# IV113 Validace a verifikace

## Převod LTL formule na Büchi automat

Jiří Barnat

# Připomenutí

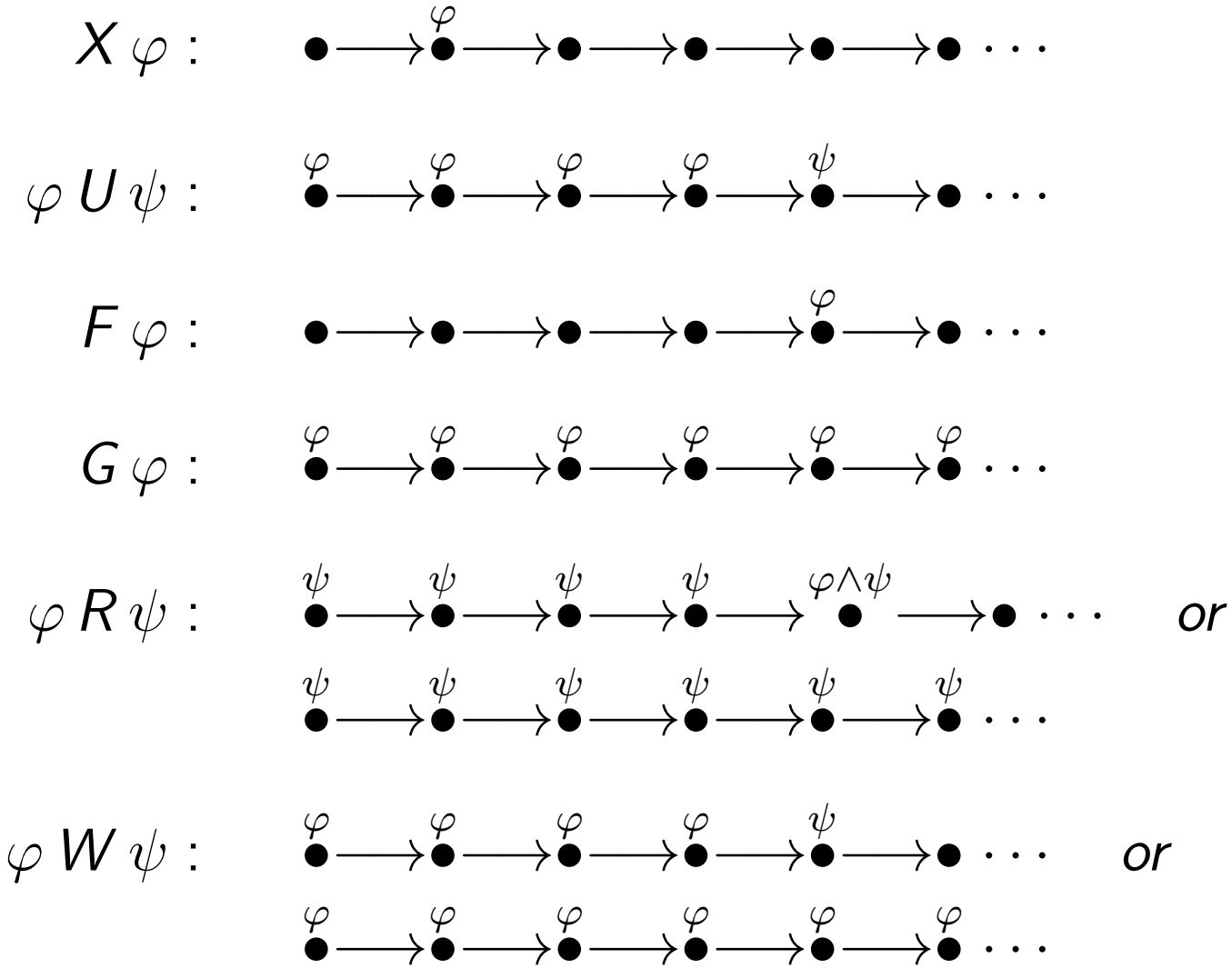
**Problém**

- Kripkeho struktura $M$
- LTL formule $\varphi$
- $M \models \varphi$ ?

