezdeti G kell

  1. Kikapcsolt állapotú fénysorompó esetén az autós körülnéz és a következő időpillanatban vagy áthalad, vagy megáll.

     G (kikapcsolt $\rightarrow$ (körülnéz $\wedge$ X (áthalad $\vee$ megáll)))
  2. Az autós előbb-utóbb át fog haladni a vasúti átjárón.

     G F áthalad
  3. Ha egy autós érkezésekor a fénysorompó piros, akkor az autós addig nem halad át, amíg fehérre nem vált a fénysorompó.

     G ((érkezik $\wedge$ piros) $\rightarrow$ (($\neg$ áthalad) U fehér))

Követelményformalizálás: Szerverterem

- Egy bonyolult szimulációt futtató szerver állapotait a következő atomi kijelentésekkel jellemezzük:
  {kikapcsolt, várakozó, bemelegítés, szimuláció}
- A szerverszoba hűtésének működését a következő atomi kijelentésekkel jellemezzük: {készenlét, normál, maximális}
- Formalizálja LTL kifejezések segítségével az alábbi követelményeket, amelyek a rendszer működésére minden esetben (folyamatosan) vonatkoznak:
  1. Ha egy adott pillanatban a szimuláció a hűtés készenlét állapota mellett zajlik, akkor a következő pillanatban a szerver várakozó állapotra kapcsol.
  2. Előbb-utóbb elkezdhető a szimuláció.
  3. Ha kikapcsolt az állapot, akkor nem hajtható végre szimuláció, amíg bemelegítés nem történik (ami előbb-utóbb megtörténik).

Követelményformalizálás: Szerverterem – Megoldás

- A szerver címkéi:
  {kikapcsolt, várakozó, bemelegítés, szimuláció}
- A hűtés címkéi:
  {készenlét, normál, maximális}
- A követelményeket formalizáló LTL kifejezések :
  „minden esetben (folyamatosan) vonatkoznak”: kezdeti G kell

1. Ha egy adott pillanatban a szimuláció a hűtés készenlét állapota mellett zajlik, akkor a következő pillanatban a szerver várakozó állapotra kapcsol.
   $G\ ((\text{szimuláció} \wedge \text{készenlét}) \rightarrow X\ \text{várakozó})$
2. Előbb-utóbb elkezdhető a szimuláció.
   $G\ F\ \text{szimuláció}$
3. Ha kikapcsolt az állapot, akkor nem hajtható végre szimuláció, amíg bemelegítés nem történik (ami előbb-utóbb megtörténik).
   $G\ (\text{kikapcsolt} \rightarrow ((\neg\ \text{szimuláció})\ U\ \text{bemelegítés}))$

Tulajdonságok ellenőrzése formális modelleken

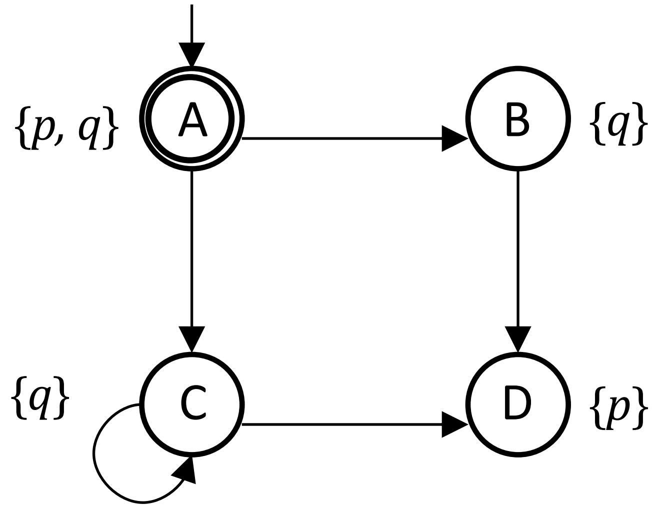
Ellenőrző kérdések: Modellellenőrző algoritmusok

1. Írja le, hogyan azonosíthatók azok az állapotok a modellben, amelyeken igaz az E(P U Q) tulajdonság!
2. Rajzolja fel a tabló felbontás szabályát az LTL temporális logika U operátora esetén!
   Írja le, mikor adódhat ellentmondásos ág az U operátorral felírt kifejezés így megadott felbontásának elvégzése során!
3. Írja le a korlátos modellellenőrzés alapötletét!

CTL tulajdonság ellenőrzése címkézéssel

Adott az alábbi Kripke-struktúra.

- A tanult iteratív állapotcímkézési eljárást végrehajtva ellenőrizze a modellen, hogy teljesül-e a kezdőállapotból az alábbi CTL kifejezés: $A(p \; U \; (EX \; \neg q))$.
- Az iteráció minden lépéséhez adja meg a címkéző kifejezést és (felsorolással) a címkézett állapotok halmazát!



CTL tulajdonság ellenőrzése címkézéssel – Háttér

Tudnivalók:

- Felhasznált séma: $A(p \mathbin{U} q) = q \vee (p \wedge AX\, A(p \mathbin{U} q))$
- Iteratív címkézési algoritmus:
  - Első lépés: A q-val már címkézett állapotok adják azokat az állapotokat, amelyekre először ráteheto az A(p U q) címke.
  - További lépések: Ha szerepel egy állapoton a p címke, és minden rákövetkező állapoton szerepel az A(p U q) címke, akkor erre az állapotra is rátehető az A(p U q) címke.

