Temporálne logiky

propozičná logika vs. logika prvého rádu
globálna vs. kompozičná
vetviaci sa čas vs. lineárny čas
časové body vs. časové intervaly
diskrétny čas vs. spojitý čas
minulosť vs. budúcnosť
distribovanosť vs. lokálnosť
....

Temporálne logiky

- časová os:
  - dvojica (S, $<$), kde $<$ je úplné usporiadanie
  - izomorfná s (N, $<$)

- čas:
  - diskrétny
  - počiatočný okamih
  - nekonečný

- AP: atomické propozície (ozn. P, Q, …)

- štruktúra lineárneho času: trojica $M = (S, x, L)$, kde

  S – množina stavov

  x: N $\rightarrow$ S – postupnosť stavov

  L: $S \rightarrow \boldsymbol{P}(AP)$ – určuje pre daný stav, ktoré atomické propozície v tomto stave platia (teda ktoré AP sú true)

Temporálne logiky

- označenie:
  - postupnosť stavov $x = s_0, s_1, s_2, s_3, \ldots$
  - definujeme $x^i = s_i, s_{i+1}, s_{i+2}, s_{i+3}, \ldots$ (teda $x = x^0$)

- základné temporálne operátory:

  $\Diamond p$ – eventually p (raz určite p)(textovo Fp)

  $\Box p$ – vždy p (textovo Gp)

  $\bigcirc p$ – nasledujúci krát p (textovo Xp)

  $p \cup q$ – p until q (raz q začne platiť, a do vtedy platí p)

Temporálne logiky

Syntax: množina PLTL formúl je definovaná ako najmenšia množina generovaná nasledujúcimi pravidlami:

- každá *AP* (atomická propozícia) *P* je formula
- ak *p*, *q* sú formuly, tak $p \wedge q$, $\neg p$ sú formuly
- ak *p*, *q* sú formuly, tak *p* U *q*, X*p* sú formuly

zavedieme označenia:

Fp = true U p	$\Diamond p$	
Gp = $\neg$F$\neg p$	$\Box p$	
F$^{\infty}p$ = GFp		(nekonečne veľa krát)
G$^{\infty}p$ = FGp		(skoro všade)

Temporálne logiky

Sémantika:

$x \models P$ iff $P \in L(s_0)$ pre atomickú propozíciu P
$x \models p \wedge q$ iff platí $x \models p$ a $x \models q$
$x \models \neg p$ iff neplatí $x \models p$
$x \models (p \text{ U } q)$ iff $\exists j\ (x^j \models q$ and $\forall k < j: x^k \models p)$
$x \models \text{X}p$ iff $x^1 \models p$
$x \models \text{F}p$ iff $\exists j\ (x^j \models p)$
$x \models \text{G}p$ iff $\forall j\ (x^j \models p)$
$x \models \text{F}^\infty p$ iff $\forall k\ \exists j \geq k\ (x^j \models p)$
$x \models \text{G}^\infty p$ iff $\exists k\ \forall j > k\ (x^j \models p)$

Príklady:

$p \Rightarrow \text{F}q$	ak p platí teraz, tak raz bude platiť q
$\text{G}(p \Rightarrow \text{F}q)$	vždy keď platí p, tak raz začne platiť aj q
$p \wedge \text{G}(p \Rightarrow \text{X}p) \Rightarrow \text{G}p$	temporálna formulácia indukcie

Temporálne logiky

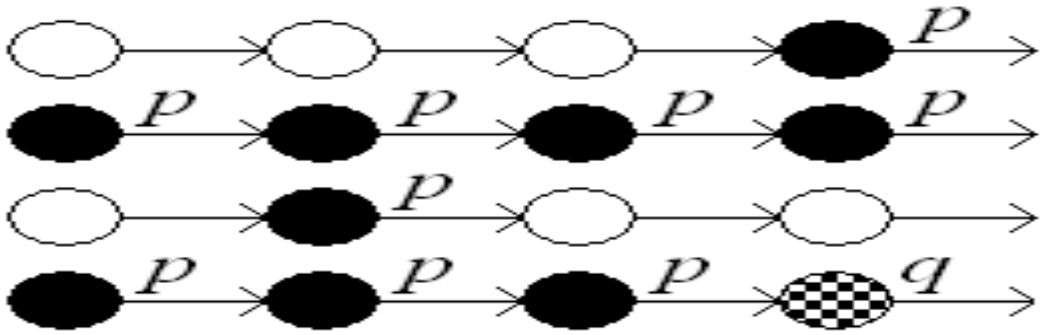
Príklady použitia operátorov F, G, X a U v propozičnej lineárnej temporálnej logiky