**Řešení pomocí Büchi automatů**

- $A_{sys}$ – automat akceptující běhy modelu
- $A_{\neg\varphi}$ – automat akceptující běhy porušující vlastnost $\varphi$
- $L(A_{sys}) \cap L(A_{\neg\varphi}) = L(A_{sys} \times A_{\neg\varphi})$
- $L(A_{sys} \times A_{\neg\varphi}) \neq \emptyset \iff$ model má běh porušující $\varphi$
- $M \models \varphi \iff A_{sys} \times A_{\neg\varphi}$ nemá akceptující cyklus

# Grafické znázornění sémantiky temporálních operátorů



# Příklad

Nechť $\Sigma = \{a, b, c\}$, najděte Büchi automat, který akceptuje $\omega$-regulární jazyk definovaný následující LTL formulí.

a) $a \, U \, b$
b) $a \, U \, (X \, b)$
c) $\neg(a \, U \, (X \, b))$
d) $a \, U \, (b \, U \, c)$
e) $\neg(a \, U \, (b \, U \, c))$

# Algoritmus převodu
# LTL formule na Büchi automat

# Postup převodu

Vstup: Množina atomických propozic $AP$, LTL formule $\varphi$.
Výstup: Büchi automat $A$ takový, že $L(A) = L_\varphi$.

**Postup:**

- Formule $\varphi$ se převede do normální formy.
- Vypočítá se přechodový graf budoucího automatu.
- Graf se doplní na zobecněný Büchi automat.
- Zobecněný BA se převede na standardní Büchi automat.

# Normální forma LTL formule

Řekneme, že LTL formule je v normální formě, pokud neobsahuje operátory $F$ a $G$, a všechny operátory unární negace jsou aplikovány na podformule tvořené pouze atomickou propozicí.

## Syntax

$$\varphi ::= p \mid \neg p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid X\,\varphi \mid \varphi\, U\, \varphi \mid \varphi\, R\, \varphi$$

## Pravidla pro převod do normální formy

$$\begin{aligned}
\neg(\varphi \vee \psi) &\equiv (\neg\varphi) \wedge (\neg\psi) \\
\neg(\varphi \wedge \psi) &\equiv (\neg\varphi) \vee (\neg\psi) \\
\neg X\varphi &\equiv X(\neg\varphi) \\
\neg(\varphi\, U\, \psi) &\equiv (\neg\varphi\, R\, \neg\psi) \\
\neg(\varphi\, R\, \psi) &\equiv (\neg\varphi\, U\, \neg\psi)
\end{aligned}$$

# Příklad

Nechť $AP = \{a, b\}$. Převeďte následující LTL formule do normální formy

a) $G(F(a))$
b) $F(G(a))$
c) $\neg(G(F(a))$
d) $G(a \implies F(b))$
e) $\neg(a\, U\, (G\, b))$

# Zobecněné Büchi automaty

## Büchi automaty

- $A = (\Sigma, S, s, \delta, F)$
- $F \subseteq S$ je množina koncových stavů.
- Běh $\rho$ je akceptující, pokud $inf(\rho) \cap F \neq \emptyset$.

## Zobecněné Büchi automaty

- $A = (\Sigma, S, s, \delta, \mathcal{F})$
- $\mathcal{F} \subseteq 2^S$ je systém množin koncových stavů.
- Běh $\rho$ je akceptující, pokud $\forall F_i \in \mathcal{F}$ platí $inf(\rho) \cap F_i \neq \emptyset$.

# Příklad

Nechť $\Sigma = \{a, b\}$ a $L = \{w \in \Sigma^\omega \mid inf(w) = \{a, b\}\}$.

Najděte zobecněný BA $\mathcal{A}$ takový, že $L(\mathcal{A}) = L$.

# Zobecněné Büchi automaty

**Tvrzení**

- Ke každému zobecněnému Büchi automatu $A$ existuje (normální) Büchi automat $B$ takový, že $L(A) = L(B)$.

**Konstrukce**

- Nechť $A = (\Sigma, S, s, \delta, \{F_1, \ldots, F_n\})$.
- $B = (\Sigma, S \times \{0, \ldots, n\}, (s, 0), \delta', S \times \{n\})$, kde
- $(q, y) \in \delta'((p, x), a)$ pokud $q \in \delta(p, a)$ a pro $x$ a $y$ platí
  - jestliže $q \in F_i$ a $x = i - 1$, tak $y = i$
  - jestliže $x = n$, tak $y = 0$
  - jinak $x = y$.