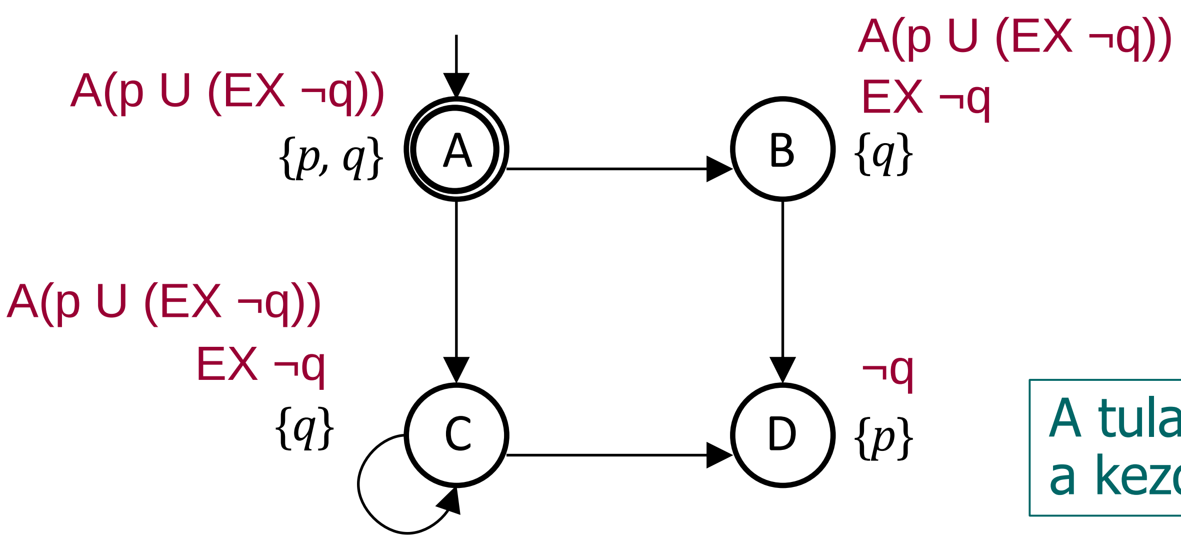
  Célszerű: Az újonnan A(p U q)-val címkézett állapotok megelőző állapotait végignézni a szabály alkalmazására.

CTL tulajdonság ellenőrzése címkézéssel – Megoldás

Ellenőrizze a modellen, hogy teljesül-e a kezdőállapotból az alábbi CTL kifejezés: A(p U (EX ¬q))

1. A ¬q címke feltehető: D
2. Az EX ¬q címke feltehető: B, C
3. Az A(p U (EX ¬q)) címke először feltehető: B, C
4. Az A(p U (EX ¬q)) címke ezután feltehető: A

Az iteráció vége, nincs több címkézhető állapot



A tulajdonság igaz a kezdőállapotra.

Modellellenőrzés: Szerverek

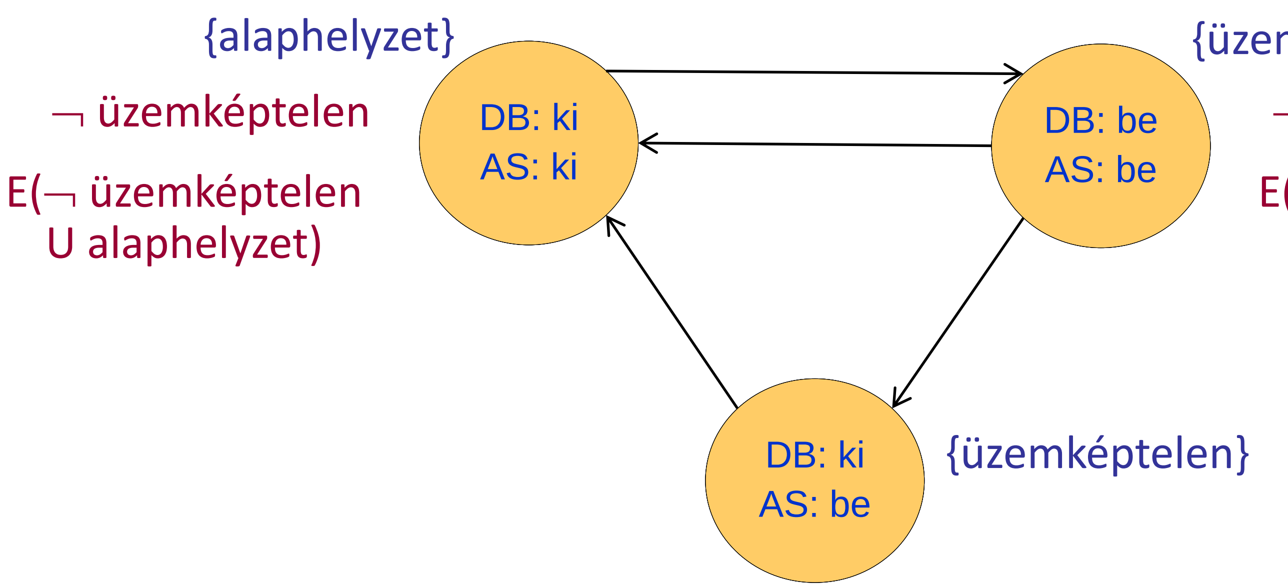
- Egy informatikai rendszer egy adatbázisszerverből és egy alkalmazásszerverből áll, amelyek kikapcsoltak vagy bekapcsoltak lehetnek. Alaphelyzetben mindkét szerver ki van kapcsolva.
- A szervereket hibamentes esetben egyszerre kapcsolják ki/be.
- Az üzemállapot az, amikor mindkét szerver be van kapcsolva.
- Ha az üzemállapotban az adatbázisszervert hiba következtében kikapcsolják, az rendszerszinten üzemképtelen állapotnak tekinthető. Ezután az alkalmazásszervert is kikapcsolják, majd mindkét szerver bekapcsolásával indítják újra a rendszert.
  - Rajzolja fel a rendszer itt leírt működését modellező Kripke-struktúrát az egyes szerverek bekapcsolását és kikapcsolását figyelembe véve! Az egyes állapotokat jellemezze a következő atomi kijelentésekkel:

    {alaphelyzet, üzemállapot, üzemképtelen}
  - Ellenőrizze a modellen, hogy az üzemállapotból tekintve teljesül-e a következő CTL kifejezés:

    E(¬üzemképtelen U alaphelyzet)

Modellellenőrzés: Szerverek – Megoldás

- Egy informatikai rendszer egy adatbázisszerverből és egy alkalmazásszerverből áll, amelyek kikapcsolt vagy bekapcsolt állapotban lehetnek. Alaphelyzetben mindkét szerver ki van kapcsolva.
- A szervereket hibamentes esetben egyszerre kapcsolják ki/be.
- Az üzemállapot az, amikor mindkét szerver be van kapcsolva.
- Ha az üzemállapotban az adatbázisszervert hiba következtében kikapcsolják, az rendszerszinten üzemképtelen állapotnak tekinthető. Ezután az alkalmazásszervert is kikapcsolják, majd mindkét szerver bekapcsolásával indítják újra a rendszert.
  1. Rajzolja fel a rendszer itt leírt működését modellező Kripke-struktúrát.
  2. Ellenőrizze az üzemállapotból tekintve: E(¬üzemképtelen U alaphelyzet)



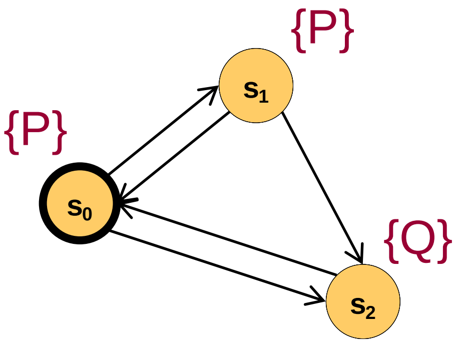
A CTL kifejezés igaz az üzemállapot esetén.

Modellellenőrzés tabló módszerrel

Adott a rajzon látható Kripke struktúra.

Végezzük el a következő kifejezés ellenőrzését a tabló módszert alkalmazva:

$$\neg (P \; U \; Q)$$

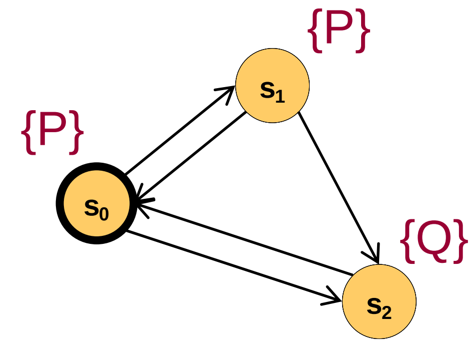


Modellellenőrzés tabló módszerrel – Háttér 1/2

Adott a rajzon látható Kripke struktúra.

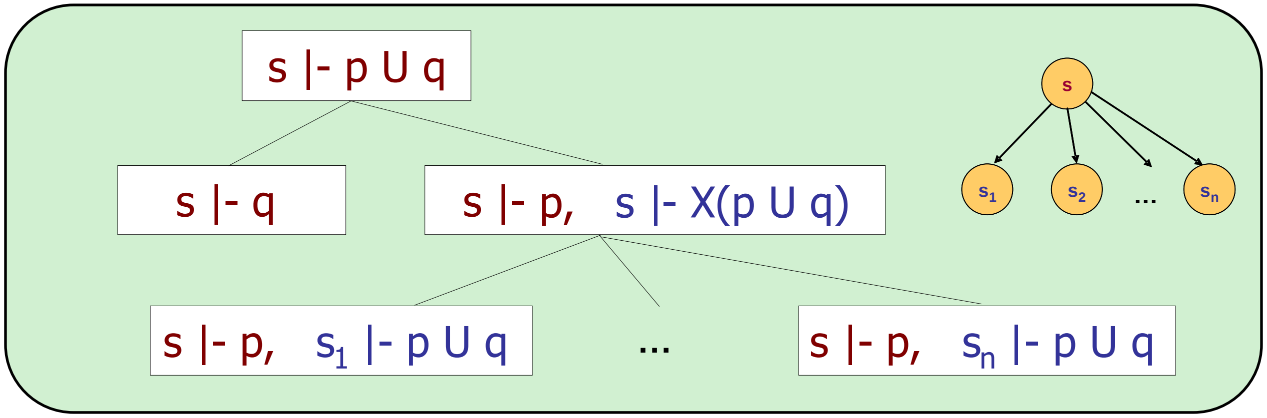
Végezzük el a következő kifejezés ellenőrzését a tabló módszert alkalmazva:

$$\neg\,(P \ U \ Q)$$



Tudnivalók:

- Negált kifejezés (ellenpélda kereséshez): (P U Q)
- Tabló építés szabálya: $(p \ U \ q) = q \vee (p \wedge X(p \ U \ q))$

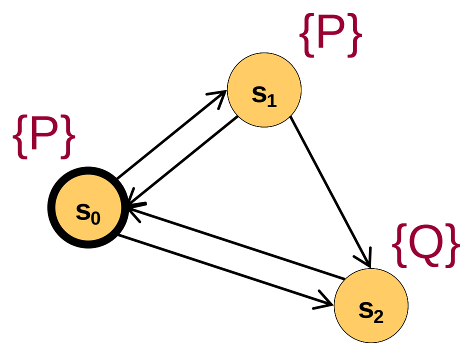


Modellellenőrzés tabló módszerrel – Háttér 2/2

Adott a rajzon látható Kripke struktúra.

