

# Időfüggő viselkedés modellezése és verifikációja

Majzik István

Budapesti Műszaki és Gazdaságtudományi Egyetem

Méréstechnika és Információs Rendszerek Tanszék

<http://www.mit.bme.hu/~majzik/>

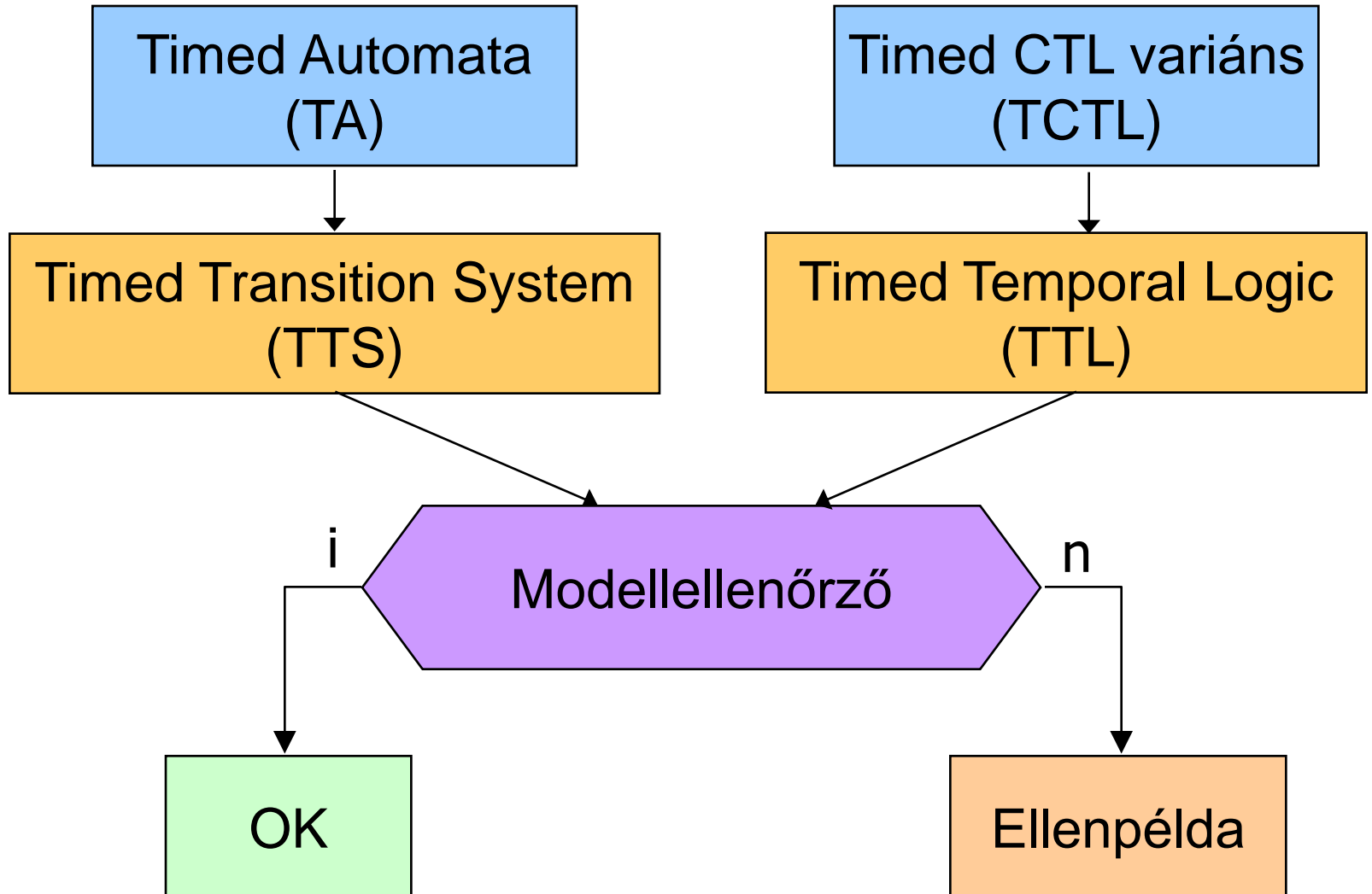
# Valós idejű vezérlők modellezése

- **Állapot-alapú, eseményvezérelt viselkedés**
  - Véges automata alapú modell
- **Időfüggő viselkedés**
  - **Idő telik** az állapotokban
  - **Feltételek** az eltelt idő függvényében
  - Megvalósítás: Fizikai óra, relatív **időmérés** (reseteléssel)
- **Tipikus ellenőrizendő követelmények**
  - **Határidő követelmény teljesítése:**  
Adott időn belül adott tulajdonságú állapotba kerüljön
    - Pl. kérésre adott időn belül válasz érkezen
  - **Biztonsági követelmények teljesítése:**  
Minden (adott időn belül elérhető) állapotban igaz legyen egy adott tulajdonság
    - Pl. adott misszió időtartama alatt biztonságos viselkedés

# Megközelítés

- Mérnöki modell → Alapszintű formális modell
  - Az alapszintű formális modell mint a mérnöki modell „szemantikája” (leképezés a formalizáláshoz)
  - Modellellenőrzés végrehajtása a formális modellen
- Minta:
  - UML állapottérkép → Kripke-struktúra (KS)
  - CTL követelmények
- Az időfüggő viselkedést leíró modell ellenőrzéséhez:
  - Timed Automata (TA) → Timed Transition System (TTS)
  - Timed CTL (TCTL) variáns → Timed Temporal Logic (TL)

# Modellellenőrzés



# Alapszintű modell: Timed Transition System (TTS)

- Jelölések (állapotok és átmenetek tulajdonságai):
  - Atomi kijelentések:  $AP = \{P, Q, \dots\}$
  - Atomi akciók:  $Act = \{a, b, c, \dots\}$
  - Késleltetés akciók:  $\Delta = \{\varepsilon(d) \mid d \in R_{\geq 0}\}$
- TTS definíció:  $TTS = (S, s_0, \rightarrow, V)$  ahol
  - $S$  állapotok halmaza
  - $s_0$  kezdőállapot
  - $\rightarrow \subseteq S \times L \times S$ , ahol  $L \in Act \cup \Delta$  (késleltetés is mint címke)
  - $V: S \rightarrow 2^{AP}$  állapotok címkézése



# Mérnöki modell: Timed Automata (TA)

- Automata + órák
  - Konkurens (rendszer)órák
  - Azonos ütemben (rátával) haladnak
  - Értékük hozzáférhető feltételekben, invariánsokban
  - Resetelhetőek akciókban, egymástól függetlenül
- Jelölések:
  - $C = \{x, y, z, \dots\}$  órák
  - $B(C)$  óra-kifejezések, egy óra-kifejezés  $g$ -vel jelölve
    - Szintaxis:  $g ::= x \sim n \mid x - y \sim n \mid g \wedge g$   
ahol  $\sim \in \{\leq, \geq, =, <, >\}$ ,  
és  $n$  természetes szám (konstans)

