Metoda tableauxa za modalnu logiku

Babić, Tin

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Tin Babić

METODA TABLEAUXA ZA MODALNU LOGIKU

Diplomski rad

Zagreb, studeni, 2018.

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Tin Babić

METODA TABLEAUXA ZA MODALNU LOGIKU

Diplomski rad

Voditelji rada:
Doc. dr. sc.
Tin Perkov

Izv. prof. dr. sc.
Mladen Vuković

Zagreb, studeni, 2018.

Ovaj diplomski rad obranjen je dana ____________________ pred ispitnim povjerenstvom u sastavu:

1. ______________________________, predsjednik
2. ______________________________, član
3. ______________________________, član

Povjerenstvo je rad ocijenilo ocjenom ________________.

Potpisi članova povjerenstva:

1. ______________________________
2. ______________________________
3. ______________________________

Mojoj majci.

Sadržaj

Uvod

Modalna logika svoje korijene vuče još iz doba stare Grčke. Naznake današnjeg shvaćanja počela je dobivati 1918. godine radom američkog filozofa i logičara C. I. Lewisa.[1] U svom djelu [7], Lewis je opisao svoje viđenje problema materijalne implikacije i uveo pojam stroge implikacije, koju je zadao korištenjem modalnog operatora $\Box$.

Nešto kasnije, 1933. godine, Kurt Gödel[2] je u svom djelu [5], formalizirao pojam dokazivosti također korištenjem modalnog operatora $\Box$. Time je omogućio svođenje intuicionističke logike na logiku sudova i modalnih operatora.

Nastavlja se veoma intenzivno razvijanje sve do 1959. godine kada Saul Kripke[3] u svom djelu [6] daje novu semantiku modalne logike i dokazuje teorem potpunosti. Ova semantika koristi se i danas.

Glavni zamah modalna logika dobiva razvojem računarstva. Javila se potreba za verifikacijom programa i programskih jezika. Ovakvi problemi se mogu na jednostavan način opisati modalnim operatorima. Najvažniji problem u modalnoj logici s računarskog aspekta jest ispunjivost neke formule.

Ispunjivost je usko vezana uz pojmove adekvatnosti i potpunosti, a tradicionalno se dokazuje aksiomatskim putem. Ovakav pristup pokazali su Blackburn[4] i ostali u [1]. Međutim, za računala ovakav način dokazivanja nije prikladno rješenje.

Ovaj rad usredotočit će se na alternativni način dokazivanja ispunjivosti formula modalne logike, a to je semantički tableaux ili glavni test. Ovaj način bliži je algoritmu koji bi računala mogla izvršavati i čovjeku je intuitivno jasniji. Osim što odlučuje je li dana formula ispunjiva, iz tableaua se, ako je odgovor potvrdan, može očitati i model na kojem je formula ispunjena.

Postoje razne verzije tableaux metode za modalnu logiku, npr. Rajeeva Gorea [3] i Valentina Goranka [4], na kojoj se i temelji ovaj rad.

Rad je podjeljen u četiri poglavlja.

[1] Clarence Irving Lewis (1883. – 1964.). Američki logičar i filozof.
[2] Kurt Gödel (1906. – 1978.). Austrijski logičar.
[3] Saul Aaron Kripke (rođen 1940.). Američki logičar i filozof.
[4] Patrick Blackburn (rođen 1959.). Novozelandski logičar i matematičar.

Prvo poglavlje iznosi osnovne činjenice o sintaksi i semantici osnovnog modalnog jezika. Definiraju se alfabet, formule, istinitost, ispunjivost i valjanost.

Drugo poglavlje iznosi osnovne činjenice o tableaux metodi za logiku sudova, pravila redukcije, pojam stabla i Hintikkinog[5] skupa. Poglavlje završava dokazom adekvatnosti i potpunosti te metode.

Treće poglavlje proširuje do tada definirane pojmove na osnovni modalni jezik. Iznose se pravila redukcije za osnovni modalni jezik, definira se pojam Hintikkinih struktura i označenog grafa. Poglavlje završava dokazom adekvatnosti i potpunosti dane metode za osnovni modalni jezik.

Četvrto poglavlje obrađuje temu odlučivosti. Definira se pojam Turingovog[6] stroja i Turing-odlučivosti. Naposljetku se pokazuje kako su problemi koje rješavaju obje metode predstavljene u ovom radu odlučivi.

[5] Kaarlo Jaakko Juhani Hintikka (1929. – 2015.). Finski filozof i logičar.

[6] Alan Turing (1912. – 1954.). Engleski matematičar, logičar i filozof.

Poglavlje 1

Osnovni modalni jezik

1.1 Sintaksa

Kako bismo mogli uopće nešto zaključivati o modalnoj logici potrebno je razviti jezik kojim ćemo se služiti u nastavku rada. Jezik koji koristimo u ovom radu se naziva **osnovni modalni jezik** i zadaje se svojim **alfabetom**, tj. znakovima koje koristi i nizovima znakova koje za taj jezik imaju posebno značenje - **formulama**.

Definicija 1.1.1. ***Alfabet*** *osnovnog modalnog jezika je unija skupova* ***PROP****,* ***VEZ****,* ***POM*** *i* ***KON*** *pri čemu su:*

PROP = $\{p_0, p_1, p_2 ...\}$ *skup prebrojivo mnogo elemenata koje zovemo* ***propozicionalne varijable***
VEZ = $\{\wedge, \vee, \neg, \rightarrow, \Diamond, \Box\}$ *skup logičkih veznika i modalnih operatora*
POM = $\{(,)\}$ *skup pomoćnih simbola*
KON = $\{\top, \bot\}$ *skup logičkih konstanti*

Vidimo da se alfabet uvelike podudara sa alfabetom logike sudova, s dodatkom modalnih operatora $\Diamond$ i $\Box$.

Modalni operator $\Diamond$ se najčešće čita kao "moguće je" i nije po svojem značenju običan veznik na kakve smo navikli u logici sudova i logici prvog reda, zato ga i drugačije zovemo.

Postoji i njegov dual $\Box$, koji čitamo "nužno je", a veza operatora $\Diamond$ i $\Box$ je

$$\Box = \neg\Diamond\neg$$

Pravo značenje tih dvaju operatora saznat ćemo u sljedećem odjeljku kada bude riječi o semantici.

Koristimo i dvije logičke konstante: $\top$ i $\bot$. Služe nam kao pokrate za formulu koja je uvijek istinita ($\top$), i formulu koja nije nikad istinita ($\bot$). O istinitosti formula više će riječi biti u sljedećem odjeljku.

Za potpunu definiciju jezika preostaje nam definirati i formule jer ne želimo baš bilo kakav niz znakova našeg alfabeta smatrati formulom sa značenjem.

Definicija 1.1.2. ***Formula*** *φ osnovnog modalnog jezika se zadaje rekurzivno:*

$$\varphi = p \mid \top \mid \bot \mid (\neg\varphi) \mid (\varphi_1 \wedge \varphi_2) \mid (\varphi_1 \vee \varphi_2) \mid (\varphi_1 \rightarrow \varphi_2) \mid (\Diamond\varphi) \mid (\Box\varphi)\,,$$

*gdje je $p \in$ **PROP**.*

Uočimo da se nigdje ne spominje veznik $\leftrightarrow$, niti formula $\varphi_1 \leftrightarrow \varphi_2$. Smatramo ju pokratom za $(\varphi_1 \rightarrow \varphi_2) \wedge (\varphi_2 \rightarrow \varphi_1)$.

