

# Formális módszerek BMEVIMIMA26

## Első zárthelyi: Gyakorló feladatok megoldással

### 1. Elméleti kérdések

- 1.1. Írja le az alábbi állításokról, hogy *igaz*, *hamis*, vagy *nem eldönthető* (az, hogy igaz vagy hamis). (3 pont)
- A. Kripke tranzíciós rendszer (KTS) modellek esetén egy állapothoz csak egy állapotcímke és egy állapotátmenethez csak egy akció tartozhat.
  - B. A korlátos modellellenőrzés nem alkalmazható olyan modellek esetén, amelyekben van ciklus.
  - C. Egy logikai függvényhez tartozó ROBDD mérete mindig független attól, hogy milyen a változók sorrendje az ROBDD-ben.
- 1.2. Rajzolja fel azt a *lehető legkevesebb* címkézett állapotot tartalmazó állapotsorozatot, amin teljesül az **XX P** és az **X (P U (Q ∧ P))** temporális logikai tulajdonság, de nem teljesül a **G P** tulajdonság. (3 pont)
- 1.3. Írjon fel egy olyan temporális logikai kifejezést, ami szintaktikailag nem helyes CTL kifejezés, de helyes CTL\* kifejezés. (1 pont)

### Megoldás:

#### 1.1

- A: Hamis. KTS-ben több címke is tartozhat egy állapothoz.  
B: Hamis. Alkalmazható ciklust tartalmazó modellekre is, de az algoritmusában figyelni kell rá (pl. ciklusmentes útvonalakat kezelni).  
C: Hamis. Az ROBDD mérete függhet a változók sorrendezésétől.

#### 1.2:



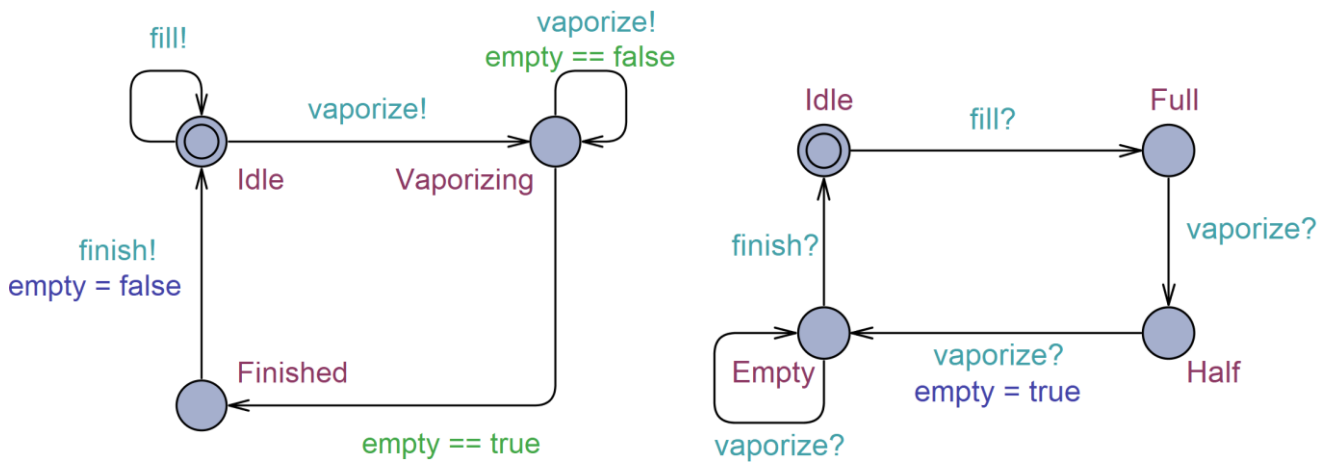
#### 1.3:

- Sokféle ilyen kifejezés lehetséges.  
Pl. CTL-ben útvonal kifejezések nem kombinálhatók (ezeket közvetlenül útvonal kvantornak kell megelőznie), de CTL\*-ban kombinálhatók.

## 2. Modellezés

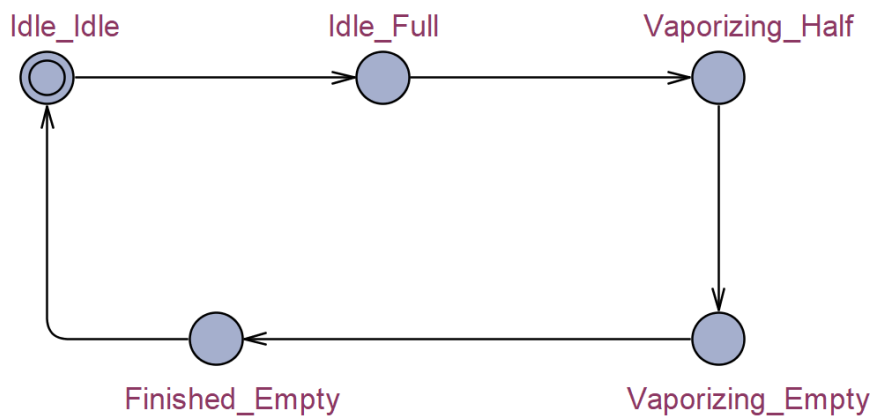
Az alábbi ábrákon látható két (az UPPAAL eszközben modellezett) automata, ezek egy levegő párasító berendezés vezérlőjének állapotait (*Idle*, *Vaporizing* és *Finished*), és a párasító állapotait (*Idle*, *Full*, *Half* és *Empty*) modellezik. Az automaták egy logikai változót (*bool empty*), és három csatornát (*chan fill*, *chan vaporize* és *chan finish*) használnak. A logikai változó kezdetben hamisra van állítva. Figyeljen arra, hogy őrfeltételekben „=” szerepel, míg értékadásokban „=”.