Végezzük el a következő kifejezés ellenőrzését a tabló módszert alkalmazva:

$\neg$ (P U Q)

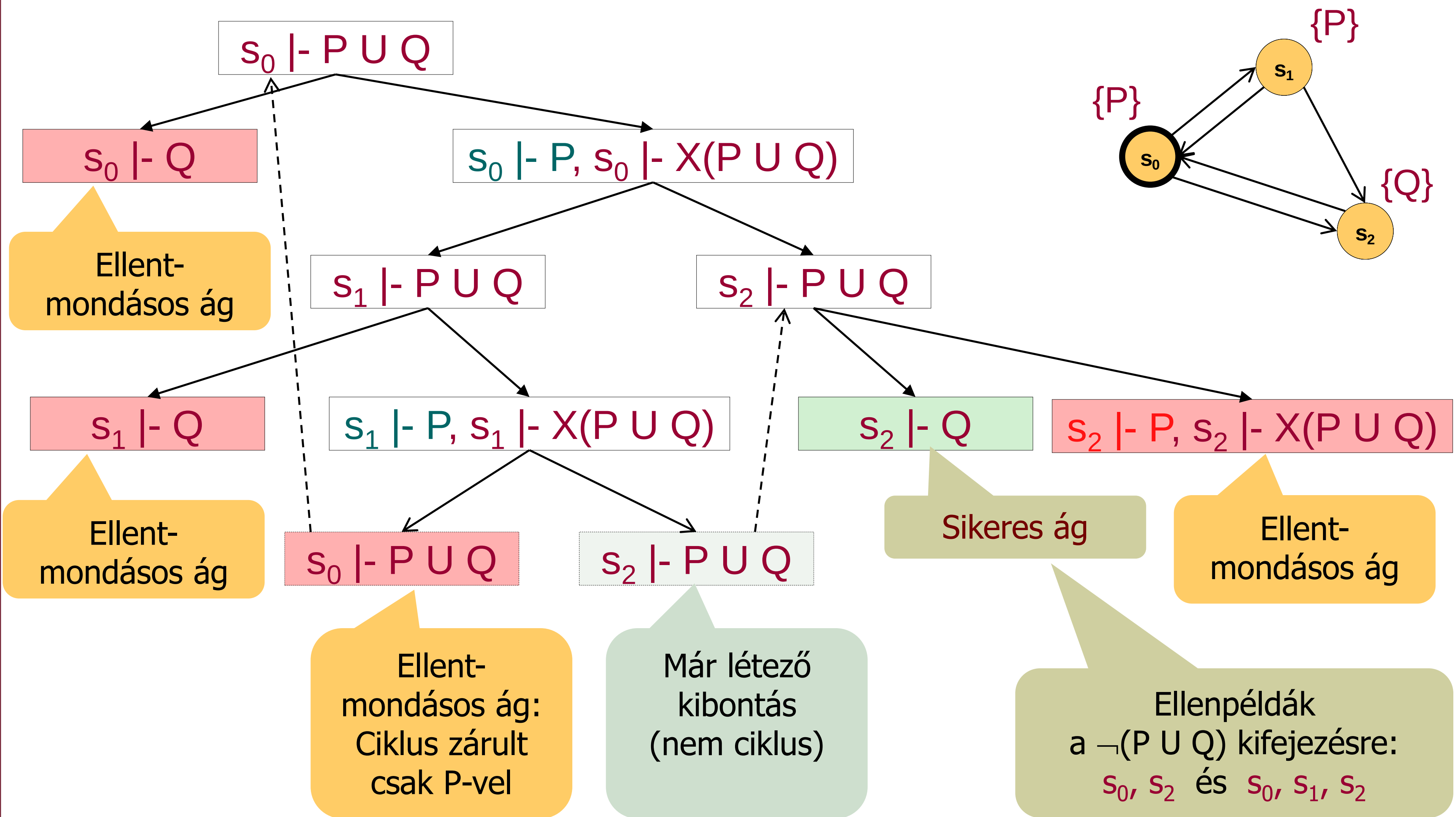


Tudnivalók:

- Negált kifejezés (ellenpélda kereséshez): (P U Q)
- Tabló építés szabálya: (p U q) = q $\vee$ (p $\wedge$ X(p U q))
- A tabló építésben ellentmondásra jutunk:
  - Atomi kijelentésre vonatkozó lokális állítás nem teljesül
  - X operátor van, de az útvonal véget ér Q teljesülése nélkül
  - Ciklus alakul ki P teljesülésével, de Q teljesülése nélkül
- A tabló sikeres ágai (itt ellenpéldát adnak):
  - Atomi kijelentésekre vonatkozó állítások listája teljesül
  - Ciklus alakul ki ellentmondás nélkül

Modellellenőrzés tabló módszerrel - Megoldás

Tabló építés: A kifejezés negáltja (P U Q)



Állapotterek reprezentációja

ROBDD kézi összeállítása

Adott az f logikai függvény igazságtáblázata:

x	y	z	f(x,y,z)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

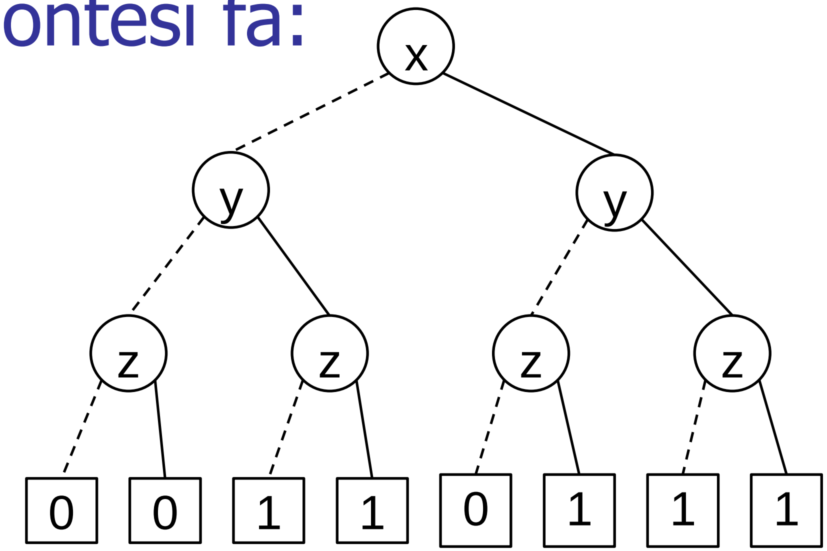
1. Rajzolja fel az f logikai függvény döntési fáját!
   A rajzoláshoz az x, y, z változósorrendet használja.
2. Ez alapján adja meg az f függvényt redukált rendezett bináris döntési diagram (ROBDD) alakban!
3. Adja meg a függvényt algebrai (képlet) alakban!