U daljnjem tekstu vanjske zagrade formula ćemo najčešće ispuštati. Ispuštanje nam omogućuje i uvođenje **prioriteta veznika**.

Najviši prioritet imaju $\Diamond$, $\Box$ i $\neg$, zatim $\wedge$ i $\vee$, a najniži prioritet ima $\rightarrow$.

Ponekad radi veće jasnoće ipak pišemo zagrade i kada ih po prioritetima nije nužno pisati.

Primjer 1.1.3. *Formulu $(((p \wedge q) \wedge r) \rightarrow p)$ često pišemo $(p \wedge q \wedge r) \rightarrow p$ iako bismo je po prioritetima mogli pisati i $p \wedge q \wedge r \rightarrow p$.*

Naravno, zagrade moramo pisati kada je u formuli redoslijed primjene veznika i operatora suprotan prioritetima.

Primjer 1.1.4. *U formuli $\Diamond p \wedge q$ operator $\Diamond$ djeluje samo na p, a u formuli $\Diamond(p \wedge q)$ djeluje na $p \wedge q$.*

Potreban nam je i način klasificiranja formula po broju veznika. Za modalnu logiku i naš rad značajni će biti modalni operatori.

Definicija 1.1.5. ***Modalna dubina*** *formule φ, u oznaci $md(\varphi)$, definira se rekurzivno:*

$$\begin{aligned} md(p) = md(\top) = md(\bot) &= 0 \\ md(\neg\varphi) &= md(\varphi) \\ md(\varphi_1 \wedge \varphi_2) = md(\varphi_1 \vee \varphi_2) = md(\varphi_1 \rightarrow \varphi_2) &= max(md(\varphi_1), md(\varphi_2)) \\ md(\Diamond\varphi) = md(\Box\varphi) &= 1 + md(\varphi) \end{aligned}$$

Intuitivno, modalnu dubinu možemo shvatiti kao maksimalan broj ugnježđenih modalnih operatora u formuli φ (vidi [4]).

Primjer 1.1.6. *Modalna dubina formule $\Box\Diamond(\Diamond p \wedge \Box\Box q)$ je 4.*

Još jedan način klasifikacije formula kojeg ćemo kasnije koristiti u dokazima je i **duljina** formule.

Definicija 1.1.7. ***Duljina** formule φ u oznaci $len(\varphi)$, definira se rekurzivno:*

$$
\begin{aligned}
len(p) = len(\top) = len(\bot) &= 1 \\
len(\neg\varphi) = len(\Box\varphi) = len(\Diamond\varphi) &= 1 + len(\varphi) \\
len(\varphi_1 \wedge \varphi_2) = len(\varphi_1 \vee \varphi_2) = len(\varphi_1 \to \varphi_2) &= 1 + len(\varphi_1) + len(\varphi_2)
\end{aligned}
$$

Primjer 1.1.8. *Duljina formule $\Box\Diamond(\Diamond p \wedge \Box\Box q)$ je 8.*

1.2 Semantika

U proučavanju logike redovito nas zanimaju pitanja kao što su **istinitost** i **ispunjivost** dane formule. U ovom odjeljku definirat ćemo što znači pojam istinitosti i ispunjivosti u osnovnom modalnom jeziku. Također, definirat ćemo i dualni pojam ispunjivosti, a to je **valjanost** dane formule.

Semantika svih modalnih logika i njihovih jezika zasniva se na proučavanju relacijskih struktura odnosno Kripkeovih okvira.

Definicija 1.2.1. ***Kripkeov okvir** ili **okvir** je uređeni par $\mathfrak{F} = (W, R)$ gdje je W neprazan skup čije elemente zovemo **svjetovi**, a $R \subseteq W \times W$ binarna relacija. Kažemo da je svijet s **dostiživ** iz svijeta t ako je $(t, s) \in R$. Relaciju R zovemo **relacija dostiživosti**.*

Uređene parove $(t, s) \in R$ ćemo u daljnjem tekstu označavati kao tRs. Sama relacijska struktura nam i dalje ništa ne govori o istinitosti formula. Ovu vezu dat će nam iduća definicija.

Definicija 1.2.2. ***Kripkeov model** ili **model** je uređen par $\mathfrak{M} = (\mathfrak{F}, V)$ gdje je $\mathfrak{F}$ okvir a $V : \mathbf{PROP} \to 2^W$ funkcija koju zovemo **valuacija**. Pritom kažemo da je $\mathfrak{M}$ **baziran na** $\mathfrak{F}$.*

Intuitivno nam valuacija daje skup svjetova u kojima određenu propozicionalnu varijablu smatramo istinitom.

Sada ćemo i formalno definirati pojam istinitosti.

Definicija 1.2.3. *Neka je w jedan od svjetova modela $\mathfrak{M}$. Rekurzivno definiramo pojam* ***istinitosti*** *formule φ* ***u svijetu*** *w u oznaci $\mathfrak{M}, w \Vdash \varphi$, ovako:*

$$
\begin{aligned}
\mathfrak{M}, w \Vdash p &\Leftrightarrow w \in V(p) \\
\mathfrak{M}, w \Vdash \top &\textit{ uvijek} \\
\mathfrak{M}, w \Vdash \bot &\textit{ nikad} \\
\mathfrak{M}, w \Vdash \neg\varphi &\Leftrightarrow \mathfrak{M}, w \nVdash \varphi \\
\mathfrak{M}, w \Vdash \varphi_1 \wedge \varphi_2 &\Leftrightarrow \mathfrak{M}, w \Vdash \varphi_1 \textit{ i } \mathfrak{M}, w \Vdash \varphi_2 \\
\mathfrak{M}, w \Vdash \varphi_1 \vee \varphi_2 &\Leftrightarrow \mathfrak{M}, w \Vdash \varphi_1 \textit{ ili } \mathfrak{M}, w \Vdash \varphi_2 \\
\mathfrak{M}, w \Vdash \varphi_1 \rightarrow \varphi_2 &\Leftrightarrow \mathfrak{M}, w \nVdash \varphi_1 \textit{ ili } \mathfrak{M}, w \Vdash \varphi_2 \\
\mathfrak{M}, w \Vdash \Diamond\varphi &\Leftrightarrow \exists w' \in W\ wRw' \textit{ i } \mathfrak{M}, w' \Vdash \varphi \\
\mathfrak{M}, w \Vdash \Box\varphi &\Leftrightarrow \forall w' \in W\ wRw' \textit{ vrijedi } \mathfrak{M}, w' \Vdash \varphi
\end{aligned}
$$

Pojam svijeta nije nam trebao u logici sudova i logici prvog reda. Istinitost modalnih operatora zahtjeva taj pojam dok se ostali otprije poznati veznici prirodno proširuju. Upravo zbog toga smo u prošlom odjeljku naglasili njihovu posebnost.

Kada jednom znamo što znači istinita formula u nekom svijetu, na prirodan se način definiraju i općenitije formulacije istinitosti.