**2.1.** Készítse el a két automata együtteseként tekintett *teljes rendszer* Kripke-struktúra modelljét, a vezérlő és a párasító lehetséges állapotkombinációit és a köztük lévő átmeneteket felvéve. A Kripke-struktúra minden állapotát címkézzze meg azzal, hogy a vezérlő és a párasító mely állapotait reprezentálja (a címkékben használhatja az állapotok neveinek kezdőbetűit). (5 pont)



### Megoldás:

**2.1:**



Note that there is no loop edge here as the empty variable has been set to false.

### 3. Követelmények formalizálása

Egy videokonferencia alkalmazás QVGA, VGA és SVGA felbontású videó átvitelét támogatja (a felbontások ebben a sorrendben nőnek). A hálózat terhelése lehet *alacsony* vagy *magas*, és előfordulhat, hogy a videó *akadozik*. Ezeket az információkat percenként rögzítjük. Formalizálja LTL operátorok és az előbbieken szereplő dőlt betűs atomi kijelentések segítségével az alábbi három követelményt, amelyek a rendszer viselkedésére minden esetben (folyamatosan) vonatkoznak:

- 3.1. Ha a terhelés alacsony, és a videó nem akadozik, akkor a következő percben átváltunk QVGA felbontásról VGA felbontásra, majd a rákövetkező percben SVGA felbontásra. (2 pont)
- 3.2. A videó egészen addig akadozik, amíg a terhelés magas marad. (2 pont)
- 3.3. Ha az SVGA vagy VGA felbontású videó akadozik, akkor előbb-utóbb átváltunk egy alacsonyabb felbontásra (SVGA-ról VGA-ra vagy QVGA-ra, illetve VGA-ról QVGA-ra). (2 pont)

#### Megoldás:

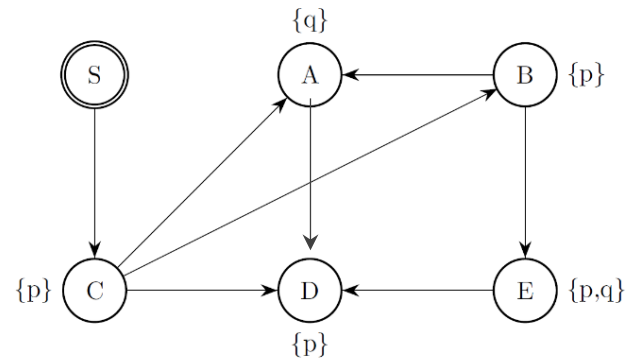
3.1:  $G ( (alacsony \wedge \neg akadozik \wedge QVGA) \rightarrow (X VGA \wedge XX SVGA) )$

3.2:  $G ( akadozik U (\neg magas) )$

3.3:  $G ( ((akadozik \wedge SVGA) \rightarrow F (VGA \vee QVGA)) \wedge ((akadozik \wedge VGA) \rightarrow F QVGA) )$

#### 4. CTL modellellenőrzés

Adott a jobb oldali Kripke-struktúra az S kezdőállapottal és a megadott állapotcímkekkel.



- 4.1.** A tanult iteratív állapotcímkezési eljárást végrehajtva ellenőrizze a modellen, hogy teljesül-e a kezdőállapotból az alábbi CTL kifejezés: **A ((¬p) U (EX q))**. Az iteráció minden lépéséhez adja meg a címkéző kifejezést és (felsorolással) a címkézett állapotok halmazát. (6 pont)

#### Megoldás:

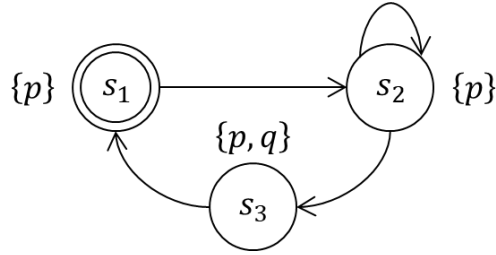
##### 4.1:

1. lépés: S, A állapotokra: **¬p** címke.
2. lépés: B, C állapotokra: **EX q** címke (van olyan közvetlen rákövetkezője, amin van **q** címke).
3. lépés: B, C állapotokra: **A ((¬p) U (EX q))** címke (ahol már van **EX q** címke).
4. lépés: S állapotra: **A ((¬p) U (EX q))** címke (ahol már van **¬p** címke és minden közvetlen rákövetkezőjén már van **A ((¬p) U (EX q))** címke). Az iterációnak vége.