# TA definíció

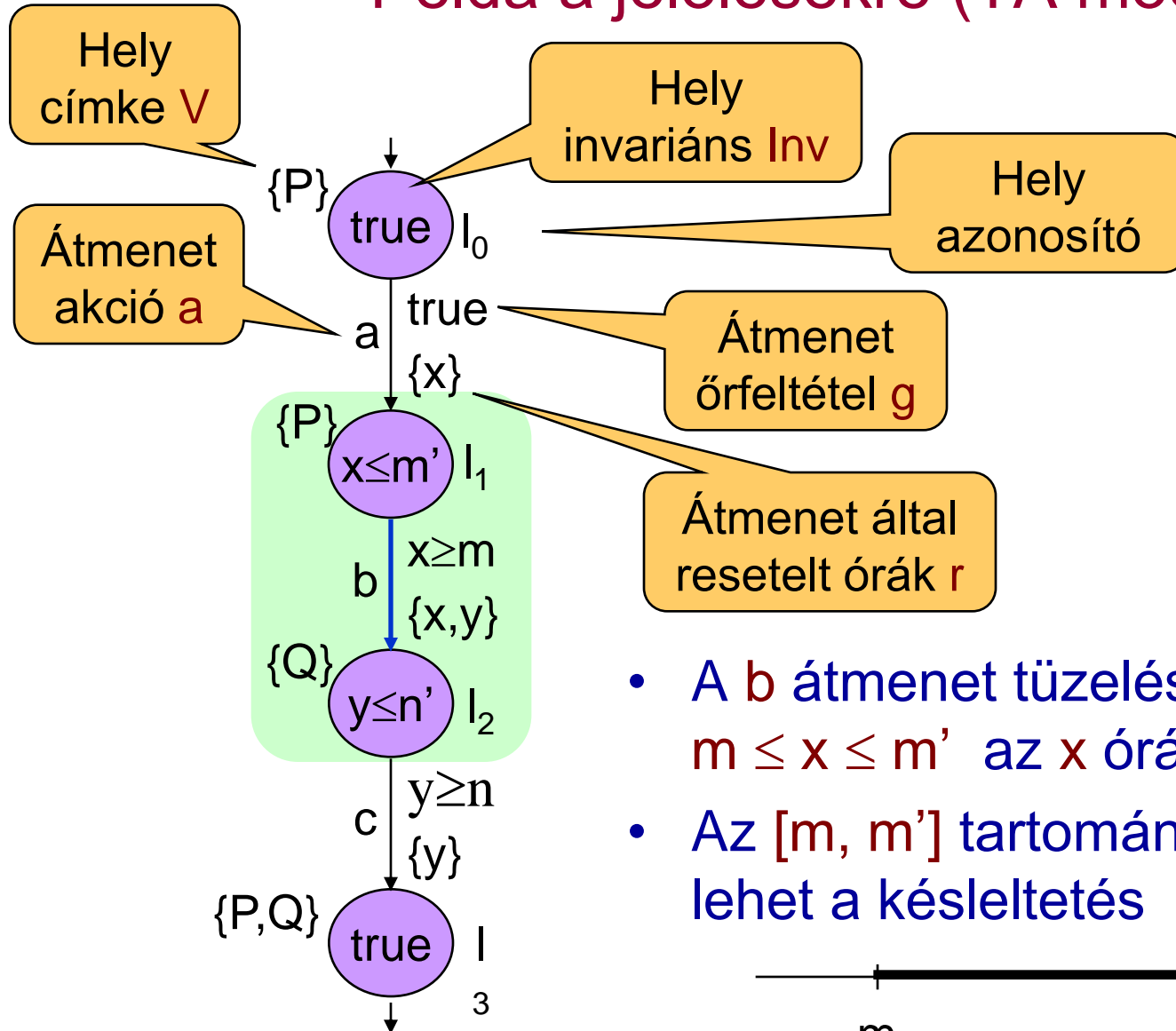
- Adott  $Act$ ,  $AP$ ,  $C$  mellett  $TA=(N, I_0, E, Inv, V)$  ahol
  - $N$  vezérlési helyek (állapotok részei lesznek)
  - $I_0 \in N$  kezdeti hely – itt indul a rendszer, 0 az órák értéke
  - $E \subseteq N \times B(C) \times Act \times 2^C \times N$  élek halmaza, ahol egy él

$$l \xrightarrow{g, a, r} l'$$

itt

- $g$ : óra-kifejezés – engedélyező feltétel
- $a$ : akció – tevékenység az élhez rendelve
- $r$ : óra-halmaz – a resetelt órák halmaza
- $Inv: N \rightarrow B(C)$  óra-invariánsok
  - Eddig növekedhetnek az órák az adott vezérlési helyen
- $V: N \rightarrow 2^{AP}$  címkézés (a helyek lokális tulajdonságai)

# Példa a jelölésekre (TA modell)



- A  $b$  átmenet tüzelésének feltétele:  $m \leq x \leq m'$  az  $x$  órára
- Az  $[m, m']$  tartományban bárhol lehet a késleltetés

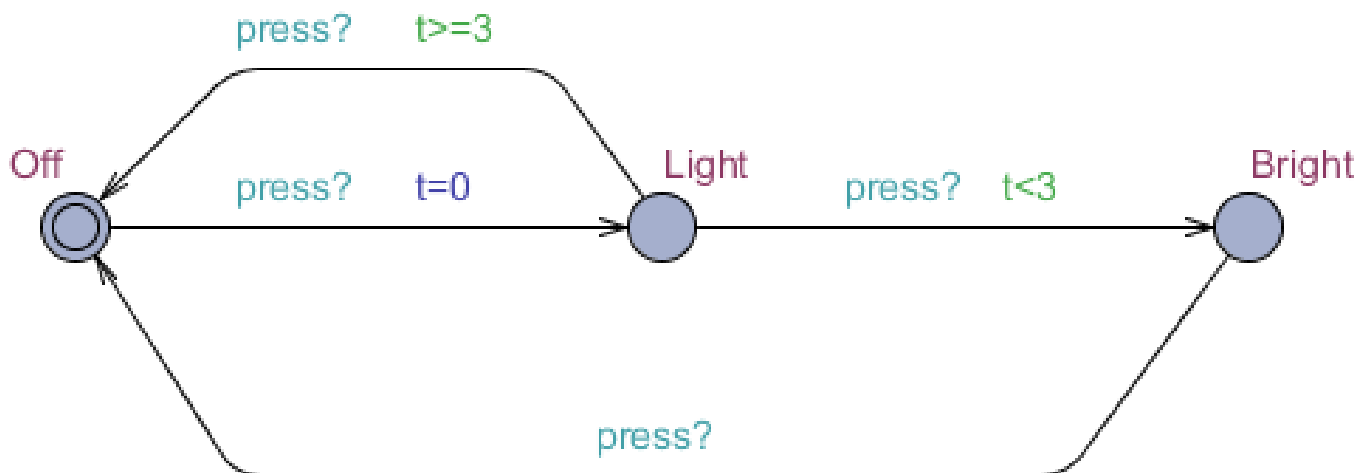




# Egy egyszerű gyakorlati példa (TA modell)

- Villanykapcsoló

- Kikapcsolt állapotban egy lenyomásra a halvány világítás bekapcsol
- Ezután 3 másodpercen belüli újabb lenyomásra a fényes világítás bekapcsol, ezen időn túli lenyomásra kikapcsol
- A fényes világítás állapotából lenyomásra mindig kikapcsol



# TA informális szemantikája

- **Kezdeti állapot:**
  - Kiindulási hely aktív, minden óra 0
- **Késleltetés:** Az órák értéke azonosan (szinkron) növekszik
  - Max. addig telhet egy adott helyen az idő, amíg a hely invariáns igaz
- **Tranzíció tüzelése:**
  - Tranzíció (él) engedélyezett, ha
    - Kiindulási helye aktív
    - Örfeltétel teljesül
    - Szinkronizáció bekövetkezhet (ha van – ld. később)
    - Resetelt órák nullázása teljesíti a cél hely invariánsát
  - Engedélyezett tranzíciók tüzelhetnek (véletlen választás)
    - Akció (értékadás) történik, a resetelt órák értéke 0 lesz
    - Az él célhelye válik aktívvá

# Időzített automaták az UPPAAL eszközben

# Kiterjesztések 1: Változók

- Egész értékű **változók** használata
  - Változók értékadása akciókban
  - Változók hivatkozhatók őrfeltételekben
- **Óra változók: Azonos rátával haladó konkurens órák**
  - Óra változó nullázása (resetelés) akciókban
  - Óra változók hivatkozhatók őrfeltételekben
  - Óra változók hivatkozhatók invariánsokban