Definicija 1.2.4. *Kažemo da je formula φ* ***ispunjiva u modelu*** *$\mathfrak{M}$ ako je istinita u nekom svijetu modela $\mathfrak{M}$.*
Kažemo da je skup formula Γ ***ispunjiv u modelu****, u oznaci $\mathfrak{M}, w \Vdash \Gamma$, ako postoji svijet w modela $\mathfrak{M}$ u kojem su sve formule iz Γ istinite.*
Kažemo da je formula φ ***ispunjiva*** *ako je ispunjiva u nekom modelu $\mathfrak{M}$.*
Kažemo da je skup formula Γ ***ispunjiv*** *ako je ispunjiv u nekom modelu $\mathfrak{M}$.*
Kažemo da je formula φ ***ispunjiva u okviru*** *$\mathfrak{F}$ ako za neku valuaciju V i neki svijet $w \in \mathfrak{F}$ vrijedi $(\mathfrak{F}, V), w \Vdash \varphi$.*

Nakon ispunjivosti, ostaje nam definirati njezin dual, **valjanost**.

Definicija 1.2.5. *Kažemo da je formula φ* ***globalno istinita na modelu*** *$\mathfrak{M}$, u oznaci $\mathfrak{M} \vDash \varphi$, ako je istinita u svakom svijetu modela $\mathfrak{M}$.*
Kažemo da je formula φ ***valjana na okviru*** *$\mathfrak{F}$, u oznaci $\mathfrak{F} \vDash \varphi$, ako je globalno istinita na svakom modelu $\mathfrak{M}$ baziranom na okviru $\mathfrak{F}$.*
Kažemo da je formula φ ***valjana*** *ako je valjana na svakom okviru $\mathfrak{F}$.*

Sve definicije do sada su se odnosile na jednu formulu ili skup formula. Međutim, ponekad je potrebno gledati odnos istinitosti neke formule i skupa. Za to nam služi relacija **logičke posljedice**.

Definicija 1.2.6. *Kažemo da je formula φ **logička posljedica** skupa formula Γ, u oznaci $\Gamma \Vdash \varphi$, ako za svaki model $\mathfrak{M}$ i za svaki svijet w iz modela $\mathfrak{M}$ vrijedi :*

$$\text{ako za sve } \rho \in \Gamma \text{ vrijedi } \mathfrak{M}, w \Vdash \rho \text{ tada vrijedi i } \mathfrak{M}, w \Vdash \varphi.$$

*Još kažemo kako skup formula Γ **logički povlači** formulu φ.*

Definirali smo dva veoma važna semantička pojma, ispunjivost i valjanost. Željeli bismo efikasan postupak dobivanja odgovora na dva problema:

1. za danu formulu φ osnovnog modalnog jezika pronaći model u kojem je ona ispunjiva;

2. za danu formulu φ osnovnog modalnog jezika vidjeti je li valjana.

Ova dva problema se mogu svesti jedan na drugi vrlo prirodno:

Napomena 1.2.7. *Formula φ je valjana ako i samo ako formula $\neg\varphi$ nije ispunjiva, odnosno φ je ispunjiva ako i samo ako $\neg\varphi$ nije valjana.*

U nastavku rada definirat ćemo postupak dobivanja modela u kojem je dana formula ispunjiva. Prvo ćemo pokazati jednostavniju inačicu koja se koristi u logici sudova, a zatim krećemo na osnovni modalni jezik.

Poglavlje 2

Metoda tableauxa za logiku sudova

Naš je cilj u ovom poglavlju predstaviti postupak za izgradnju modela u kojem je zadana formula ispunjiva. Naziv te metode je **metoda tableauxa** ili **glavni test**. Više o logici sudova i aksiomatskom pristupu ovom problemu možete vidjeti u [10].

Najprije moramo nešto reći općenito o formulama.

2.1 Tipovi formula

Formule se mogu podjeliti u tri klase:

1. **Konjunktivne formule** - svaka konjunktivna formula α se može shvatiti kao konjunkcija najviše dvije formule, koje zovemo **konjunktivne komponente**.
2. **Disjunktivne formule** - svaka disjunktivna formula β se može shvatiti kao disjunkcija najviše dvije formule, koje zovemo **disjunktivne komponente**.
3. **Literali** - logičke konstante $\top$ i $\bot$, propozicionalne varijable i njihove negacije. Oni nemaju komponenata.

Primjer 2.1.1. *Konjunktivne komponente formule* $\alpha = \neg\varphi_1 \land \varphi_2$ *su* $\neg\varphi_1$ *i* φ_2.
Disjunktivne komponente formule $\beta = \neg\varphi_1 \lor \neg\varphi_2$ *su* $\neg\varphi_1$ *i* $\neg\varphi_2$.

Prikazat ćemo sve tipove formula tablično kako bismo imali bolji pregled.

Konjunktivna formula	Konjunktivne komponente	
$\varphi \wedge \psi$	φ	ψ
$\neg(\varphi \vee \psi)$	$\neg\varphi$	$\neg\psi$
$\neg(\varphi \to \psi)$	φ	$\neg\psi$
$\neg\neg\varphi$	φ	

Tablica 2.1: Konjunktivne formule i njihove komponente

Disjunktivna formula	Disjunktivne komponente	
$\varphi \vee \psi$	φ	ψ
$\varphi \to \psi$	$\neg\varphi$	ψ
$\neg(\varphi \wedge \psi)$	$\neg\varphi$	$\neg\psi$

Tablica 2.2: Disjunktivne formule i njihove komponente

Literali
$p, \forall p \in \mathbf{PROP}$
$\neg p, \forall p \in \mathbf{PROP}$
$\top$
$\perp$

Tablica 2.3: Literali

Definicija 2.1.2. ***Interpretacija****, u oznaci I, je svako preslikavanje sa skupa* ***PROP*** *u skup* $\{0, 1\}$.

Interpretacija se proširuje na sve formule logike sudova rekurzivno na sljedeći način:

$$
\begin{aligned}
I(\neg\varphi) = 1 &\Leftrightarrow I(\varphi) = 0 \\
I(\varphi_1 \wedge \varphi_2) = 1 &\Leftrightarrow I(\varphi_1) = 1 \text{ } i \text{ } I(\varphi_2) = 1 \\
I(\varphi_1 \vee \varphi_2) = 1 &\Leftrightarrow I(\varphi_1) = 1 \text{ } ili \text{ } I(\varphi_2) = 1 \\
I(\varphi_1 \to \varphi_2) = 1 &\Leftrightarrow I(\varphi_1) = 0 \text{ } ili \text{ } I(\varphi_2) = 1 \\
I(\top) = 1 &\text{ } uvijek \\
I(\perp) = 1 &\text{ } nikad
\end{aligned}
$$

Ako je vrijednost $I(\varphi) = 1$ za neku formulu φ, kažemo da je φ istinita za interpretaciju I. Ako je vrijednost $I(\varphi) = 0$ za neku formulu φ, kažemo da je φ neistinita za interpretaciju I.

Za skup formula logike sudova kažemo da je ***ispunjiv*** *ako postoji interpretacija za koju su sve formule tog skupa istinite, inače kažemo da nije ispunjiv.*

Očigledno vrijedi sljedeća lema:

Lema 2.1.3. *Vrijede sljedeće tvrdnje:*

1. *Svaka konjunktivna formula ekvivalentna je konjunkciji svojih konjunktivnih komponenata.*

2. *Svaka disjunktivna formula ekvivalentna je disjunkciji svojih disjunktivnih komponenata.*

Posljednja lema pokazuje nam kako je za odrediti istinosnu vrijednost neke formule potrebno promatrati istinosnu vrijednost manjih fomula koje su sastavni dio polazne. Ova ideja je nit vodilja u razvoju metode tabelauxa.