ROBDD kézi összeállítása – Megoldás 1/3

Adott az f logikai függvény igazságtáblázata:

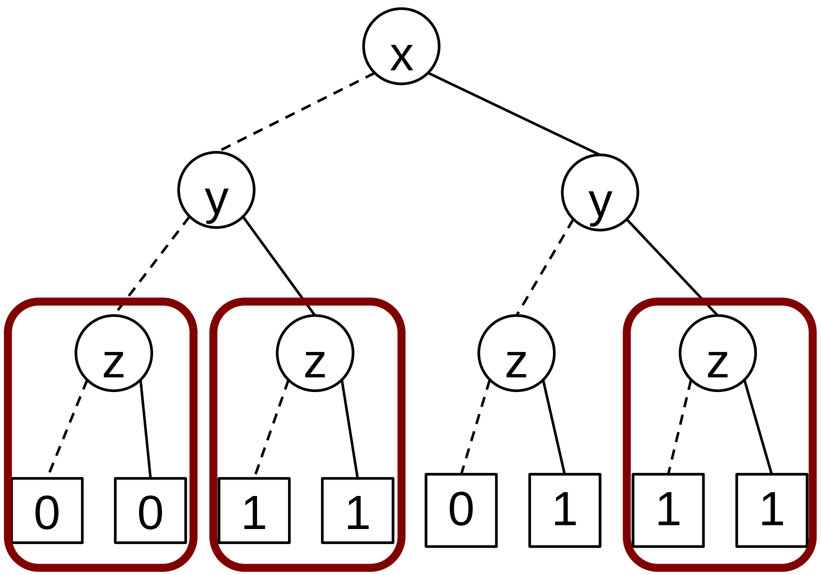
x	y	z	f(x,y,z)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

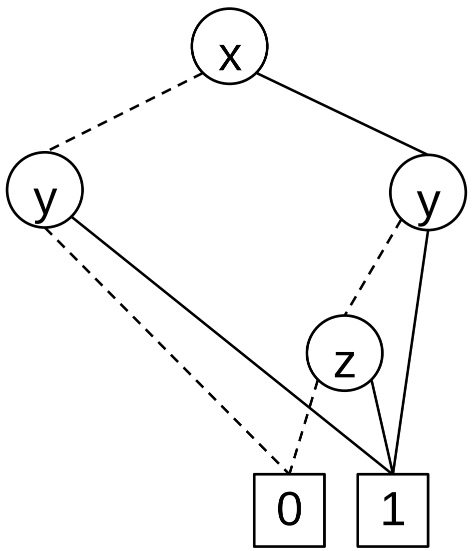
1. Bináris döntési fa:



ROBDD kézi összeállítása – Megoldás 2/3

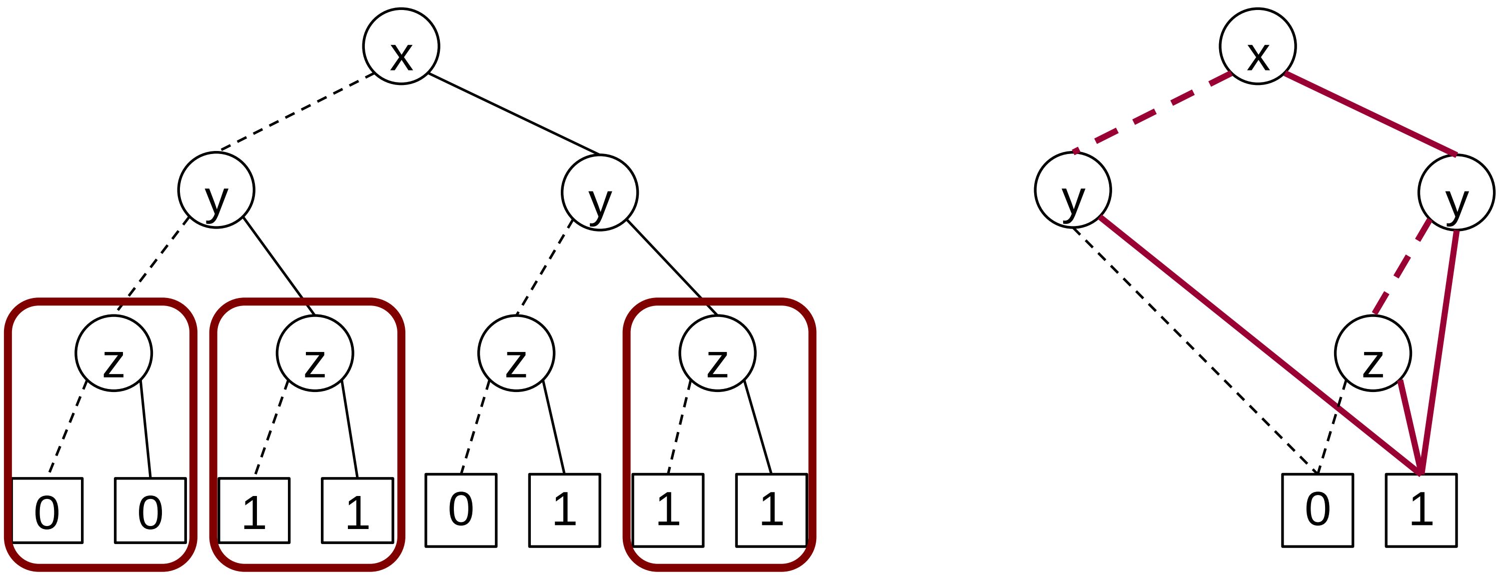
2. Az ROBDD meghatározása az f függvényhez:





ROBDD kézi összeállítása – Megoldás 3/3

2. Az ROBDD meghatározása az f függvényhez:

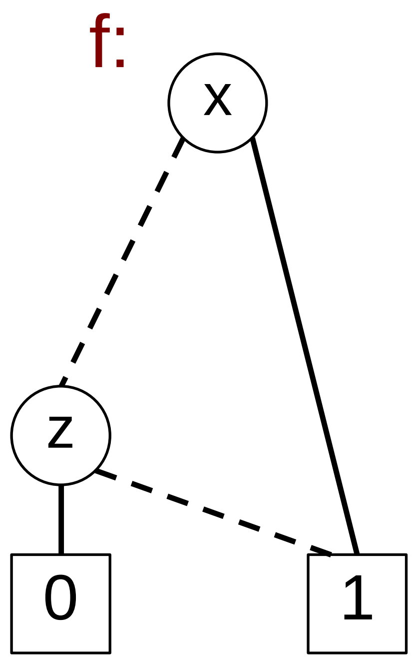


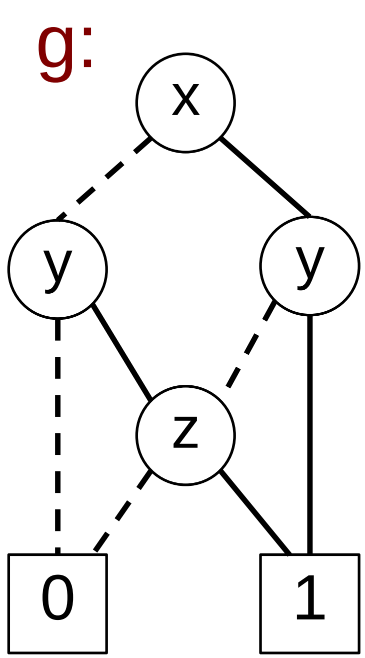
3. Algebrai alak: Az 1 csomóponthoz vezető utak alapján

$$f = (\neg x \wedge y) \vee (x \wedge \neg y \wedge z) \vee (x \wedge y)$$

ROBDD alapú műveletek függvényeken

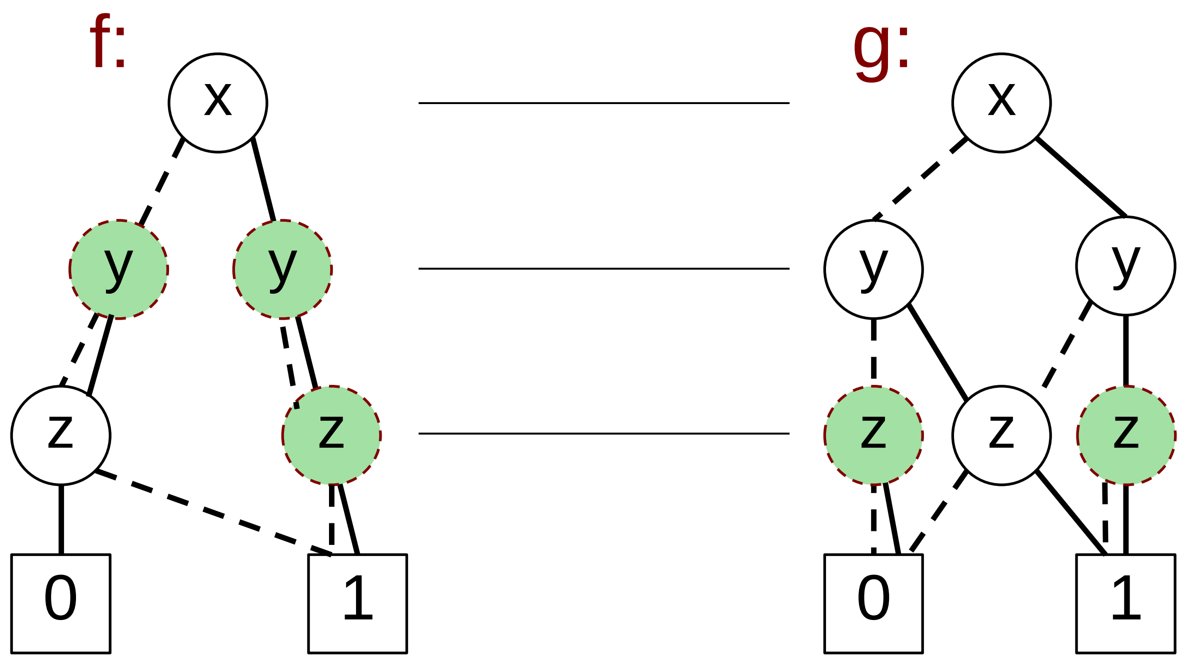
Tekintse az alábbi, ROBDD alakban megadott f és g függvényeket, és rajzolja fel ezek alapján ROBDD alakban az f∧g függvényt!