F*p*

G*p*

X*p*

p U *q*



Temporálne logiky

Variácie

Slabé until
Silné p U q sa zvykne označovať ako p U_s q alebo p $U_\exists$ q

$x \models p\ U_\forall\ q$ iff $\forall\ j\ ((\forall k \leq j, x^k \models \neg q) \Rightarrow x^j \models p)$

t.j. môže sa stať, že q nebude platiť nikdy

p $U_\exists$ q ..... p $U_\forall$ q $\wedge$ Fq
p $U_\forall$ q .... p $U_\exists$ q $\vee$ Gp

Temporálne logiky

Miesto (N, <) zoberiem len podmnožinu I
(môže byť aj konečná)

Gp	pre všetky nasledujúce stavy v I platí p
Fp	pre nejaký stav v I platí p
$X_{\forall}p$	(weak nextime) ak existuje následný stav v I tak v ňom platí p
$X_{\exists}p$	(strong nextime) existuje následný stav v I a v ňom platí p

$X_{\exists}p$ ..... $\neg X_{\forall} \neg p$

$X_{\forall}p$ ..... $\neg X_{\exists} \neg p$

Temporálne logiky

$G^{-}p$	vždy v minulosti platilo p
$F^{-}p$	niekedy v minulosti platilo p
$X^{-}p$	naposledy platilo p
$p\ U^{-}q$	niekedy platilo q a odvtedy platí q

Temporálne logiky

Branching (time) tempopral logic

- temporálna štruktúra: trojica $M = (S, R, L)$, kde

  S – množina stavov

  $R \subseteq S \times S$ taká, že $\forall s \in S \; \exists t \in S \; (s, t) \in R$

  $L: S \rightarrow \boldsymbol{P}(AP)$ – určuje pre daný stav, ktoré atomické propozície v tomto stave platia (teda ktoré AP sú true)

- M možno chápať ako značkovaný orientovaný graf s vrcholmi S, hranami danými R a vrcholy majú značky dané L

- Hovoríme, že M je
  - *acyklický*, ak nemá orientované cykly
  - *stromová štruktúra*, ak každý vrchol má nanajvýš jedného predchodcu
  - *strom*, ak je stromová štruktúra a má koreň
- označenie:
  - plná cesta $x = (s_0, s_1, s_2, s_3, \ldots)$: pre $\forall i$: $(s_i, s_{i+1}) \in R$
  - definujeme $x^i = (s_i, s_{i+1}, s_{i+2}, s_{i+3}, \ldots)$

Temporálne logiky

- CTL (Computational Tree Logic)
- CTL* (Full Branching Time Logic) – silnejšia ako CTL, historicky mladšia; najprv popíšeme CTL*

- Syntax: (*stavové* a *path* („cestové“) formuly)

  - každá atomická propozícia je stavová formula (S1)
  - ak p, q sú stavové formuly, tak $p \wedge q$, $\neg p$ sú stavové formuly (S2)
  - ak p je path formula, tak Ep, Ap sú stavové formuly (S3)

  - každá stavová formula je aj path formula (P1)
  - ak p, q sú path formuly, tak potom aj $p \wedge q$, $\neg p$ sú path formuly (P2)
  - ak p, q sú path formuly, tak potom aj p U q, Xp sú path formu (P3)

Temporálne logiky

CTL* tvoria stavové formuly

CTL tvoria pravidlá (S1), (S2), (S3) a pravidlo (P0):

ak *p*, *q* sú stavové formuly, tak *p* U *q*, X*p* sú stavové formuly (P0)

CTL operátory:

A – pre všetky budúcnosti
E – existuje budúcnosť

za A alebo E vždy nasleduje jeden z operátorov G, F, X, U
dá sa ukázať, že CTL* má väčšiu vyjadrovaciu silu než CTL

Temporálne logiky

Sémantika: (pre CTL*)

(S1) $M, s_0 \models p$ iff $p \in L(s_0)$

(S2) $M, s_0 \models p \wedge q$ iff platí $M, s_0 \models p$ a $M, s_0 \models q$