2.2 Tableaux - osnovni pojmovi

Cilj ovog odjeljka je dati osnovne definicije pojmova koji će nam biti važni u daljnjim razmatranjima.

Definicija 2.2.1. ***Stablo*** *je uređen par $(S, <)$ gdje je S neprazan konačan skup čije članove zovemo* ***čvorovi****, a $<$ je irefleksivna i tranzitivna binarna relacija sa svojstvima:*

1. *postoji najmanji element $s_0 \in S$ (njega nazivamo* ***korijen****)*

2. *za svaki $s \in S$, $s \neq s_0$, postoji jedinstveni konačan niz $s_1, ..., s_n \in S$, tako da vrijedi:*

$$s_0 < s_1 < s_2 < ... < s_n = s,$$

   te za sve $i = 0, ...n-1$ vrijedi da je s_{i+1} ***neposredni sljedbenik*** *ili* ***dijete*** *od s_i, tj. ne postoji s_y takav da $s_i < s_y < s_{i+1}$.*

List *je čvor koji nema sljedbenika.* ***Put*** *je svaki niz čvorova $s_1, ..., s_n$, pri čemu je za svaki $i \in \{1...n\}$ čvor s_{i+1} neposredni sljedbenik od s_i. Za svaki $s \in S$, parcijalno uređen skup $\{x \in S | s \leq x\}$ nazivamo* ***podstablo****.* ***Visina čvora*** *s, u oznaci $h(s)$, je maksimalna vrijednost* *duljine puta od tog čvora do nekog lista.* ***Visina stabla*** *je visina korijena.* ***Granom*** *nazivamo put od nekog lista l do korijena.* ***Označeno stablo*** *je stablo čijem je svakom čvoru pridružen skup formula. Označeno stablo se naziva i* ***tableau***[1].

Metoda tableauxa formalni je sustav traženja kontradikcije za neku formulu ili skup formula logike koristeći pritom pravila razgradnje formula na njihove komponente. Ta pravila su dobivena tako da istinitost polazne formule ovisi o istinitosti komponenata. Kažemo još da je istinitost polazne formule **reducirana** na istinitost komponenti.

U prošlom smo odjeljku vidjeli tipove formula i njihove komponente, a pravila koja ćemo sada navesti nisu bitno drugačija i jedina razlika je zapravo u stablastoj strukturi.

Definicija 2.2.2. *Pravila za redukciju formula* ***metode tableauxa*** *u logici sudova podijeljena su na dvije skupine: pravila koja ne zahtijevaju grananje i ona koja zahtijevaju. Niže su stupčano prikazane te dvije skupine:*

$$(\wedge)\ \begin{array}{c} \varphi \wedge \psi \\ | \\ \varphi, \psi \end{array} \qquad (\neg\wedge)\ \begin{array}{c} \neg(\varphi \wedge \psi) \\ \diagup \quad \diagdown \\ \neg\varphi \quad \neg\psi \end{array}$$

$$(\neg\vee)\ \begin{array}{c} \neg(\varphi \vee \psi) \\ | \\ \neg\varphi, \neg\psi \end{array} \qquad (\vee)\ \begin{array}{c} \varphi \vee \psi \\ \diagup \quad \diagdown \\ \varphi \quad \psi \end{array}$$

$$(\neg \rightarrow)\ \begin{array}{c} \neg(\varphi \rightarrow \psi) \\ | \\ \varphi, \neg\psi \end{array} \qquad (\rightarrow)\ \begin{array}{c} \varphi \rightarrow \psi \\ \diagup \quad \diagdown \\ \neg\varphi \quad \psi \end{array}$$

Primjena ovih pravila na polaznoj formuli ili skupu nastavlja se do ***popunjenja****, tj. do onog trenutka kada za svaku granu tableaua vrijedi da je za svaku formulu te grane osim literala primjenjeno neko pravilo redukcije. Za takav tableau kažemo da je* ***pun****.*

Za granu kažemo da je ***zatvorena*** *ako se na njoj nalazi komplementaran par literala p i $\neg p$, za neki $p \in$* ***PROP****, ili $\bot$ ili $\neg\top$. Za pun tableau kome je svaka grana zatvorena kažemo da je* ***zatvoren****, inače kažemo da je* ***otvoren****.*

[1] *Tableaux* je na francuskom množina riječi *tableau*. Stoga ćemo cjelokupnu metodu zvati *metoda tableauxa*, a pojedino označeno stablo dobiveno primjenom te metode zvat ćemo *tableau*.

2.3 Adekvatnost i potpunost tableaua za logiku sudova

U prošlom odjeljku govorili smo o pravilima redukcije formula logike sudova i samo smo načelno rekli da se primjenom tih pravila istinitost veće formule svodi na istinitost komponenti. Sada nam je cilj to formalno i dokazati.

Teorem 2.3.1. *Tableau čiji je korijen označen formulom φ logike sudova je zatvoren ako i samo ako formula φ nije ispunjiva.*

Ovaj teorem se zapravo sastoji od dva smjera i tako ćemo ga i mi dokazati. Prvi smjer naziva se i **teorem adekvatnosti**, a glasi:

Teorem 2.3.2. *Ako za formulu φ logike sudova postoji zatvoren tableau čiji je korijen označen formulom φ, tada φ nije ispunjiva.*

Dokaz. Za dokaz ovog teorema, dokazat ćemo nešto jaču tvrdnju.

Neka je $\mathcal{T}_n$ zatvoreno podstablo tableaua $\mathcal{T}$ s korijenom n i neka je $U(n)$ skup formula kojim je označen n. Tada $U(n)$ nije ispunjiv. (*)

Tvrdnju * dokazujemo indukcijom po visini podstabla.

Baza

Neka je $\mathcal{T}_n$ zatvoreno podstablo i pretpostavimo da je $h(n) = 0$. Iz definicije visine slijedi da je n list, a kako znamo da je $\mathcal{T}_n$ zatvoreno podstablo, za $U(n)$ vrijedi jedna od sljedećih tvrdnji:

1. $U(n)$ sadrži komplementaran par literala te po definiciji istinitosti, taj par ne može istovremeno biti istinit za neku interpretaciju, pa tako po definiciji ispunjivosti skupa ni skup $U(n)$ nije ispunjiv.

2. $U(n)$ sadrži formulu $\perp$ koja po definiciji istinitosti nikad nije istinita pa ni skup $U(n)$ nije ispunjiv.

3. $U(n)$ sadrži formulu $\neg\top$ koja po definiciji istinitosti nikad nije istinita pa ni skup $U(n)$ nije ispunjiv.

Pretpostavka: Za svaki čvor m visine $h(m) < h(n)$, ako je $\mathcal{T}_m$ zatvoren, onda skup $U(m)$ nije ispunjiv.