# Kiterjesztések 2: Szinkronizált automaták

- Automaták hálózata (párhuzamos kompozíció)

- Szinkronizáció az egyes automaták között:

## Együttlépő átmenetek (randevú)

- Modellezhető: Szinkron kommunikáció (üzenet küldés és fogadás csak együtt valósulhat meg)
- Erre építve aszinkron kommunikáció is leírható

- Akció kiterjesztések:

- Csatornák definiálása (szinkron csatorna)

- Üzenetküldés: **!** operátor a csatornára

Üzenetfogadás: **?** operátor a csatornára

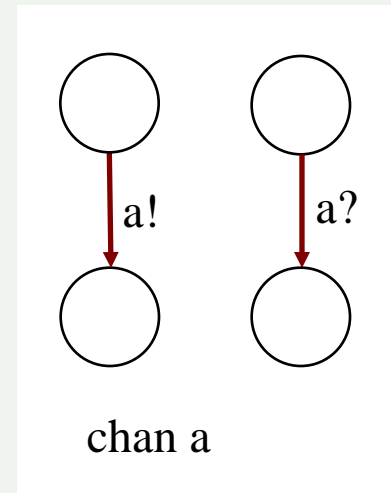
- Pl: az **a** nevű csatorna esetén **a!** és **a?** akció

- Csatornatömbök

- Pl. **a[id]** egy **id** index esetén

- Rendszermodell: Processzek

- Automaták „példányosítása” (paraméterekkel)

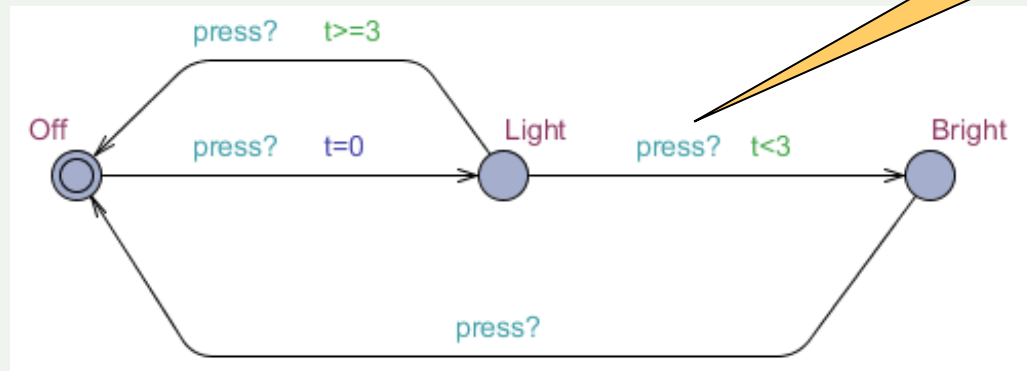


# Példa óraváltózkodra és szinkronizálásra

Deklarációk:

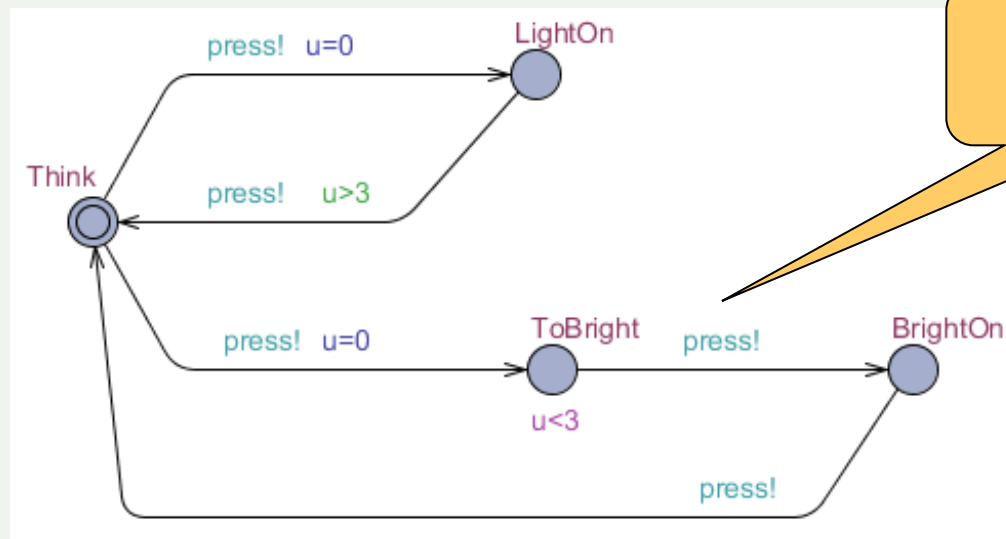
```
clock t, u;  
chan press;
```

Kapcsoló:



„Üzenet fogadás”

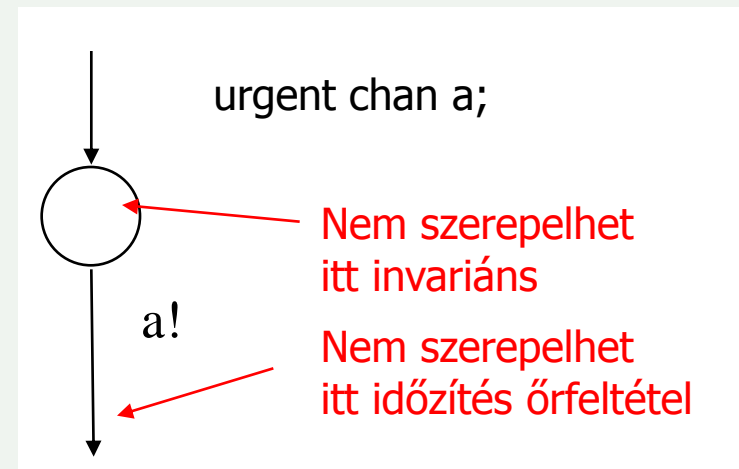
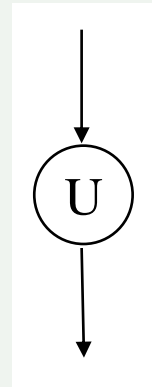
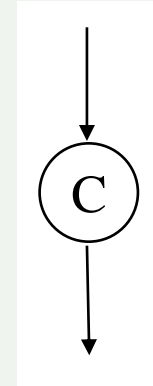
Felhasználó:



„Üzenet küldés”

# További lehetőségek

- **Committed** állapot: Atomi végrehajtást ír elő
  - A kimenő átmenet végrehajtása előtt más automata átmenete nem lehet végrehajtva:  
bemenő és kimenő átmenet egy atomi műveletként végrehajtva
- **Urgent** állapot: késleltetés korlátozása
  - Nem telhet idő az adott állapotban, ha enélkül lehetséges a kilépés
- **Urgent csatorna:**  
Nem enged késleltetést
  - Korlátozások az állapot invariánsok és őrfeltételek használatára
- **Broadcast csatorna**
  - Egy küldő (mindig tud küldeni) és több fogadó (nem szinkron)