A kifejezés teljesül a kezdőállapotban, mivel az S állapotban ott van az **A ((¬p) U (EX q))** címke.

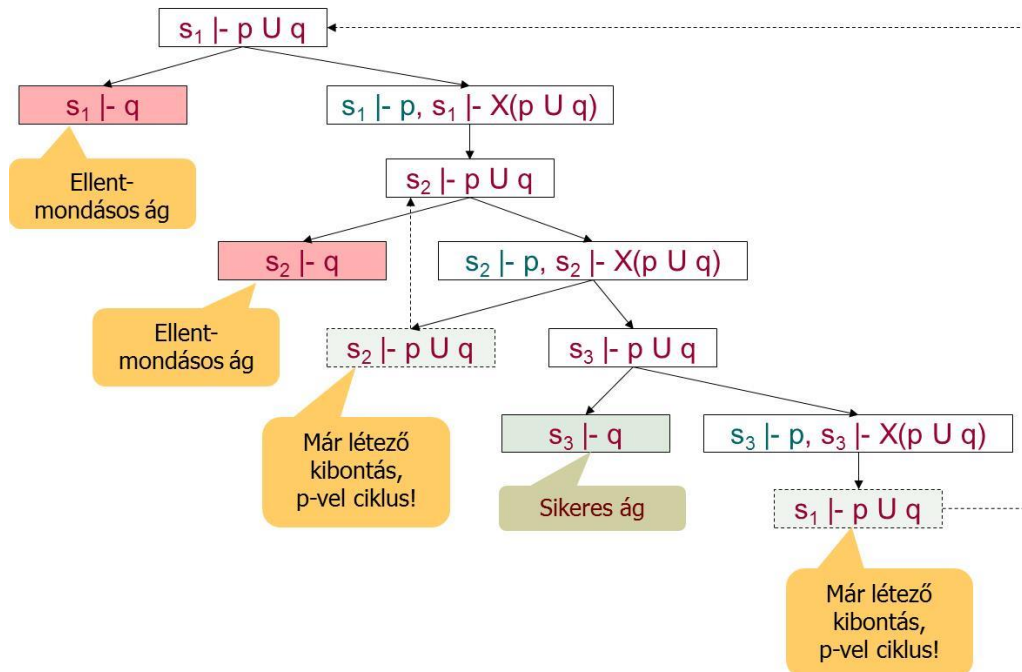
## 5. LTL modellellenőrzés

5.1. Tabló módszerrel ellenőrizze a lenti Kripke-struktúrán (amelynek kezdőállapota  $s_1$ ) a  $\neg(p \text{ U } q)$  követelmény teljesülését! Ha a követelmény nem teljesül, adjon meg a *tabló alapján* egy ellenpéldát is. (6 pont)



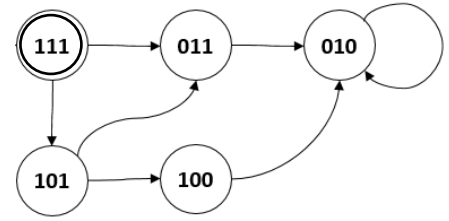
### Megoldás:

5.1: A  $p \text{ U } q$  kifejezés tablójának építése szükséges  $s_1$  állapotból. Ellenpélda a sikeres ágon:  $s_1, s_2, s_3$



## 6. Bináris döntési diagramok

Adott a jobb oldali ábrán látható Kripke-struktúra, melynek állapotai 3 biten, sorban az  $x, y, z$  változók segítségével vannak kódolva (tehát például az 111 kódolású kezdőállapot esetén  $x=1, y=1, z=1$ ).



- 6.1. Adja meg a Kripke-struktúra *kezdőállapotának*, valamint a kezdőállapotból induló  $111 \rightarrow 101 \rightarrow 011$  *útvonalának* karakterisztikus függvényét! (2 pont)
- 6.2. Ábrázolja a Kripke-struktúra *állapottainak halmazát* reprezentáló karakterisztikus függvényt ROBDD alakban! A változók sorrendezése legyen  $x, y, z$ ! (3 pont)

### Megoldás:

6.1:

$$C_{111} = x \wedge y \wedge z$$

$$C_{111 \rightarrow 101 \rightarrow 011} = (x \wedge y \wedge z) \wedge (x' \wedge \neg y' \wedge z') \wedge (\neg x'' \wedge y'' \wedge z'')$$

6.2:

Az ROBDD (megkapható a bináris döntési fa felírásával, majd az azonos részfák összevonásával és a redundáns csomópontok redukálásával):