Korak

Neka je $h(n) > 0$. Znamo da je onda korišteno neko pravilo metode tableauxa kako bi se dobila djeca čvora n koja označimo sa $n^{'}$ i ukoliko je potrebno $n^{''}$:

1. Korišteno je pravilo $(\wedge)$: Tada $U(n) = \{\varphi_1 \wedge \varphi_2\} \cup U_0$, a dijete dobiveno pravilom $U(n^{'}) = \{\varphi_1, \varphi_2\} \cup U_0$ za neki moguće prazan skup U_0. Znamo da je $h(n^{'}) = h(n) - 1$ pa po pretpostavci indukcije vrijedi da $U(n^{'})$ nije ispunjiv. Imamo dva slučaja:

   a) U_0 nije ispunjiv, ali $U_0 \subseteq U(n)$ pa tada ni $U(n)$ nije ispunjiv.

   b) Ako je U_0 ispunjiv, onda za svaku interpretaciju I vrijedi jedno od sljedećeg: ili je $I(\varphi_1) = 0$ ili $I(\varphi_2) = 0$. No po definiciji istinitosti za formulu $\varphi_1 \wedge \varphi_2$, ona neće nikad biti istinita, pa tako ni $U(n)$ nije ispunjiv u oba slučaja.

2. Korišteno je pravilo $(\neg\vee)$: Tada $U(n) = \{\neg(\varphi_1 \vee \varphi_2)\} \cup U_0$, a dijete dobiveno pravilom $U(n^{'}) = \{\neg\varphi_1, \neg\varphi_2\} \cup U_0$ za neki moguće prazan skup U_0. Znamo da je $h(n^{'}) = h(n) - 1$ pa po pretpostavci indukcije vrijedi da $U(n^{'})$ nije ispunjiv. Imamo dva slučaja:

   a) U_0 nije ispunjiv, ali $U_0 \subseteq U(n)$ pa tada ni $U(n)$ nije ispunjiv.

   b) Ako je U_0 ispunjiv, onda za svaku interpretaciju I vrijedi jedno od sljedećeg: ili je $I(\varphi_1) = 1$ ili $I(\varphi_2) = 1$. No po lemi 2.1.3 za formulu $\neg(\varphi_1 \vee \varphi_2)$, ona je istinita samo ako niti φ_1 niti φ_2 nisu istinite, pa tako $U(n)$ nije ispunjiv u oba slučaja.

3. Korišteno je pravilo $(\neg \rightarrow)$: Tada $U(n) = \{\neg(\varphi_1 \rightarrow \varphi_2)\} \cup U_0$, a dijete dobiveno pravilom $U(n^{'}) = \{\varphi_1, \neg\varphi_2\} \cup U_0$ za neki moguće prazan skup U_0. Znamo da je $h(n^{'}) = h(n) - 1$ pa po pretpostavci indukcije vrijedi da $U(n^{'})$ nije ispunjiv. Imamo dva slučaja:

   a) U_0 nije ispunjiv, ali $U_0 \subseteq U(n)$ pa tada ni $U(n)$ nije ispunjiv.

   b) Ako je U_0 ispunjiv, onda za svaku interpretaciju I vrijedi jedno od sljedećeg: ili je $I(\varphi_1) = 0$ ili $I(\varphi_2) = 1$. No po lemi 2.1.3 za formulu $\neg(\varphi_1 \rightarrow \varphi_2)$, ona je istinita samo ako je φ_1 istinita a φ_2 nije istinita, pa tako $U(n)$ nije ispunjiv u oba slučaja.

4. Korišteno je pravilo $(\neg\wedge)$: Tada $U(n) = \{\neg(\varphi_1 \wedge \varphi_2)\} \cup U_0$, a djeca dobivena pravilom su $U(n^{'}) = \{\neg\varphi_1\} \cup U_0$ i $U(n^{''}) = \{\neg\varphi_2\} \cup U_0$ za neki moguće prazan skup U_0. Znamo da je $h(n^{'}) < h(n)$ i $h(n^{''}) < h(n)$ pa po pretpostavci indukcije vrijedi da $U(n^{'})$ i $U(n^{''})$ nisu ispunjivi. Imamo dva slučaja:

a) U_0 nije ispunjiv, ali $U_0 \subseteq U(n)$ pa tada ni $U(n)$ nije ispunjiv.

b) Ako je U_0 ispunjiv, onda za svaku interpretaciju I vrijede sljedeće dvije tvrdnje: $I(\varphi_1) = 1$ (jer $U(n^{'})$ nije ispunjiv) i $I(\varphi_2) = 1$ (jer $U(n^{''})$ nije ispunjiv). No po lemi 2.1.3 za formulu $\neg(\varphi_1 \wedge \varphi_2)$, ona je istinita samo ako φ_1 nije istinita ili φ_2 nije istinita, pa $U(n)$ nije ispunjiv.

5. Korišteno je pravilo ($\vee$): Tada $U(n) = \{\varphi_1 \vee \varphi_2\} \cup U_0$, a djeca dobivena pravilom su $U(n^{'}) = \{\varphi_1\} \cup U_0$ i $U(n^{''}) = \{\varphi_2\} \cup U_0$ za neki moguće prazan skup U_0. Znamo da je $h(n^{'}) < h(n)$ i $h(n^{''}) < h(n)$ pa po pretpostavci indukcije vrijedi da $U(n^{'})$ i $U(n^{''})$ nisu ispunjivi. Imamo dva slučaja:

   a) U_0 nije ispunjiv, ali $U_0 \subseteq U(n)$ pa tada ni $U(n)$ nije ispunjiv.

   b) Ako je U_0 ispunjiv, onda za svaku interpretaciju I vrijede sljedeće dvije tvrdnje: $I(\varphi_1) = 0$ (jer $U(n^{'})$ nije ispunjiv) i $I(\varphi_2) = 0$ (jer $U(n^{''})$ nije ispunjiv). No po definiciji istinitosti za formulu $\varphi_1 \vee \varphi_2$, ona neće nikad biti istinita, pa $U(n)$ nije ispunjiv.

6. Korišteno je pravilo ($\rightarrow$): Tada $U(n) = \{\varphi_1 \rightarrow \varphi_2\} \cup U_0$, a djeca dobivena pravilom su $U(n^{'}) = \{\neg\varphi_1\} \cup U_0$ i $U(n^{''}) = \{\varphi_2\} \cup U_0$ za neki moguće prazan skup U_0. Znamo da je $h(n^{'}) < h(n)$ i $h(n^{''}) < h(n)$ pa po pretpostavci indukcije vrijedi da $U(n^{'})$ i $U(n^{''})$ nisu ispunjivi. Imamo dva slučaja:

   a) U_0 nije ispunjiv, ali $U_0 \subseteq U(n)$ pa tada ni $U(n)$ nije ispunjiv.

   b) Ako je U_0 ispunjiv, onda za svaku interpretaciju I vrijede sljedeće dvije tvrdnje: $I(\varphi_1) = 1$ (jer $U(n^{'})$ nije ispunjiv) i $I(\varphi_2) = 0$ (jer $U(n^{''})$ nije ispunjiv). No po definiciji istinitosti za formulu $\varphi_1 \rightarrow \varphi_2$, ona neće nikad biti istinita, pa $U(n)$ nije ispunjiv.

Po principu matematičke indukcije, tvrdnja * vrijedi. Tada posebno vrijedi i tvrdnja teorema. □

Drugi smjer, tj. da se za neispunjivu formulu svaki tableau zatvori, obično se zove **teorem potpunosti** i nije toliko jednostavan kao teorem adekvatnosti. Jednostavnije ga je dokazati u obliku obrata po kontrapoziciji:

Teorem 2.3.3. *Za otvoreni tableau čiji je korijen označen formulom φ logike sudova, φ je ispunjiva.*

Prije nego li krenemo dokazivati, treba definirati još neka svojstva skupa formula koja će nam dobro doći.