# Příklad ZBA $\rightarrow$ BA

ZBA: $\mathcal{F} = \{F_1 = \{p\}, F_2 = \{q\}\}$

| | a | b |
|---|---|---|
| $\rightarrow$ p | p | q |
| q | p | q |

BA: $F = \{(p, 2), (q, 2)\}$

| | a | b |
|---|---|---|
| $\rightarrow$ (p,0) | (p,1) | (q,0) |
| (q,0) | (p,1) | (q,0) |
| (p,1) | (p,1) | (q,2) |
| (q,1) | (p,1) | (q,2) |
| $\leftarrow$ (p,2) | (p,0) | (q,0) |
| $\leftarrow$ (q,2) | (p,0) | (q,0) |

# Výpočet přechodového grafu

# Přechodový graf intuitivně

**Pozorování**

- Přechod v Büchi automatu stráží jeden stav běhu.
- Pro definici přechodu, je třeba vědět, co platí v aktuálním stavu, a co má platit v následujícím stavu běhu.

**Rozbalené definice modálních operátorů**

$$
\begin{array}{lcll}
X\,a & \equiv & tt & \wedge \quad X(\,a\,) \\
a\,U\,b & \equiv & a & \wedge \quad X(\,a\,U\,b\,) \\
 & \vee & b & \wedge \quad X(\,tt\,) \\
a\,R\,b & \equiv & b & \wedge \quad X(\,a\,R\,b\,) \\
 & \vee & a \wedge b & \wedge \quad X(\,tt\,)
\end{array}
$$

# Přechodový graf intuitivně

**Pozorování**

- Přechod v Büchi automatu stráží jeden stav běhu.
- Pro definici přechodu, je třeba vědět, co platí v aktuálním stavu, a co má platit v následujícím stavu běhu.

**Rozbalené definice modálních operátorů**

| New | | Now | Next |
|---|---|---|---|
| $X\,a$ | $\equiv$ | | $a$ |
| $a\,U\,b$ | $\equiv$ | $a$ | $a\,U\,b$ |
| | $\vee$ | $b$ | |
| $a\,R\,b$ | $\equiv$ | $b$ | $a\,R\,b$ |
| | $\vee$ | $a \wedge b$ | |

# Přechodový graf LTL formule

## Uzel je uspořádaná pětice

- Id – Číslo
  - Unikátní označení uzlu.
- Incoming – Množina označení uzlů.
  - Množina přímých předchůdců vrcholu ve výsledném grafu.
  - Kóduje hrany grafu.
- Now – Množina LTL formulí.
  - Seznam podformulí, které platí v daném uzlu.
- New – Množina LTL formulí.
  - Množina ještě nezpracovaných formulí, které musí být splněny v tomto uzlu.
- Next – Množina LTL formulí.
  - Seznam formulí, které musí být splněny v následujícím uzlu.

# Graf LTL formule

**Vytvoření grafu** (výpočet množiny uzlů)

proc $create\_graph(\varphi)$
  $N = (new\_ID(), \{init\}, \emptyset, \{\varphi\}, \emptyset)$
  return $expand(N, \emptyset)$
end

**Pomocné funkce**

- $expand(n, Nodes)$
  - Funkce volaná pro uzel $n$ a dosud známe uzly $Nodes$.
  - Vrací množinu uzlů (po zpracování uzlu $n$).
- $new\_ID()$
  - Každé volání této funkce vrátí dosud nevrácené číslo.
- $Neg(\_)$
  - $Neg(A) = \neg A$ pro všechny $A \in AP$.
  - $Neg(True) = False$
  - $Neg(False) = True$
  - $\neg\neg A = A$

# Graf LTL formule – funkce expand

```
proc expand(q, Nodes)
  if New(q) == ∅
    then if (∃r ∈ Nodes takový, že Now(r) == Now(q) ∧ Next(r) == Next(q))
           then Incoming(r) = Incoming(r) ∪ Incoming(q)
                return Nodes
           else N = (new_ID(), {ID(q)}, ∅, Next(q), ∅)
                return expand(N, Nodes ∪ {q})          /* q je nový uzel */
         fi
    else let η ∈ New(q)
         New(q) = New(q) ∖ {η}
         if η ∈ Now(q)                                 /* η již byla zpracována */
           then return expand(q, Nodes)
         fi
         switch (η)    /* pokračuj podle typu nejvnějšnějšího operátoru η */
         ...
         end
  fi
end
```