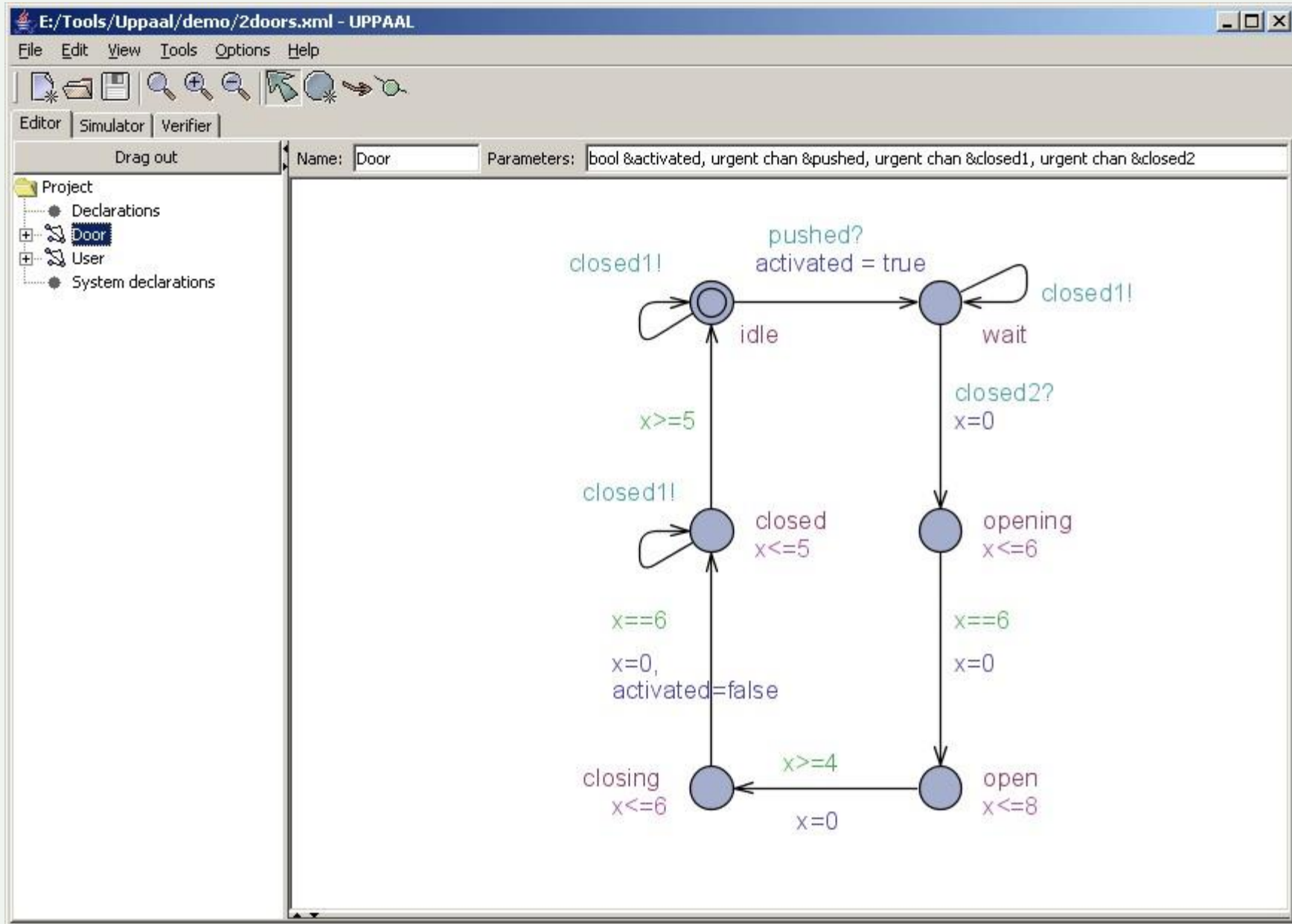


# Az UPPAAL eszköz

- Fejlesztése (1999-től):
  - Uppsala University, Svédország
  - Aalborg University, Dánia
- Web lap (információk, letöltés, példák):  
<http://www.uppaal.org/>
- Kapcsolódó eszközök:
  - CoVer: Test generation with specified coverage
  - CORA: Planning and scheduling (costs)
  - TRON: On-line testing
  - TIGA: Timed games
  - Port: Verification of component based systems



# Automata modell



# Szimulátor

E:/Tools/Uppaal/demo/2doors.xml - UPPAAL

File Edit View Tools Options Help

Editor Simulator Verifier

Drag out

Enabled Transitions

User2

closed2: Door2 --> Door1

Next Reset

Simulation Trace

(idle, idle, idle, idle)

User1

(idle, idle, -, idle)

pushed1: User1 --> Door1

(wait, idle, idle, idle)

Trace File:

Prev Next Replay

Open Save Random

Slow Fast

Drag out

activated1 = 1  
activated2 = 0  
Door1.x >= 0  
Door2.x >= 0  
User1.w = 0  
User2.w >= 0  
Door1.x = Door2.x  
Door2.x = User2.w  
User2.w = Door1.x

**Door1**

**Door2**

**User1**

**User2**

**Door1 Door2 User1 User2**

# TA szemantikája (1)

- Jelölések a szemantikához:
  - $u: C \rightarrow R_{\geq 0}$  óra-hozzárendelés
    - $R^C$  a lehetséges óra-hozzárendelések halmaza  $C$  órahalmazhoz
    - $u \in R^C$  egy hozzárendelés,  $u(x)$  egy  $x$  óra értéke
  - $u+d$  az óra-hozzárendelés növelése minden órára
    - Egy  $x$  óra esetén  $u(x)+d$  lesz az új érték
  - $uv$ : hozzárendelések egyesítése órahalmazokra, ahol  $u \in R^C$ ,  $v \in R^K$  és  $K, C$  függetlenek:  $C \cap K = \emptyset$ 
    - $uv(x) = u(x)$  ha  $x \in C$
    - $uv(x) = v(x)$  ha  $x \in K$
  - $[C' \rightarrow 0]u$  minden  $x \in C'$  órára  $0$  lesz a hozzárendelés, egyébként marad az eredeti
  - $g(u)$  egy  $g$  feltétel kiértékelése  $u$  hozzárendelés mellett
- *TA állapota*:  $(l, u)$  hely és óra-hozzárendelés

## TA szemantikája (2)

- TA szemantikája egy  $TTS=(S, s_0, \rightarrow, V)$  ahol
  - $S$  állapotok mint TA állapotai  $(l,u)$  alakúak
  - $s_0 = (l_0, u_0)$  kezdőállapot
  - $\rightarrow \subseteq S \times L \times S$  definiálható a következőképpen:
    - $(l,u) \rightarrow^a (l',u')$  lehetséges, ha van  $r, g$  úgy, hogy

$l \xrightarrow{g, a, r} l'$  átmenet van helyek között,  
 $g(u)$  örfeltétel igaz,  
 $u' = [r \rightarrow 0]u$  az óra resetelés megtörténik