Definicija 2.3.4. *Za neki skup kažemo da je* ***uočljivo inkonzistentan*** *ako sadrži komplementaran par literala, ili $\perp$ ili $\neg\top$.*

Definicija 2.3.5. *Za skup Γ formula kažemo da je* ***Hintikkin skup*** *ako vrijedi:*

1. *Skup Γ nije uočljivo inkonzistentan*
2. *Ako je $\varphi \in \Gamma$ konjunktivna formula, tada Γ sadrži i sve njene konjunktivne komponente.*
3. *Ako je $\varphi \in \Gamma$ disjunktivna formula, tada Γ sadrži bar jednu od njezinih disjunktivnih komponenti*

Teorem 2.3.6. *Svaki Hintikkin skup Γ je ispunjiv.*

Dokaz. Neka je Γ jedan Hintikkin skup. Neka je $\mathcal{P}_\Gamma$ skup svih propozicionalnih varijabli koje se pojavljuju u formulama iz Γ. Definiramo interpretaciju I ovako:

Za svaki $p \in \mathcal{P}_\Gamma$:

$$I(p) = \left\{ \begin{array}{c|l} 0 & \neg p \in \Gamma \\ 1 & \text{inače} \end{array} \right.$$

Kako znamo da je Γ Hintikkin skup, tada po (1) znamo da je interpretacija I dobro definirana, tj. svakoj propozicionalnoj varijabli pridružena je samo jedna vrijednost. Drugi dio u definiciji I obuhvaća i sve literale koji se možda pojave u nekoj formuli ali nikada ne sami za sebe, a to je moguće jer više grana stabla mogu dijeliti dio puta, a tako i dio formula.

Matematičkom indukcijom po strukturi formule φ pokazujemo da su sve formule u Γ istinite za interpretaciju I:

Baza

1. Za $\varphi = p$ izravno iz definicije interpretacije I vrijedi $I(p) = 1$.
2. Z $\varphi = \top$ izravno iz definicije interpretacije I vrijedi $I(\top) = 1$.
3. Za $\varphi = \neg p$ izravno iz definicije interpretacije I vrijedi $I(p) = 0$ što je po definiciji istinitosti ekvivalentno $I(\neg p) = 1$.
4. Za $\varphi = \neg\perp$ izravno iz definicije interpretacije I vrijedi $I(\perp) = 0$ što je po definiciji istinitosti ekvivalentno $I(\neg\perp) = 1$.

Pretpostavka: Svaka formula $\psi \in \Gamma$, takva da je $len(\psi) < len(\varphi)$, je istinita za interpretaciju I.

Korak

1. Za $\varphi = \varphi_1 \wedge \varphi_2$ po (2) iz definicije 2.3.5 vrijedi $\varphi_1 \in \Gamma$ i $\varphi_2 \in \Gamma$. Iz pretpostavke indukcije slijedi $I(\varphi_1) = 1$ i $I(\varphi_2) = 1$. Po definiciji istinitosti za $\wedge$ vrijedi $I(\varphi) = 1$.

2. Za $\varphi = \neg(\varphi_1 \vee \varphi_2)$ po (2) iz definicije 2.3.5 vrijedi $\neg\varphi_1 \in \Gamma$ i $\neg\varphi_2 \in \Gamma$. Iz pretpostavke indukcije slijedi $I(\neg\varphi_1) = 1$ i $I(\neg\varphi_2) = 1$ što je po definiciji istinitosti za $\neg$ ekvivalentno $I(\varphi_1) = 0$ i $I(\varphi_2) = 0$. Po lemi 2.1.3 vrijedi $I(\varphi) = 1$.

3. Za $\varphi = \neg(\varphi_1 \to \varphi_2)$ po (2) iz definicije 2.3.5 vrijedi $\varphi_1 \in \Gamma$ i $\neg\varphi_2 \in \Gamma$. Iz pretpostavke indukcije slijedi $I(\varphi_1) = 1$ i $I(\neg\varphi_2) = 1$ što je po definiciji istinitosti za $\neg$ ekvivalentno $I(\varphi_2) = 0$. Po lemi 2.1.3 vrijedi $I(\varphi) = 1$.

4. Za $\varphi = \neg\neg\psi$ po (2) iz definicije 2.3.5 vrijedi $\psi \in \Gamma$. Iz pretpostavke indukcije slijedi $I(\psi) = 1$. Po lemi 2.1.3 vrijedi $I(\varphi) = 1$.

5. Za $\varphi = \neg(\varphi_1 \wedge \varphi_2)$ po (3) iz definicije 2.3.5 vrijedi $\neg\varphi_1 \in \Gamma$ ili $\neg\varphi_2 \in \Gamma$. Iz pretpostavke indukcije slijedi $I(\neg\varphi_1) = 1$ ili $I(\neg\varphi_2) = 1$ što je po definiciji istinitosti za $\neg$ ekvivalentno $I(\varphi_1) = 0$ ili $I(\varphi_2) = 0$. Po lemi 2.1.3 vrijedi $I(\varphi) = 1$.

6. Za $\varphi = \varphi_1 \vee \varphi_2$ po (3) iz definicije 2.3.5 vrijedi $\varphi_1 \in \Gamma$ ili $\varphi_2 \in \Gamma$. Iz pretpostavke indukcije slijedi $I(\varphi_1) = 1$ ili $I(\varphi_2) = 1$. Po definiciji istinitosti za $\vee$ vrijedi $I(\varphi) = 1$.

7. Za $\varphi = \varphi_1 \to \varphi_2$ po (3) iz definicije 2.3.5 vrijedi $\neg\varphi_1 \in \Gamma$ ili $\varphi_2 \in \Gamma$. Iz pretpostavke indukcije slijedi $I(\neg\varphi_1) = 1$ ili $I(\varphi_2) = 1$ što je po definiciji istinitosti za $\neg$ ekvivalentno $I(\varphi_1) = 0$. Po definiciji istinitosti za $\to$ vrijedi $I(\varphi) = 1$.

Po principu matematičke indukcije, tvrdnja teorema vrijedi. □

Sada kada imamo vezu ispunjivosti i Hintikkinih skupova, za dokazati potpunost trebamo pokazati sljedeću tvrdnju:

Teorem 2.3.7. *Neka je $\mathcal{T}$ otvoreni tabelau i neka je U unija svih skupova koji označavaju čvorove jedne njegove otvorene grane. Tada je U Hintikkin skup.*

Dokaz. Kako bismo dokazali da je U zbilja Hintikkin skup, moramo pokazati da vrijede tvrdnje (1), (2) i (3) iz definicije Hintikkinog skupa.

Prvo pokazujemo da vrijedi tvrdnja (1).

Iz iskaza teorema znamo da je U unija skupova formula kojima su označeni čvorovi jedne otvorene grane tableaua $\mathcal{T}$. To znači da za svaki $p \in$ **PROP**, U ne sadrži odjednom literale p i $\neg p$. Također, U ne sadrži niti $\bot$ ni $\neg\top$. Ovime odmah vrijedi tvrdnja (1).

Zatim pokazujemo da vrijedi tvrdnja (2).