$M, s_0 \models \neg p$ iff neplatí $M, s_0 \models p$

(S3) $M, s_0 \models \mathrm{E}p$ iff $\exists x$ v M tak, že $M, x \models p$

$M, s_0 \models \mathrm{A}p$ iff pre $\forall x$ v M platí $M, x \models p$

(P1) $M, x \models p$ iff $M, s_0 \models p$

(P2) $M, x \models p \wedge q$ iff platí $M, x \models p$ and $M, x \models q$

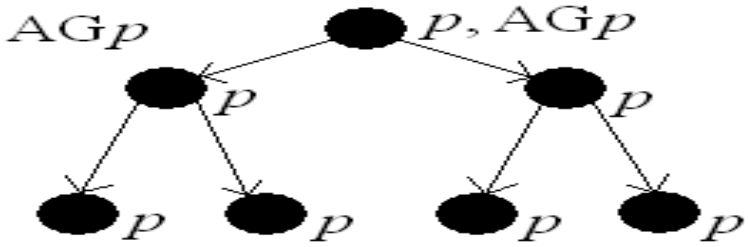
$M, x \models \neg p$ iff neplatí $M, x \models p$

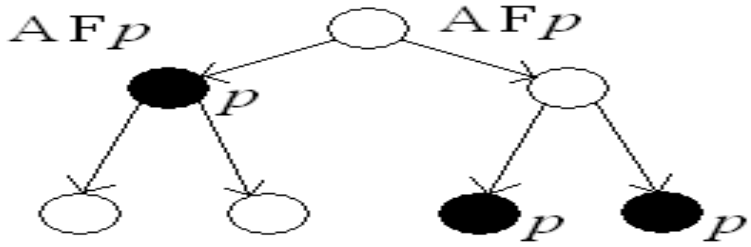
(P3) $M, x \models (p \mathrm{U} q)$ iff $\exists j$ ($M, x^j \models q$ a $\forall k$ ($k < j$ implikuje $M, x^k \models p$)

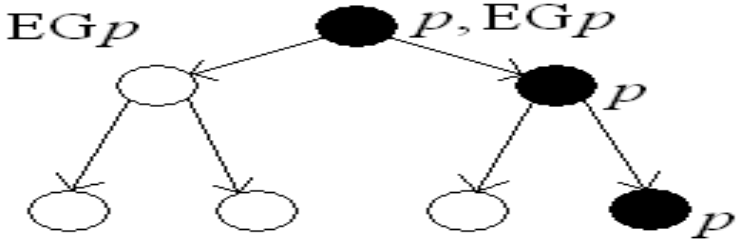
$M, x \models \mathrm{X}p$ iff $M, x^1 \models p$

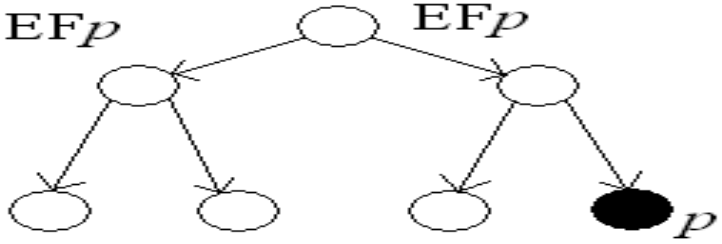
Temporálne logiky

Príklady použitia operátorov A, E v propozičnej branching temporal logic (značky sú pri vrcholoch; vrchol je vyplnený, ak má značku *p*)









Temporálne logiky

- množina schém axióm
- množina odvodzovacích (inferenčných) pravidiel

formula *p* je *dokázateľná* (zapisujeme |- *p*), ak pre ňu existuje *dôkaz*, t.j. konečná postupnosť formúl taká, že na jej konci je *p* a každá formula v nej je buď prípad axiómy alebo vyplýva z predchádzajúcich použitím nejakého odvodzovacieho pravidla

Temporálne logiky

Schémy axióm:

tautológie propozičnej logiky	(Ax1)
$\mathrm{EF}p \equiv \mathrm{E}(\mathrm{true}\ \mathrm{U}\ p)$	(Ax2)
$\mathrm{AG}p \equiv \neg\mathrm{EF}\neg p$	(Ax2’)
$\mathrm{AF}p \equiv \mathrm{A}(\mathrm{true}\ \mathrm{U}\ p)$	(Ax3)
$\mathrm{EG}p \equiv \neg\mathrm{AF}\neg p$	(Ax3’)
$\mathrm{EX}(p \vee q) \equiv \mathrm{EX}p \vee \mathrm{EX}q$	(Ax4)
$\mathrm{AX}p \equiv \neg\mathrm{EX}\neg p$	(Ax5)
$\mathrm{E}(p\ \mathrm{U}\ q) \equiv q \vee (p \wedge \mathrm{EXE}(p\ \mathrm{U}\ q))$	(Ax6)
$\mathrm{A}(p\ \mathrm{U}\ q) \equiv q \vee (p \wedge \mathrm{AXA}(p\ \mathrm{U}\ q))$	(Ax7)
$\mathrm{EX}\ \mathrm{true} \wedge \mathrm{AX}\ \mathrm{true}$	(Ax8)
$\mathrm{AG}(r \Rightarrow (\neg q \wedge \mathrm{EX}r)) \Rightarrow (r \Rightarrow \neg\mathrm{A}(p\ \mathrm{U}\ q))$	(Ax9)
$\mathrm{AG}(r \Rightarrow (\neg q \wedge \mathrm{EX}r)) \Rightarrow (r \Rightarrow \neg\mathrm{AF}q)$	(Ax9’)
$\mathrm{AG}(r \Rightarrow (\neg q \wedge (p \Rightarrow \mathrm{AX}r))) \Rightarrow (r \Rightarrow \neg\mathrm{E}(p\ \mathrm{U}\ q))$	(Ax10)
$\mathrm{AG}(r \Rightarrow (\neg q \wedge \mathrm{AX}r)) \Rightarrow (r \Rightarrow \neg\mathrm{EF}q)$	(Ax10’)
$\mathrm{AG}(p \Rightarrow q) \Rightarrow (\mathrm{EX}p \Rightarrow \mathrm{EX}q)$	(Ax11)

Temporálne logiky

Odvodzovacie pravidlá:

ak |- p, tak potom |- AGp (R1, zovšeobecnenie)

ak |- p a |- $p \Rightarrow q$, tak potom |- q (R2, modus ponens)

Veta: Deduktívny systém s axiómami (Ax1)–(Ax11) a odvodzovacími pravidlami (R1), (R2) je sound (zdravý, korektný) and complete (úplný) pre CTL.