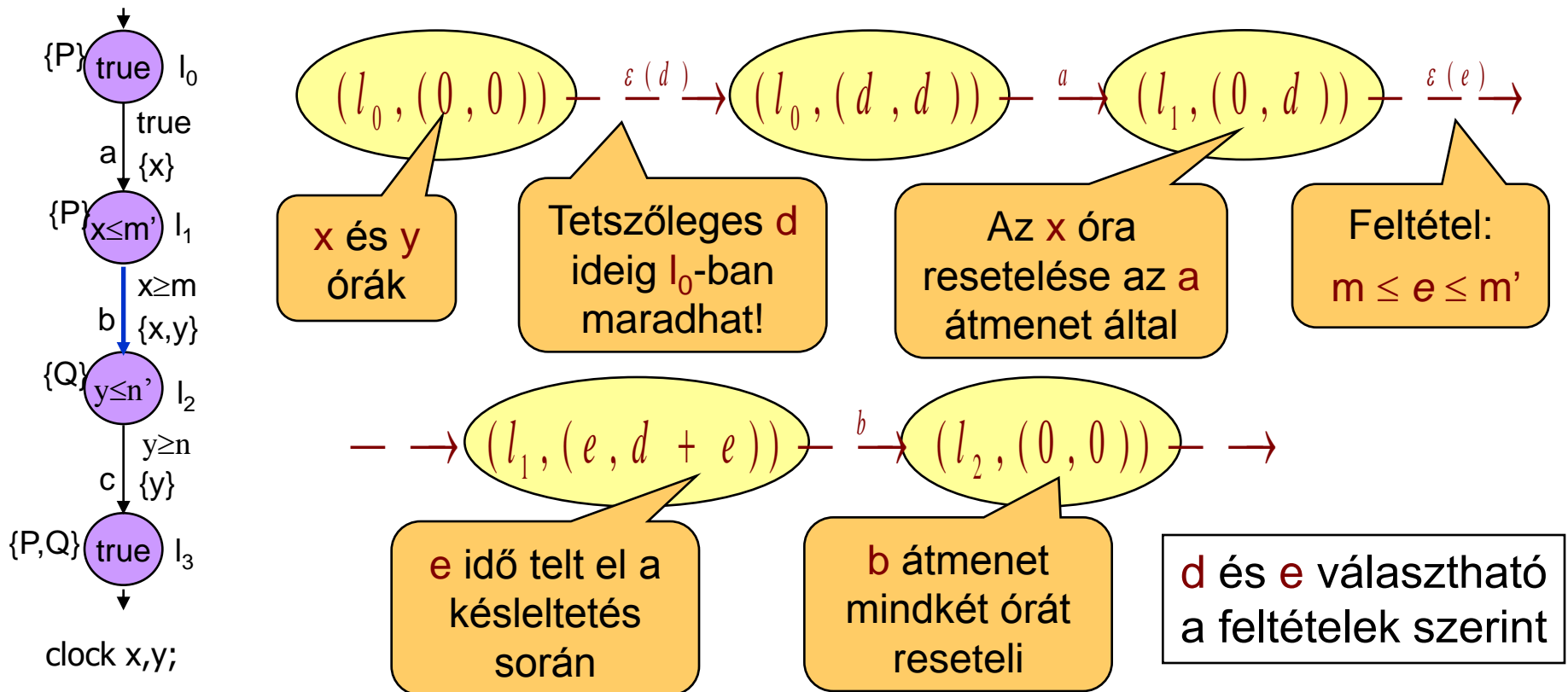
- $(l,u) \rightarrow^{\varepsilon(d)} (l',u')$  lehetséges, ha

$l = l'$  csomópont nem változik,  
 $u' = u + d$  eltelik  $d$  idő,  
 $Inv(u')$  óra-invariáns teljesül

- $V(l,u) = V(l)$  az állapotok címkézése

# Példa a TA szemantikájára

- Egy TA esetén a szemantika egy TTS *halmaz!*
  - Az őrfeltételek és invariánsok sokféle késleltetést engednek (ld. késleltetések értékei tartományokban)
- Egy lehetséges TTS az előbbi TA esetén:

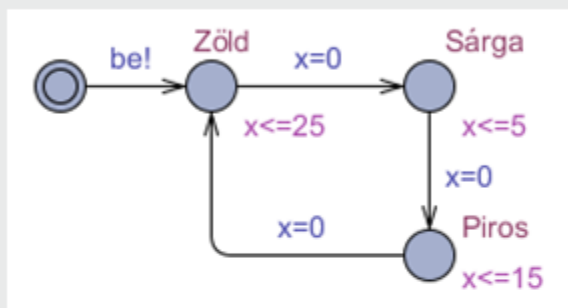


# TA szemantikája (összefoglalás)

Időzített automata  
(TA)

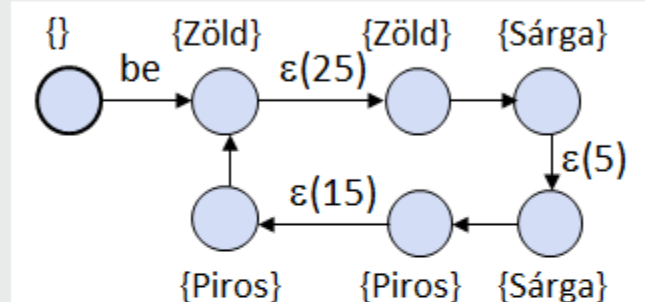
Leképezés

Időzített tranzíciós rendszer  
(TTS)



$TA = (N, l_0, E, Inv, V)$ , ahol

- Helyek véges halmaza:  $N$
- Kezdő hely:  $l_0$
- Élek őrfeltételekkel, akciókkal és nullázott órákkal:  
 $E \subseteq N \times B(C) \times Act \times 2^C \times N$
- Hely invariánsok:  $Inv: N \rightarrow B(C)$
- Helyek címkézése:  $V: N \rightarrow 2^{AP}$



$TTS = (S, s_0, \rightarrow, V)$  ahol

- Állapotok halmaza:  $S$
- Kezdőállapot:  $s_0$
- Tranzíciók:  $\rightarrow \subseteq S \times A \times S$ ,  
 $A = Act \cup \{\varepsilon(d) \mid d \in \mathcal{R}_{\geq 0}\}$ 
  - Akció tranzíciók:  $Act$  címke
  - Késleltetés tranzíciók:  $\varepsilon(d)$
- Címkézés:  $V: S \rightarrow 2^{AP}$

# TA-k párhuzamos kompozíciója

- TA kompozíció: Automaták hálózata
  - Szinkronizáció az egyes automaták között
    - Együttlépő átmenetek
    - Szinkron kommunikáció: Üzenet küldés és fogadás
  - Kompozíció általános meghatározása:
    - Milyen átmenetekre írjuk elő az együttlépést
    - Leírható:  $f$  szinkronizációs függvénnyel, ami akciókon (így implicit az átmeneteken) értelmezett
  - $TA_1 \mid_f TA_2$  kompozíció:
    - Szemantikája TTS-sel fejezhető ki
    - Előtte: TTS-ek kompozícióját kell áttekinteni

# TTS-ek párhuzamos kompozíciója

- **Paraméter: Szinkronizációs függvény  $f$** 
  - $f: (\text{Act} \cup \{0\}) \times (\text{Act} \cup \{0\}) \rightarrow \text{Act}$   
ahol  $0$  a hiányzó akció helye (helyben maradáskor is)
- **Kompozíció:  $\text{TTS}_1 \mid_f \text{TTS}_2 = \text{TTS}_0$ ,**  
ahol  $\text{TTS}_1 = (S_1, s_{1,0}, \rightarrow_1, V_1)$ ,  $\text{TTS}_2 = (S_2, s_{2,0}, \rightarrow_2, V_2)$   
és  $\text{TTS}_0 = (S, s_0, \rightarrow, V)$ 
  - $(s_1 \mid_f s_2) \in S$  (a kompozíció szerint összeállított párok)
  - $s_0 = (s_{1,0} \mid_f s_{2,0}) \in S$  (kezdő hely)
  - $\rightarrow$  induktívan definiálható:
    - $(s_1 \mid_f s_2) \rightarrow^e (s'_1 \mid_f s'_2)$  ha  $s_1 \xrightarrow{a_1} s'_1$  és  $s_2 \xrightarrow{b_2} s'_2$  és  $f(a,b)=e$
    - $(s_1 \mid_f s_2) \xrightarrow{\varepsilon^{(d)}} (s'_1 \mid_f s'_2)$  ha  $s_1 \xrightarrow{\varepsilon^{(d)}} s'_1$  és  $s_2 \xrightarrow{\varepsilon^{(d)}} s'_2$
  - $V(s_1 \mid_f s_2) = V_1(s_1) \cup V_2(s_2)$



# TA párhuzamos kompozíció szemantikája

- Jelölés:  $TA_1 \mid_f TA_2$  automaták hálózata
- $TA_1 \mid_f TA_2$  szemantikája  $TTS_1 \mid_f TTS_2$  ahol
  - $TA_1$  szemantikája  $TTS_1$ ,  $TA_2$  szemantikája  $TTS_2$
  - Érdekesség: Konstruálható olyan  $TA_1 \otimes TA_2$  szorzat automata, hogy  $TA_1 \otimes TA_2$  szemantikájára:  $TTS_{TA_1 \otimes TA_2} \sim TTS_1 \mid_f TTS_2$ , azaz biszimuláció ekvivalensek (biszimuláció fogalma: ld. később)
- UPPAAL esetén az  $f$  szinkronizációs fv:
  - $f(a!, a?) = 0$  együttes akció ( $a!$  „küldés” és  $a?$  „fogadás”)
  - $f(a, 0) = a$  egyik automata akciója
  - $f(0, a) = a$  egyik automata akciója