Za sve čvorove i različite od lista:

1. Neka je $\varphi = \varphi_1 \wedge \varphi_2 \in U$ formula. Kako je grana otvorena, tada znamo da je u nekom čvoru i te grane primjenjeno pravilo ($\wedge$) nad φ. Po definiciji pravila ($\wedge$) tada $\{\varphi_1, \varphi_2\} \subseteq U$.

2. Neka je $\varphi = \neg(\varphi_1 \vee \varphi_2) \in U$ formula. Kako je grana otvorena, tada znamo da je u nekom čvoru i te grane primjenjeno pravilo ($\neg\vee$) nad φ. Po definiciji pravila ($\neg\vee$) tada $\{\neg\varphi_1, \neg\varphi_2\} \subseteq U$.

3. Neka je $\varphi = \neg(\varphi_1 \rightarrow \varphi_2) \in U$ formula. Kako je grana otvorena, tada znamo da je u nekom čvoru i te grane primjenjeno pravilo ($\neg \rightarrow$) nad φ. Po definiciji pravila ($\neg \rightarrow$) tada $\{\varphi_1, \neg\varphi_2\} \subseteq U$.

Odmah vrijedi tvrdnja (2).

Naposljetku, pokazujemo da vrijedi i tvrdnja (3).

Za sve čvorove i različite od lista:

1. Neka je $\varphi = \neg(\varphi_1 \wedge \varphi_2) \in U$ formula. Kako je grana otvorena, tada znamo da je u nekom čvoru i te primjenjeno pravilo ($\neg\wedge$) nad φ. Po definiciji pravila ($\neg\wedge$) tada vrijedi ili $\{\neg\varphi_1\} \subseteq U$ ili $\{\neg\varphi_2\} \subseteq U$.

2. Neka je $\varphi = \varphi_1 \vee \varphi_2 \in U$ formula. Kako je grana otvorena, tada znamo da je u nekom čvoru i te grane primjenjeno pravilo ($\vee$) nad φ. Po definiciji pravila ($\vee$) tada vrijedi ili $\{\varphi_1\} \subseteq U$ ili $\{\varphi_2\} \subseteq U$.

3. Neka je $\varphi = \varphi_1 \rightarrow \varphi_2 \in U$ formula. Kako je grana otvorena, tada znamo da je u nekom čvoru i te grane primjenjeno pravilo ($\rightarrow$) nad φ. Po definiciji pravila ($\rightarrow$) tada vrijedi ili $\{\neg\varphi_1\} \subseteq U$ ili $\{\varphi_2\} \subseteq U$.

Odmah vrijedi tvrdnja (3). □

Sada imamo dovoljno moćan alat i možemo dati dokaz teorema 2.3.3, tj. teorema potpunosti.

Dokaz. Neka je $\mathcal{T}$ jedan otvoreni tableau s formulom φ u korijenu. Tada po definiciji postoji otvorena grana o. Sa U označimo uniju oznaka formula svih čvorova na toj grani. Po teoremu 2.3.7 U je Hintikkin skup, i formula φ pripada tom skupu. Po teoremu 2.3.6 vrijedi da je U ispunjiv, pa posebno i da je φ ispunjiva. $\square$

Time smo dobili i da za tableaux metodu u logici sudova vrijedi teorem 2.3.1.

Poglavlje 3

Metoda tableauxa za osnovni modalni jezik

U ovom poglavlju cilj nam je proširiti već postojeće znanje tableaux metode na osnovni modalni jezik. Prije nego li krenemo sa definiranjem tableauxa, potrebni su nam još neki pojmovi.

3.1 Tipovi formula

U prošlom poglavlju rekli smo već nešto o podjeli formula na klase, sada ćemo to još proširiti.

U osnovnom modalnom jeziku prisutna je, uz one iz tablica 2.1, 2.2 i 2.3 još jedna klasa formula, koju zovemo **nasljedne formule**.

Ova se klasa odnosi na sve formule čija istinosna vrijednost ovisi o istinitosti određene potformule u nekom ili svim dostiživim svjetovima. Tu potformulu zovemo **nasljedna komponenta**. Formule oblika $\Diamond\varphi$ i $\neg\Box\varphi$ zovemo **egzistencijalno nasljedne formule** dok su one oblika $\Box\varphi$ i $\neg\Diamond\varphi$ **univerzalno nasljedne formule**.

Primjer 3.1.1. *Nasljedna komponenta formule $\theta = \Diamond\varphi$ jest φ.*

Za bolji pregled, i ovaj tip formula navest ćemo tablično.

Nasljedna formula	Nasljedne komponente
$\Diamond\varphi$	φ
$\Box\varphi$	φ
$\neg\Diamond\varphi$	$\neg\varphi$
$\neg\Box\varphi$	$\neg\varphi$

Tablica 3.1: Nasljedne formule i njihove komponente

Nasljedne formule i literale nazivamo **primitivnim formulama**.

3.2 Prošireno zatvorenje formule

Kada u logici spomenemo zatvorenje formule uobičajeno se misli na formulu logike prvog reda dobivenu primjenom kvantifikatora na sve slobodne varijable u nekoj formuli.

U modalnoj logici nam je isto tako potreban pojam zatvorenja, međutim za nas će imati malo drugačije značenje.

Definicija 3.2.1. ***Prošireno zatvorenje formule*** *φ, u oznaci $ecl(\varphi)$, najmanji je skup formula koji zadovoljava sljedeće:*

1. *$\varphi \in ecl(\varphi)$;*
2. *$ecl(\varphi)$ je zatvoren na sve konjunktivne, disjunktivne i nasljedne komponente formula koje se nalaze u $ecl(\varphi)$.*

Definicija 3.2.2. *Za skup formula Φ definiramo $ecl(\Phi) = \bigcup\{ecl(\varphi) | \varphi \in \Phi\}$. Skup Φ za koji vrijedi $\Phi = ecl(\Phi)$ zovemo* ***zatvoren skup****.*

Primjer 3.2.3. *Pokažimo sada nekoliko primjera proširenog zatvorenja:*

1. *Za formulu $\varphi = \Diamond p \to \Box\Box q$ prošireno zatvorenje je $ecl(\varphi) = \{\varphi, \neg\Diamond p, \Box\Box q, \neg p, \Box q, q\}$*
2. *Za formulu $\varphi = \Box(p \to \neg\Diamond(q \vee \neg r))$ prošireno zatvorenje je $ecl(\varphi) = \{\varphi, p \to \neg\Diamond(q \vee \neg r), \neg p, \neg\Diamond(q \vee \neg r), \neg(q \vee \neg r), \neg q, r\}$*
3. *Za skup $\Phi = \{p \wedge \Diamond(q \vee (p \to \Box r)), r \to (\Box\Diamond s \to \neg q)\}$ prošireno zatvorenje je $ecl(\Phi) = \{p \wedge \Diamond(q \vee (p \to \Box r)), p, \Diamond(q \vee (p \to \Box r)), q \vee (p \to \Box r), q, p \to \Box r, \neg p, \Box r, r, r \to (\Box\Diamond s \to \neg q), \neg r, \Box\Diamond s \to \neg q, \neg\Box\Diamond s, \neg q, \neg\Diamond s, \neg s\}$*

Lema 3.2.4. *Neka je φ proizvoljna formula osnovnog modalnog jezika a φ_1 i ako je potrebno φ_2 oznake za njezine komponente kako je prikazano u tablicama 2.1, 2.2, 2.3 i 3.1. Tada vrijedi:*

$$len(\varphi_1) + len(\varphi_2) \leq len(\varphi)$$

Dokaz. Dokaz provodimo matematičkom indukcijom po strukturi formule:

Baza

Za literale tvrdnja vrijedi trivijalno jer oni nemaju komponenti.