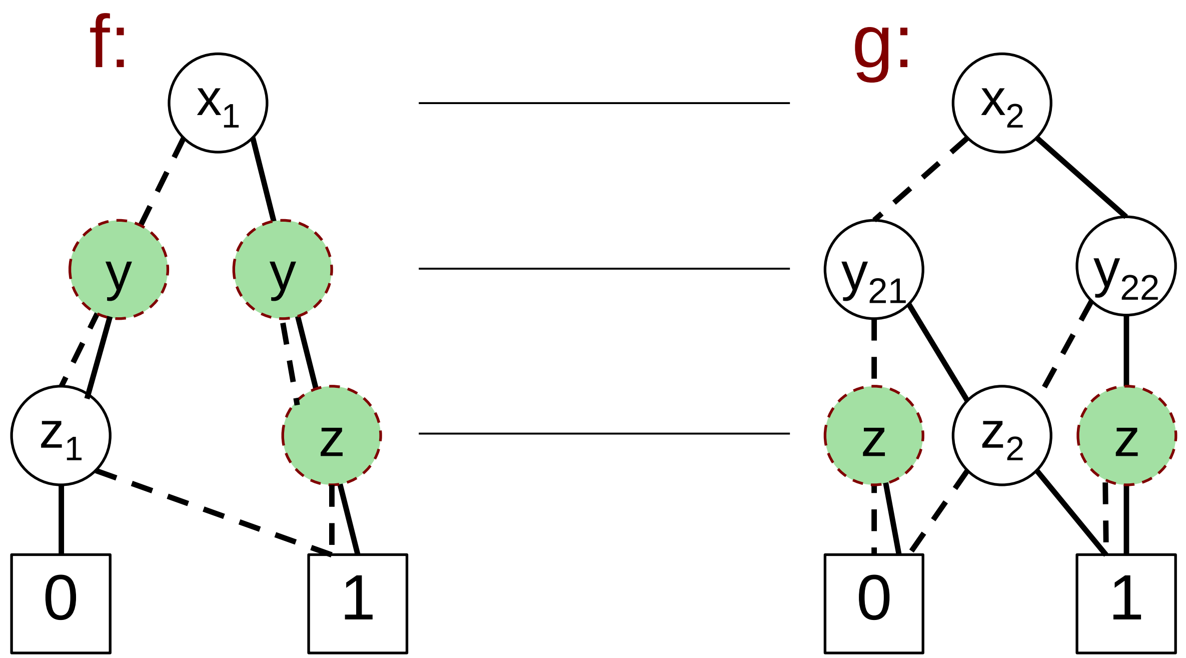
ROBDD alapú műveletek függvényeken – Megoldás

A redukált csomópontok feltüntetése:



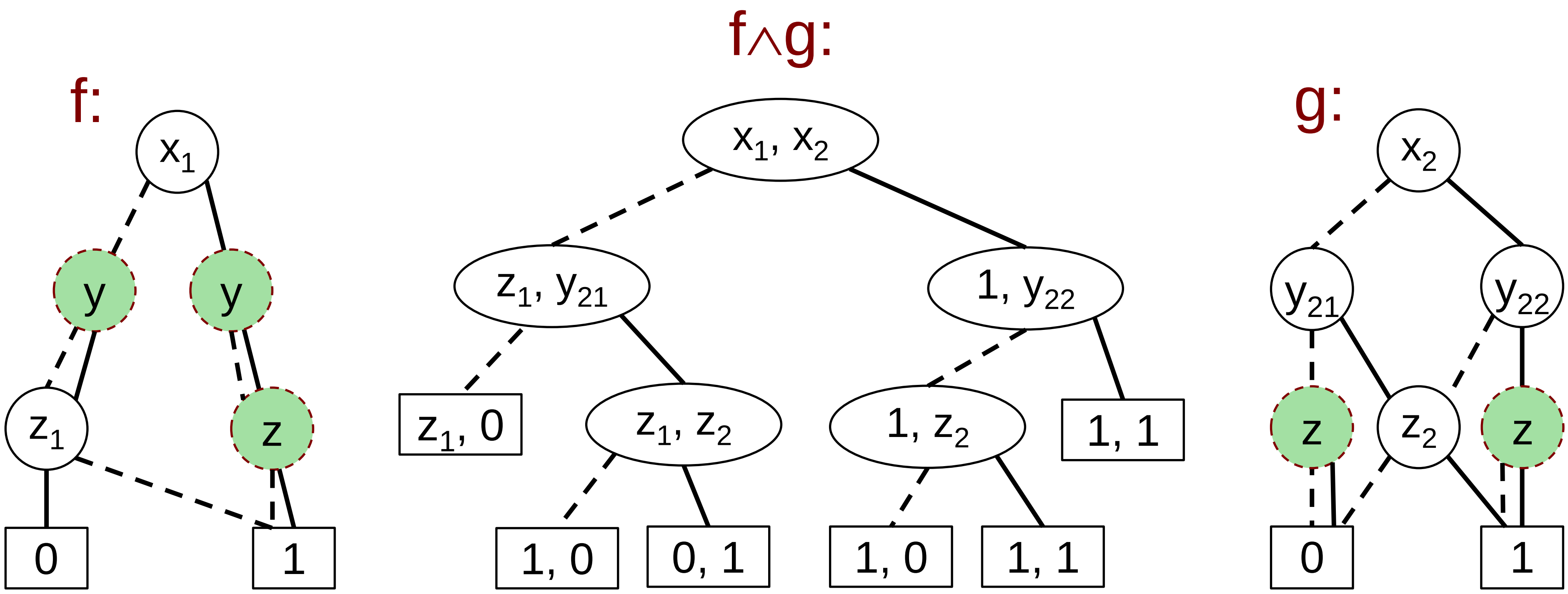
ROBDD alapú műveletek függvényeken – Megoldás

A hivatkozandó csomópontok azonosítása:



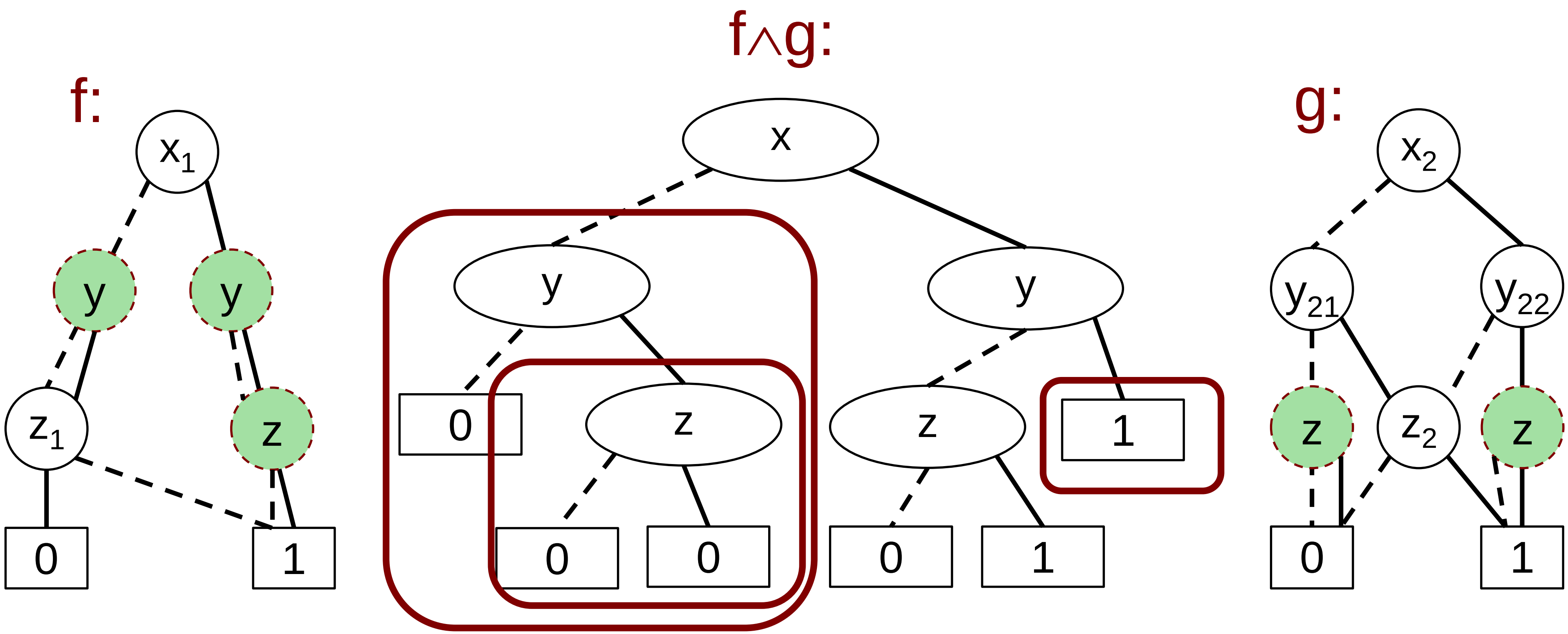
ROBDD alapú műveletek függvényeken – Megoldás

Az f∧g függvény ROBDD-jének konstrukciója:
A csomópontok összeállítása az igaz és hamis ágakon.



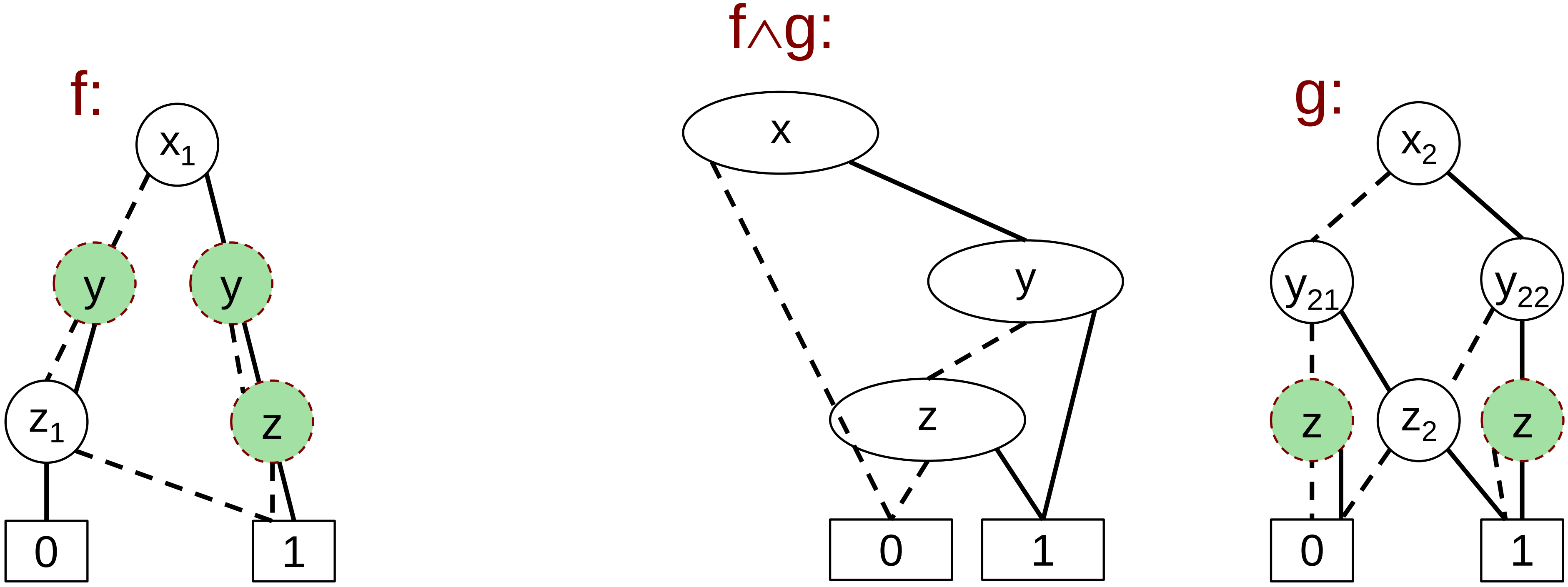
ROBDD alapú műveletek függvényeken – Megoldás

Az f∧g függvény ROBDD-jének konstrukciója: Terminális csomópontok és izomorf részfák azonosítása



ROBDD alapú műveletek függvényeken – Megoldás

Az $f \wedge g$ függvény ROBDD-jének konstrukciója:
A redukálás és összevonás elvégzése.



ROBDD alapú műveletek függvényeken – Megoldás

Az $f \wedge g$ függvény ROBDD-jének konstrukciója:
Az eredmény szebb alakban.