# TA hálózat szemantikája (összefoglalás)



- Szinkronizált akciók megadása:

$$f: (\text{Act} \cup \{\}) \times (\text{Act} \cup \{\}) \rightarrow \text{Act}$$

- $TA_1$  szemantikája:

$$TTS_1 = (S_1, s_{1,0}, \rightarrow_1, V_1)$$

- $TA_2$  szemantikája:

$$TTS_2 = (S_2, s_{2,0}, \rightarrow_2, V_2)$$

$TTS_0 = TTS_1 \mid_f TTS_2 = (S, s_0, \rightarrow, V)$ , ahol

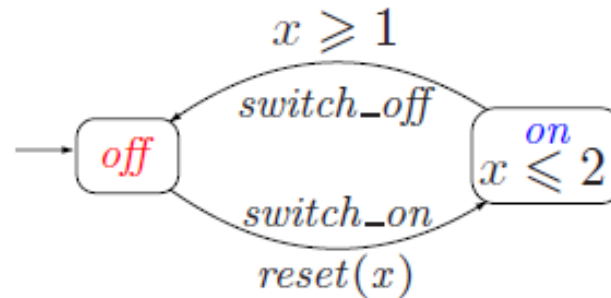
- Állapotok:  $(s_1, s_2)$  párok,  $s_1 \in S_1, s_2 \in S_2$
- Kezdőállapot:  $s_0 = (s_{1,0}, s_{2,0})$
- Tranzíciók:  $\rightarrow$  definiálható:
  - Akció:  $(s_1, s_2) \rightarrow^e (s_1', s_2')$   
ha  $s_1 \xrightarrow{a_1} s_1'$  és  $s_2 \xrightarrow{b_2} s_2'$  és  $f(a,b)=e$
  - Késleltetés:  $(s_1, s_2) \xrightarrow{\varepsilon^{(d)}} (s_1', s_2')$   
ha  $s_1 \xrightarrow{\varepsilon^{(d)}} s_1'$  és  $s_2 \xrightarrow{\varepsilon^{(d)}} s_2'$
- Állapotok címkézése:  
 $V(s_1, s_2) = V_1(s_1) \cup V_2(s_2)$

# Érdekességek időzített rendszerek modellezésében

- Nem realiztikus végrehajtási útvonalak:  
Megnehezítik az ellenőrzést
  - **Idő konvergencia** (time convergence): Végtelen késleltetés sorozat egy konkrét időponthoz konvergál
  - **Időzár** (timelock): Megáll az idő haladása
  - **Zeno** (Zénón) tulajdonság (zenoness): Végtelen sok akció végrehajtható véges idő alatt
- **Kezelésük:**
  - Az idő konvergenciára a modellellenőrzés során kell figyelni (lehetséges, de nem „fair” útvonalként kizárva)
  - Az időzár és a zeno tulajdonság a modell megfelelő kialakításával elkerülhetők

# Idő konvergencia (time convergence)

- Példa modell:



- Példa útvonal:

$$\langle \textit{off}, 0 \rangle \xrightarrow{2^{-1}} \langle \textit{off}, 1-2^{-1} \rangle \xrightarrow{2^{-2}} \langle \textit{off}, 1-2^{-2} \rangle \xrightarrow{2^{-3}} \langle \textit{off}, 1-2^{-3} \rangle \dots\dots$$

- Idő konvergens útvonal általánosan:

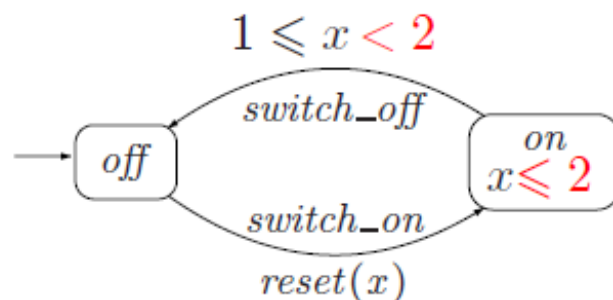
- Végtelen  $d_1, d_2, \dots$  késleltetés sorozat,  
ahol  $d_1+d_2+\dots$  összege  $d$  (konstans)

- Idő divergens útvonal:

- Nem konstans végrehajtási idő (késleltetések összege)

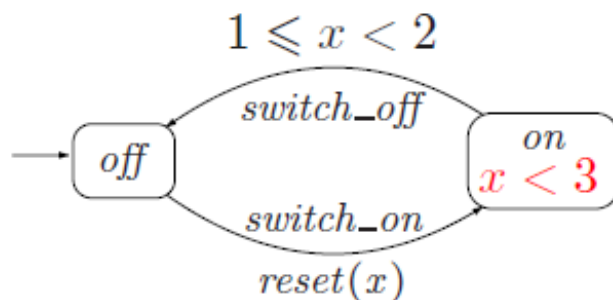
## Időzár (timelock)

- Egy állapot időzárnak tekinthető, ha nincs belőle divergens útvonal
  - A kimenő útvonalakon nem „telhet tovább” az idő (a végtelenségig)
  - Terminális állapot
    - A végállapot nem feltétlenül jelent időzárat (pl. az invariáns true)
- Egy példa automata, ahol van időzár:
  - $(on, 2)$  elérhető, és nincs belőle divergens útvonal



# Időzár konvergens útvonallal

- A példa modell:

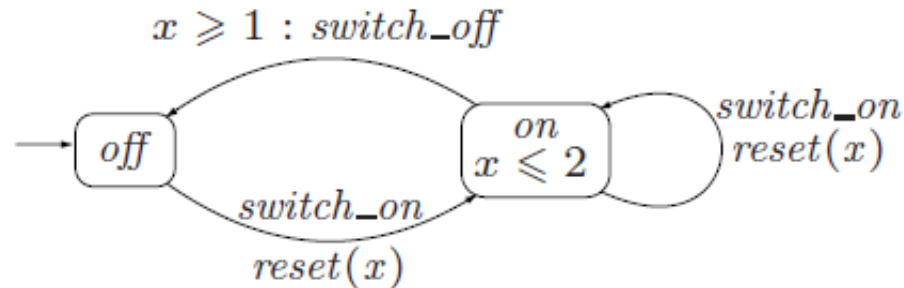


- Konvergens útvonal:

$\langle on, 2 \rangle \langle on, 2.9 \rangle \langle on, 2.99 \rangle \langle on, 2.999 \rangle \langle on, 2.9999 \rangle \dots$

# Zeno tulajdonság

- Zeno útvonal:
  - Idő-konvergens, ugyanakkor
  - végtelenül sok  $a \in \text{Act}$  akció végrehajtható
- Példa modell:



- Zeno útvonalak:

switch\_on hurok késleltetés nélkül

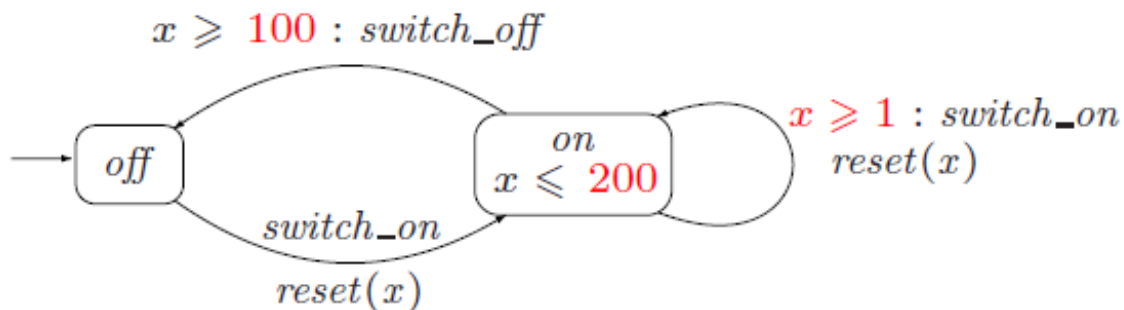
$\langle \text{off}, 0 \rangle \xrightarrow{\text{sw\_on}} \langle \text{on}, 0 \rangle \xrightarrow{\text{sw\_on}} \langle \text{on}, 0 \rangle \xrightarrow{\text{sw\_on}} \langle \text{on}, 0 \rangle \xrightarrow{\text{sw\_on}} \dots$