# Graf LTL formule – funkce expand (switch)

switch $(\eta)$ $\quad$ /* pokračuj podle typu nejvnějšnějšího operátoru $\eta$ */

case $(\eta \in (AP \cup Neg(AP) \cup \{True, False\}))$
  if $(\eta == False \vee Neg(\eta) \in Now(q))$
    then return Nodes
     else N $= (new\_ID(), Incoming(q), Now(q) \cup \{\eta\}, New(q), Next(q))$
          return *expand*(N, Nodes)
  fi
end

case $(\eta \equiv \varphi \, U \, \psi)$
  N1 $= (new\_ID(), Incoming(q),$
        $Now(q) \cup \{\eta\}, New(q) \cup \{\varphi\}, Next(q) \cup \{\varphi \, U \, \psi\})$
  N2 $= (new\_ID(), Incoming(q),$
        $Now(q) \cup \{\eta\}, New(q) \cup \{\psi\}, Next(q))$
  return *expand*(N2, *expand*(N1, Nodes))
end

...

# Graf LTL formule – funkce expand (switch)

case $(\eta \equiv \varphi \, R \, \psi)$
  N1 $= (new\_ID(), Incoming(q),$
    $Now(q) \cup \{\eta\}, New(q) \cup \{\varphi, \psi\}, Next(q))$
  N2 $= (new\_ID(), Incoming(q),$
    $Now(q) \cup \{\eta\}, New(q) \cup \{\psi\}, Next(q) \cup \{\varphi \, R \, \psi\})$
  return $expand(\text{N2}, expand(\text{N1}, \text{Nodes}))$
end

case $(\eta \equiv \varphi \vee \psi)$
  N1 $= (new\_ID(), Incoming(q),$
    $Now(q) \cup \{\eta\}, New(q) \cup \{\varphi\}, Next(q))$
  N2 $= (new\_ID(), Incoming(q),$
    $Now(q) \cup \{\eta\}, New(q) \cup \{\psi\}, Next(q))$
  return $expand(\text{N2}, expand(\text{N1}, \text{Nodes}))$
end

...

# Graf LTL formule – funkce expand (switch)

case $(\eta \equiv \varphi \wedge \psi)$
  N $= (\mathit{new\_ID}(), \mathit{Incoming}(q),$
    $\mathit{Now}(q) \cup \{\eta\}, \mathit{New}(q) \cup \{\varphi, \psi\}, \mathit{Next}(q))$
  return $\mathit{expand}(\mathsf{N}, \mathsf{Nodes})$
end

case $(\eta \equiv X\,\varphi)$
  N $= (\mathit{new\_ID}(), \mathit{Incoming}(q),$
    $\mathit{Now}(q) \cup \{\eta\}, \mathit{New}(q), \mathit{Next}(q) \cup \{\varphi\})$
  return $\mathit{expand}(\mathsf{N}, \mathsf{Nodes})$
end

end /* end of switch */

# Příklad výpočet grafu pro formuli X(a)

## Vypočtené uzly

| | |
|---|---|
| Id: | 2 |
| Incoming: | init |
| Now: | X(a) |
| New: | ∅ |
| Next: | a |

| | |
|---|---|
| Id: | 4 |
| Incoming: | 2 |
| Now: | a |
| New: | ∅ |
| Next: | ∅ |

| | |
|---|---|
| Id: | 5 |
| Incoming: | 4,5 |
| Now: | ∅ |
| New: | ∅ |
| Next: | ∅ |

## Výpočet

| Id | Incoming | Now | New | Next |
|---|---|---|---|---|
| 1 | init | ∅ | {X(a)} | ∅ |
| 2 | init | {X(a)} | ∅ | {a} |

Uzel 2 je nově vypočtený uzel.

| | | | | |
|---|---|---|---|---|
| 3 | 2 | ∅ | {a} | ∅ |
| 4 | 2 | {a} | ∅ | ∅ |

Uzel 4 je nově vypočtený uzel.

| | | | | |
|---|---|---|---|---|
| 5 | 4 | ∅ | ∅ | ∅ |

Uzel 5 je nově vypočtený uzel.

| | | | | |
|---|---|---|---|---|
| 6 | 5 | ∅ | ∅ | ∅ |

Uzel 6 je shodný s uzlem 5.
Incoming(5) = Incoming(5)∪{5}

# Příklad $a\,U\,(b\,U\,c)$

01| in, ∅, {aU(bUc)}, ∅
02| in, {aU(bUc)}, {a}, {aU(bUc)}
03| in, {aU(bUc)}, {bUc}, ∅
04| in, {aU(bUc),a}, ∅, {aU(bUc)}
Uzel 04 je nově vypočtený uzel.
05| 04, ∅, {aU(bUc)}, ∅
06| 04, {aU(bUc)}, {a}, {aU(bUc)}
07| 04, {aU(bUc)}, {bUc}, ∅
08| 04, {aU(bUc),a}, ∅, {aU(bUc)}
Uzlu 04 je přidán předchůdce 04.