Vzťah CTL a MU-kalkulus

A (p U q) .... $\mu Z \;\; q \vee (p \wedge (\langle\rangle tt \wedge [] Z))$

E (p U q) ..... $\mu Z \;\; q \vee (p \wedge \;\; \langle\rangle Z)$

AF p ..... $\mu Z \;\; p \vee (\langle\rangle tt \wedge [] Z)$

EF p ..... $\mu Z \;\; p \vee \langle\rangle Z$

Verifikácia vlastností systémov vyjadrených pomocou logických formúl

1.theorem proving

axiomy logiky+vlastnosti systému/programu |- theorema

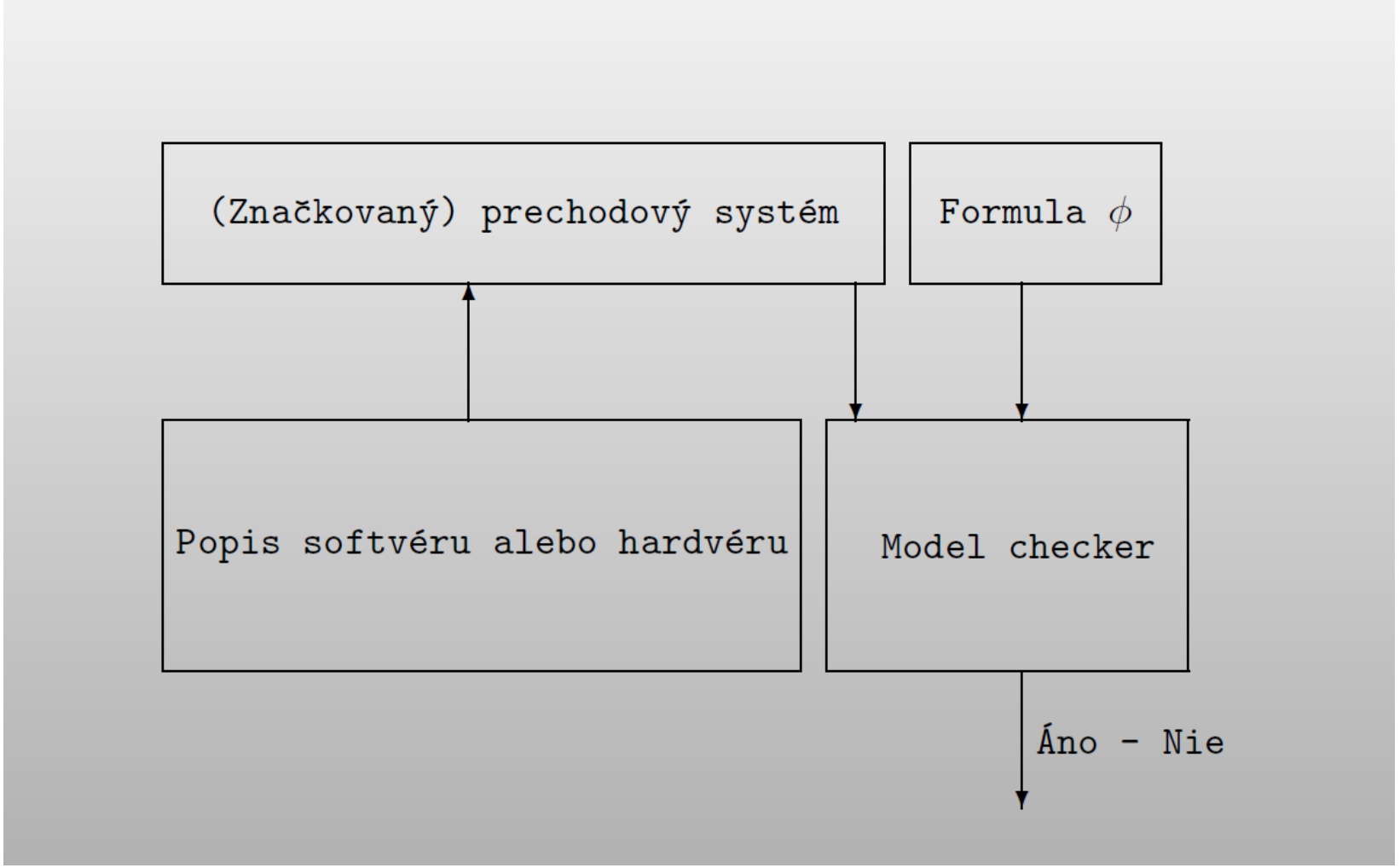
- automatický alebo poloautomatický theorem proving alebo proof verification

- v závislosti od odpovedajúcej logiky sa zložitosť riešenia pohybuje od triviálneho až po nemožné.

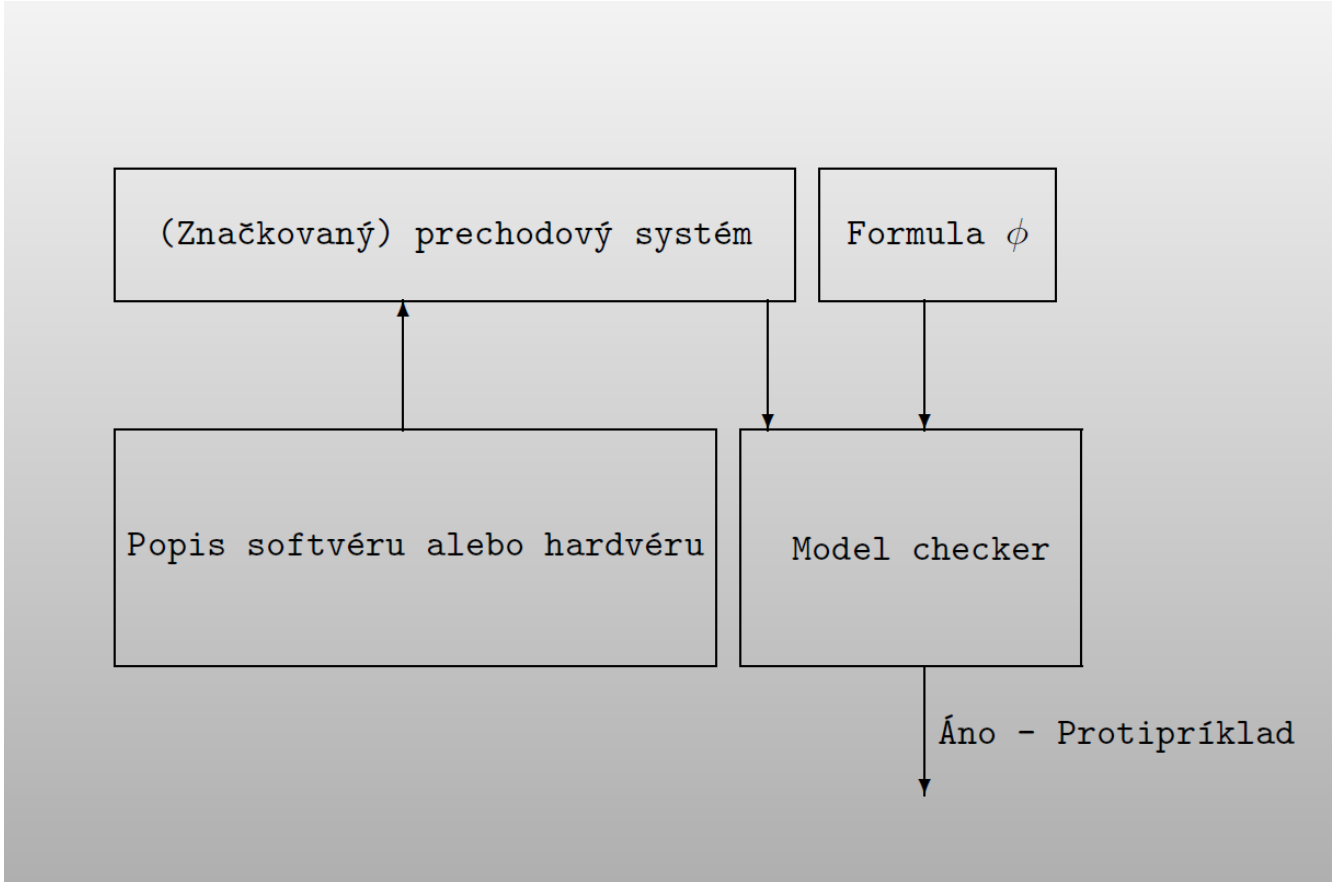
Praktické využitie: návrh a verifikácia integrovaných obvodov, (verifikácia aritmetických operácií, ...) Existuje množstvo softvérových nástrojov.