$\langle \text{off}, 0 \rangle \xrightarrow{\text{sw\_on}} \langle \text{on}, 0 \rangle \xrightarrow{0.5} \langle \text{on}, 0.5 \rangle \xrightarrow{\text{sw\_on}} \langle \text{on}, 0 \rangle \xrightarrow{0.25} \langle \text{on}, 0.25 \rangle \xrightarrow{\text{sw\_on}} \dots$

Vannak késleltetések, de 1-et nem éri el ezek összege:  
 $0,5 + 0,25 + 0,125 + \dots$

# A zeno tulajdonság elkerülése

- Az előző példa esetén:
  - (Minimális) késleltetés az egymás utáni `switch_on` akciók között (így telik az idő)
- A módosított példa modell:
  - Minimális késleltetés 1 egység (egész óraérték esetén)
  - Arányában megnövelve a vezérlés-specifikus késleltetés





# Időzített logika (Timed TL) bevezetése

- **Elvárások:**
  - Órák használata (intuitív)
  - **Rekurzió** megengedett
  - Elégséges a TA-n megfogalmazható leggyakoribb követelményekhez
  - Eldönthető a kifejezések igazsága
- **Jelölések:**
  - **K** óra-halmaz (formula-órák)
    - A modell óráival azonos rátával haladnak, de külön hivatkozhatók
  - **Id** azonosítók (formulában értékadáshoz; rekurzió lehet!)
    - $Z \in Id$  változó
    - $D(Z)$  hozzárendelés  $Z$  változóhoz,  $D(Z) = \varphi$ , más alakban  $Z := \varphi$

# TL szintaxis

- $\varphi ::= P \mid c \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists \varphi \mid \forall \varphi \mid \langle a \rangle \varphi \mid [a] \varphi \mid x \text{ in } \varphi \mid x+n \sim y+m \mid Z$

ahol  $P \in AP$ ,  $c \in B(K)$ ,  $a \in Act$ ,  $x \in K$ , és  $Z \in Id$ ,  $m, n \in \mathbb{N}$

- Temporális operátorok informálisan:
  - $\exists \varphi$  – létezik olyan késleltetés, hogy  $\varphi$  igaz lesz
  - $\forall \varphi$  – minden késleltetésre  $\varphi$  igaz lesz
  - $x \text{ in } \varphi$  –  $x$  resetelésével  $\varphi$  igaz lesz
  - $x+n \sim y+m$  – óra-kifejezések összevetése
- TL értelmezhető (igazsága megállapítható) TTS-ek felett (így közvetve TA-k felett is)
  - $s: (l, u)$  TTS állapotok (TA-ból)
  - $(s, v)$  jelölés TTS-beli állapotra és formula-órára,  $v \in \mathbb{R}^K$

# TL szemantika (1)

- $(s,v) \models P$  atomi kijelentésre a.cs.a.  $P \in V(s)$   
azaz  $P$  az  $s$  állapot címkéi között van
- $(s,v) \models c$  órakifejezésre a.cs.a.  $c(v)$  igaz,  
azaz a  $v$  által meghatározott óra-hozzárendelés esetén  
igaz a  $c$  óra-kifejezés
- $(s,v) \models \varphi_1 \wedge \varphi_2$  a.cs.a.  $(s,v) \models \varphi_1$  és  $(s,v) \models \varphi_2$
- $(s,v) \models \varphi_1 \vee \varphi_2$  a.cs.a.  $(s,v) \models \varphi_1$  vagy  $(s,v) \models \varphi_2$
- $(s,v) \models \exists \varphi$  a.cs.a.  $\exists d, s': s \xrightarrow{\varepsilon(d)} s'$  és  $(s', v+d) \models \varphi$   
azaz létezik olyan,  $(s,v)$ -ből késleltetéssel elérhető állapot,  
ahol igaz  $\varphi$
- $(s,v) \models \forall \varphi$  a.cs.a.  $\forall d, s': s \xrightarrow{\varepsilon(d)} s' \Rightarrow (s', v+d) \models \varphi$   
azaz minden,  $(s,v)$ -ből késleltetéssel elérhető állapotban  
igaz  $\varphi$

## TL szemantika (2)

- $(s,v) \models \langle a \rangle \varphi$  a.cs.a.  $\exists s': s \xrightarrow{a} s'$  és  $(s',v) \models \varphi$   
azaz létezik olyan,  $(s,v)$ -ből  $a$  akcióval elérhető állapot,  
ahol igaz  $\varphi$
- $(s,v) \models [a]\varphi$  a.cs.a.  $\forall s': s \xrightarrow{a} s' \Rightarrow (s',v) \models \varphi$   
azaz minden,  $(s,v)$ -ből  $a$  akcióval elérhető állapotban  
igaz  $\varphi$
- $(s,v) \models x \text{ in } \varphi$  a.cs.a.  $(s,v') \models \varphi$  ahol  $v' = [\{x\} \rightarrow 0]v$   
azaz  $x$  formula-óra resetelésével igaz lesz  $\varphi$
- $(s,v) \models x+n \sim y+m$  a.cs.a.  $v(x)+n \sim v(y)+m$   
azaz a formula-órák értéke megfelelő
- $(s,v) \models Z$  a.cs.a.  $(s,v) \models D(Z)$   
azaz a  $Z$ -hez rendelt formula igaz

# TL jellemzők

- Eredeti logika:

$\varphi ::= c \mid P \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists \varphi \mid \forall \varphi \mid \langle a \rangle \varphi \mid [a] \varphi \mid$   
 $x \text{ in } \varphi \mid x+n \sim y+m \mid Z$

- Alacsony szintű, nagyon egyszerű operátorok

- Egzisztenciális és univerzális operátorok késleltetéssel illetve akcióval címkézett átmenetekre
- „Alap logika” (hasonló szerepű, mint a  $\mu$ -kalkulus)
- Kifejezőképessége nagy (mivel a rekurzió is megengedett), de önmagában nehezen használható

- Tovább lépés:

- Összetett operátorok képzése az egyszerűbbekből
  - Közelebb állnak az intuíciónak és a gyakorlati követelményeknek: pl. invaráns, UNTIL, UNTIL<sub><t</sub>, BEFORE t
- Korlátozások bevezetése az egyes modellellenőrökben a hatékonyabb ellenőrzési algoritmusok érdekében
  - pl. UPPAAL, KRONOS

# Hasznos kifejezések

- $INV(\varphi)$  invariáns: az a  $Z$  kifejezés,  
ahol  $Z := \varphi \wedge \forall Z \wedge [Act]Z$

A rekurzió miatt  
kell  $Z$  bevezetése

itt  $[Act]Z$  jelentése  $[a_1]Z \wedge [a_2]Z \wedge \dots \wedge [a_n]Z$  minden  $a_i \in Act$ -ra

# Hasznos kifejezések

- $INV(\varphi)$  invariáns: az a  $Z$  kifejezés,

ahol  $Z := \varphi \wedge \forall Z \wedge [Act]Z$

itt  $[Act]Z$  jelenti  $[a_1]Z \wedge [a_2]Z \wedge \dots \wedge [a_n]Z$  minden  $a_i \in Act$ -ra