07| 04, {aU(bUc)}, {bUc}, ∅
09| 04, {aU(bUc),bUc}, {b}, {bUc}
10| 04, {aU(bUc),bUc}, {c}, ∅
11| 04, {aU(bUc),bUc,b}, ∅, {bUc}
Uzel 11 je nově vypočtený uzel.
12| 11, ∅, {bUc}, ∅
13| 11, {bUc}, {b}, {bUc}
14| 11, {bUc}, {c}, ∅
15| 11, {bUc,b}, ∅, {bUc}
Uzel 15 je nově vypočtený uzel.
16| 15, ∅ {bUc}, ∅
17| 15, {bUc}, {b}, {bUc}
18| 15, {bUc}, {c}, ∅
19| 15, {bUc,b}, ∅, {bUc}
Uzlu 15 je přidán předchůdce 15.

# Příklad $a\,U\,(b\,U\,c)$ – pokračování

18| 15, {bUc}, {c}, ∅
20| 15, {bUc,c}, ∅, ∅
Uzel 20 je nově vypočtený uzel.
21| 20, ∅, ∅, ∅
Uzel 21 je nově vypočtený uzel.
22| 21, ∅, ∅, ∅
Uzlu 21 je přidán předchůdce 21.

14| 11, {bUc}, {c}, ∅
23| 11, {bUc,c}, ∅, ∅
Uzlu 20 je přidán předchůdce 11.

10| 04, {aU(bUc),bUc}, {c}, ∅
24| 04, {aU(bUc),bUc,c}, ∅, ∅
Uzel 24 je nově vypočtený uzel.
25| 24, ∅, ∅, ∅
Uzlu 21 je přidán předchůdce 24.

# Příklad $a\,U\,(b\,U\,c)$ – pokračování

03| in, {aU(bUc)}, {bUc}, ∅
26| in, {aU(bUc),bUc}, {b}, {bUc}  27| in, {aU(bUc),bUc}, {c}, ∅
28| in, {aU(bUc),bUc,b}, ∅, {bUc}
Uzlu 11 je přidán předchůdce *in*.

27| in, {aU(bUc),bUc}, {c}, ∅
29| in, {aU(bUc),bUc,c}, ∅, ∅
Uzlu 24 je přidán předchůdce *in*.

# Převedení grafu na zobecněný Büchi automat

**Předpoklady**

- Dána množina AP.
- *Nodes* je množina vrcholů grafu LTL formule.

**Zobecněný Büchi automat** $A = (S, \Sigma, \delta, \mathit{init}, \mathcal{F})$

- $S = \mathit{Nodes} \cup \{\mathit{init}\}$
- $\Sigma = 2^{AP}$
- $r' \in \delta(r, \alpha)$ pokud
  - $r \in \mathit{Incoming}(r')$, $\alpha \in \Sigma$
  - $\alpha$ splňuje omezení dané množinou $((AP \cup \neg AP) \cap \mathit{Now}(r'))$
- $\mathcal{F} = \{F_1, \ldots, F_n\}$
  - Pro každou podformuli ve tvaru $\varphi\, U\, \psi$ definujeme $F_i$.
  - $F_i = \{r \in \mathit{Nodes} \mid \psi \in \mathit{Now}(r) \vee \varphi U \psi \notin \mathit{Now}(r)\}$.

# Příklad $a\,U\,(b\,U\,c)$ – Zobecněný Büchi automat

Přechodová funkce (stráže)

| | a | b | c | tt |
|---|---|---|---|---|
| → init | 04 | 11 | 24 | |
| 04 | 04 | 11 | 24 | |
| 11 | | 15 | 20 | |
| 15 | | 15 | 20 | |
| 20 | | | | 21 |
| 21 | | | | 21 |
| 24 | | | | 21 |

$\mathcal{F}$ – akceptující množiny

- $\mathsf{F}_{aU(bUc)}$ — $\{11,15,20,21,24\}$
- $\mathsf{F}_{bUc}$ — $\{04,20,21,24\}$