2. model checking

Model Checking



Model Checking



Úloha: daný (značkovaný) prechodový systém M (Kripkeho štruktúra) a formula temporálnej alebo modálnej logiky φ, úlohou je nájsť všetky stavy M také, že

$M,s \models \varphi$

Otcovia zakladatelia E. M. Clarke, E. A. Emerson, and J. Sifakis (roku 2007 Turingová cena za ich práce v tejto oblasti)

Model checking problem – prehľadávanie grafu - vrcholy = stavy

"state explosion" problém

- symbolické algoritmy nevytvárať graf explicitne , ale reprezentovať ho implicitne napr. pomocou formúl propozičnej logiky

- ohraničený model checking algoritmy - vytvorenie ZPS je obmedzene na k krokov a skúma sa, či neprestane platiť formula, vhodné pre niektoré modely

- "partial order reduction" redukuje sa počet skúmaných prelínaní konkurentných procesov - nemá význam skúmať zvlášť všetky možnosti ak neovplyvňujú platnosť formuly

- abstrakcia (abstract interpretation) - zjednodušenie modelu – ten spravidla nespĺňa rovnaké vlastnosti, ďalšie zjemňovanie je možné, zdravá abstrakcia - vlastnosti vyhovujú aj pôvodnému systému, úplnosť spravidla neplatí

- protipríkladom riadená abstrakcia - vytvoríme abstraktnejší modela keď nájdeme protipríklad tak zisťujeme či odpovedá modelu alebo vznikol zlou abstrakciou, v prvom prípade výsledok beriem ako výsledok. Ak nenájdeme protipríklad zjemníme model- ....

Temporálne logiky

Model Checking

daná štruktúra M a formula p
Úloha: zistiť, či M je modelom pre p

Branching–Time–Logic–Model–Checking

daná konečná štruktúra $M = (S, R, L)$ a BTL formula p

Úloha: pre každý stav $s \in S$ určiť, či platí $M, s \models p$ a ak áno, s sa označí značkou "p"

Temporálne logiky

Príklad: CTL model checking prer AFp:

predpokladajme, že už máme vrcholy, pre ktoré platí p už "označené"

ideme označovať tie, pre ktoré platí AF*p*:

1. ak je vrchol označený *p*, tak ho označíme aj AF*p*

2. (opakuje sa, kým sa dá) označ vrchol s AF*p*, ak všetci jeho následníci sú označení AF*p*
3. označ ¬AF*p* tie, ktoré nie sú označené AF*p*