A rekurzió miatt  
kell  $Z$  bevezetése

Minden, késleltetéssel  
elérhető állapotban  $Z$   
újra igaz lesz

Minden, akcióval  
elérhető állapotban  $Z$   
újra igaz lesz

Ezek együtt: Általános  
„next” kifejezése

# Hasznos kifejezések

- $INV(\varphi)$  invariáns: az a  $Z$  kifejezés,

ahol  $Z := \varphi \wedge \forall Z \wedge [Act]Z$

itt  $[Act]Z$  jelentése  $[a_1]Z \wedge [a_2]Z \wedge \dots \wedge [a_n]Z$  minden  $a_i \in Act$ -ra

- $\varphi_1$  UNTIL  $\varphi_2$  „weak until”: az a  $Z$ ,

ahol  $Z := \varphi_2 \vee (\varphi_1 \wedge \forall Z \wedge [Act]Z)$

$\varphi_2$  nem feltétlenül  
kell teljesülnön

- $\varphi_1$  UNTIL<sub><n</sub>  $\varphi_2 \equiv x$  in  $((\varphi_1 \wedge x < n)$  UNTIL  $\varphi_2)$

itt  $x$  resetelés után értékelendő ki, vagyis  $n$  relatív idő

- $\varphi$  BEFORE  $n \equiv true$  UNTIL<sub><n</sub>  $\varphi$

- Példa:  $at(l_i)$  BEFORE  $t$  az  $l_i$  hely elérése  $t$  előtt

Itt jelölés:  $at(l_i)$  – az  $l_i$  helyen van az automata



# Egyszerűsítés hatékonyabb kiértékeléshez

- Eredeti logika:

$\varphi ::= c \mid P \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists \varphi \mid \forall \varphi \mid \langle a \rangle \varphi \mid [a] \varphi \mid$   
 $x \text{ in } \varphi \mid x+n \sim y+m \mid Z$

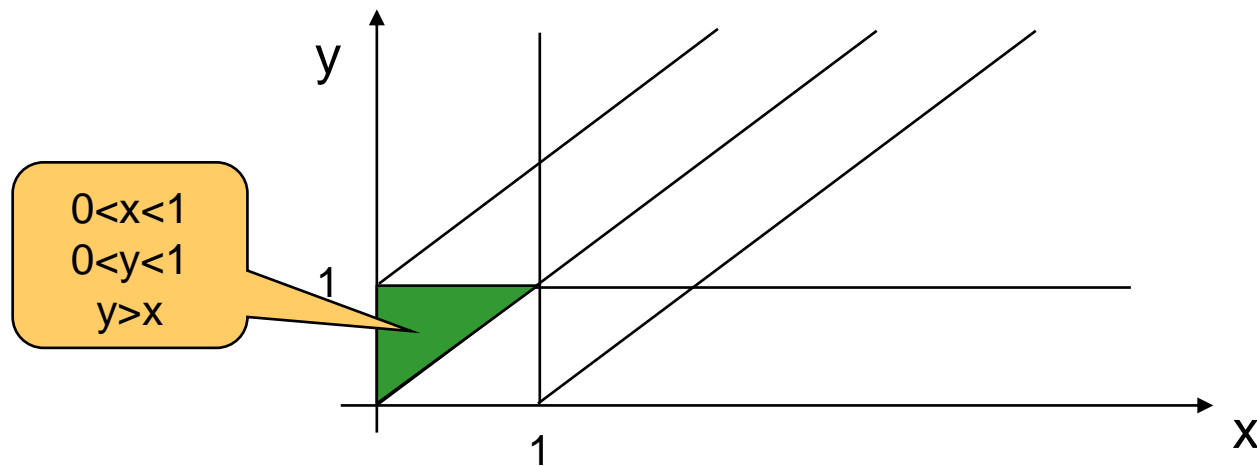
- Biztonsági és kötött előségi kritériumok ellenőrzéséhez:

- $\exists \varphi$  kihagyva (késleltetéseken egzisztenciális kvantor)
- $\langle a \rangle$  kihagyva (akciókon egzisztenciális kvantor)
- $c \vee \varphi$  alakú formulák
- $P \vee \varphi$  alakú formulák

Invariáns, UNTIL, UNTIL<sub><n</sub>, BEFORE t kifejezhető

# Modellellenőrzés alapgondolata

- Régiók megállapítása, ahol azonosan értékelhetők ki a feltételek
  - A régiókon értelmezett a TL igazsága
    - Sokféle késleltetés tud igazzá tenni egy követelményt; ezeket az invariánsok és őrfeltételek határolják be;
    - Így alakulnak ki a régiók az óra-változókon
- Szemantika alapú modellellenőrzés:
  - Kényszer-kielégítési problémára is visszavezethető
  - Modellellenőrzés: Van-e olyan óra-hozzárendelés, hogy  $\varphi$  igaz?



# UPPAAL modellellenőrző

- **Temporális logika:**
  - $\Phi ::= A[] \beta \mid E\langle\rangle \beta$   
ahol  $\beta ::= a \mid \beta \text{ and } \beta \mid \beta \text{ or } \beta \mid \beta \text{ implies } \beta \mid \text{not } \beta$ 
    - itt  $a$  lehet  $A_i.l$  vagy  $x_i \sim n$  alakú,
    - itt  $A_i.l$  jelöli, hogy az  $A_i$  automata az  $l$  helyen van
  - **Speciális operátor:  $-->$  („leadsto”)**
    - $a --> b$  egy rövidítés:  $A[] (a \text{ implies } A\langle\rangle b)$
  - **Speciális atomi kijelentés: deadlock**
    - Arra az állapotra igaz, ahonnan nincs továbblépés
    - $A[] \text{not deadlock}$  jellegzetes kritérium



Overview

```
A[] not (Door1.open and Door2.open)
A[] (Door1.opening imply User1.w<=31) and (Door2.opening imply User2.w<=31)
E<> Door1.open
E<> Door2.open
Door1.wait --> Door1.open
Door2.wait --> Door2.open
A[] not deadlock
```

Check  
Insert  
Remove  
Comments

Query

```
A[] not (Door1.open and Door2.open)
```

Comment

Mutex: The two doors are never open at the same time.

Status

```
Established direct connection to local server.
UPPAAL version 4.0.4 (rev. 2887), January 2007 -- server.
Disconnected.
Established direct connection to local server.
UPPAAL version 4.0.4 (rev. 2887), January 2007 -- server.
Disconnected.
Established direct connection to local server.
UPPAAL version 4.0.4 (rev. 2887), January 2007 -- server.
A[] not (Door1.open and Door2.open)
Property is satisfied.
A[] (Door1.opening imply User1.w<=31) and (Door2.opening imply User2.w<=31)
Property is satisfied.
```

# KRONOS modellellenőrző

- TA formalizmus használatos itt is
- TCTL temporális logika
  - $\exists \langle \rangle$  (EF-nek megfelelő)
  - $\forall []$  (AG-nek megfelelő)
  - $\exists \langle \rangle_{=1}$  (egy időegység alatt elérhető)
  - $\forall []_{\leq 5}$  (legfeljebb 5 időegység alatt elérhető)
- Érdekes követelmény (gyakran megadják):
  - $\forall [] \exists \langle \rangle_{=1} \text{ true}$ 
    - azaz minden állapotban igaz, hogy *telhet az idő* 1 egységet
    - a kritériumok nem azért teljesülnek, mert „megáll az idő”

# Miről volt szó?

- Motiváció: Valós idejű rendszerek ellenőrzése
- Modellek
  - Timed Transition System (TTS)
  - Timed Automata (TA)
  - Leképezés: TA  $\rightarrow$  TTS
- Érdekességek valós idejű rendszerek modellezése során
  - Idő konvergencia, időzár, zeno tulajdonság
- Követelmények formalizálása
  - TL (Timed Temporal Logic)
  - Timed CTL variánsok
- Modellellenőrzés
  - Korlátozások, alapötlet