Pretpostavka: Tvrdnja vrijedi za formule složenosti manje od n.

Korak

Neka je φ formula složenosti n.

1. $\varphi = \varepsilon_1 \wedge \varepsilon_2, \varphi_1 = \varepsilon_1, \varphi_2 = \varepsilon_2$

   $len(\varphi_1) + len(\varphi_2) = len(\varepsilon_1) + len(\varepsilon_2) \leq len(\varphi) = 1 + len(\varepsilon_1) + len(\varepsilon_2)$

   Tvrdnja se analogno dokazuje za $\varphi = \varepsilon_1 \vee \varepsilon_2$.

2. $\varphi = \neg(\varepsilon_1 \vee \varepsilon_2), \varphi_1 = \neg\varepsilon_1, \varphi_2 = \neg\varepsilon_2$

   $len(\varphi_1)+len(\varphi_2) = 2+len(\varepsilon_1)+len(\varepsilon_2) \leq len(\varphi) = 1+len(\varepsilon_1\vee\varepsilon_2) = 2+len(\varepsilon_1)+len(\varepsilon_2)$

   Tvrdnja se analogno dokazuje za $\varphi = \neg(\varepsilon_1 \wedge \varepsilon_2)$.

3. $\varphi = \neg(\varepsilon_1 \rightarrow \varepsilon_2), \varphi_1 = \varepsilon_1, \varphi_2 = \neg\varepsilon_2$

   $len(\varphi_1) + len(\varphi_2) = 1 + len(\varepsilon_1) + len(\varepsilon_2) \leq len(\varphi) = 1 + len(\varepsilon_1 \rightarrow \varepsilon_2) = 2 + len(\varepsilon_1) + len(\varepsilon_2)$

4. $\varphi = \neg\neg\varepsilon_1, \varphi_1 = \varepsilon_1$

   $len(\varphi_1) = len(\varepsilon_1) \leq len(\varphi) = 1 + len(\neg\varepsilon_1) = 2 + len(\varepsilon_1)$

5. $\varphi = \varepsilon_1 \to \varepsilon_2, \varphi_1 = \neg\varepsilon_1, \varphi_2 = \varepsilon_2$

   $len(\varphi_1) + len(\varphi_2) = 1 + len(\varepsilon_1) + len(\varepsilon_2) \leq len(\varphi) = 1 + len(\varepsilon_1) + len(\varepsilon_2)$

6. $\varphi = \Diamond\varepsilon_1, \varphi_1 = \varepsilon_1$

   $len(\varphi_1) = len(\varepsilon_1) \leq len(\varphi) = 1 + len(\varepsilon_1)$

   Tvrdnja se analogno dokazuje za $\varphi = \Box\varepsilon_1$.

7. $\varphi = \neg\Diamond\varepsilon_1, \varphi_1 = \neg\varepsilon_1$

   $len(\varphi_1) = 1 + len(\varepsilon_1) \leq len(\varphi) = 1 + len(\Diamond\varepsilon_1) = 2 + len(\varepsilon_1)$

   Tvrdnja se analogno dokazuje za $\varphi = \neg\Box\varepsilon_1$.

$\Box$

Sljedeća tvrdnja je izravna posljedica definicije proširenog zatvorenja i gornje leme.

Korolar 3.2.5. *Za svaku formulu φ osnovnog modalnog jezika $ecl(\varphi)$ je konačan skup.*

3.3 Tableaux - osnovni modalni jezik

U ovom odjeljku razradit ćemo pravila i pojmove potrebne za izgradnju tableaua u osnovnom modalnom jeziku.

3.3.1 Lokalni tableau

U drugom poglavlju razradili smo tableaux metodu za logiku sudova, te dokazali adekvatnost i potpunost. Osnovna razlika između osnovnog modalnog jezika i logike sudova jest u pojmu svjetova. Kada već razmatranu metodu promatramo sa stajališta osnovnog modalnog jezika, uočavamo kako ta metoda dosta dobro pokriva traženje tableaua u svakom pojedinačnom svijetu.

Promjena koja je potrebna kako bi metoda ispravno radila u nekom svijetu jest primitivne formule definirane na stranici 20 promatrati kao literale. Tableau dobiven ovako prilagođenom metodom zvat ćemo **lokalni tableau**.

Primjer 3.3.1. *Prilagođena metoda za skup formula $\Gamma = \{\Diamond\top, \Box p, \neg\Diamond p\}$ vraća otvoreni lokalni tableau. Naša prilagođena metoda ne sadrži pravilo kojim bi reducirali dane elemente skupa Γ. Odmah dobivamo otvoreni tableau. Međutim, iz definicije istinitosti za $\Box$ i*

$\neg\Diamond$, znamo da ovakav skup formula nije ispunjiv u osnovnom modalnom jeziku.

Napomena 3.3.2. *Ako za neku formulu postoji zatvoreni lokalni tableau, ona nije ispunjiva. Međutim, postojanje otvorenog lokalnog tableaua za neku formulu nije dovoljno za zaključiti ispunjivost te formule u osnovnom modalnom jeziku.*

Iako postoje nedostaci koje smo upravo vidjeli, lokalni tableau je dobar početak nove metode tableauxa za osnovni modalni jezik.

Definicija 3.3.3. *Kažemo da je Hintikkin skup Γ* ***potpuno proširen*** *ako je dobiven iz nekog skupa formula Θ primjenom sljedećih pravila tako da na početku niti jedan element skupa Θ nije označen kao iskorišten:*

1. *Za svaku konjunktivnu formulu $\varphi \in \Theta$ koja nije označena kao iskorištena dodaj u Θ sve njezine konjunktivne komponente i φ označi kao iskorištenu.*

2. *Za svaku disjunktivnu formulu $\varphi \in \Theta$ koja nije označena kao iskorištena dodaj u Θ jednu njezinu disjunktivnu komponentu i označi φ kao iskorištenu.*

Uočimo da pravilo (2) nedeterministički bira neku komponentu pa neki skup Θ može imati više, ali možda i nijedno potpuno proširenje. Ovakva proširenja su zapravo izgradnja otvorenih grana lokalnog tableaua.

Primjer 3.3.4. *Skup $\Theta_1 = \{\Diamond(p \to q), \bot\}$ je uočljivo inkonzistentan i nema niti jedno potpuno proširenje, dok skup $\Theta_2 = \{(p \lor q) \land r\}$ ima dva. To su $\Gamma_1 = \{(p \lor q) \land r, r, p \lor q, p\}$ i $\Gamma_2 = \{(p \lor q) \land r, r, p \lor q, q\}$.*

Primjer 3.3.5. *Skup $\Gamma = \{p \to q, \neg p, q\}$ jest jedan Hintikkin skup, ali nije dobiven potpunim proširenjem skupa $\Theta = \{p \to q\}$. Razlog leži u pravilu (2) koje ne dopušta dodavanje obje disjunktivne komponente jer će nakon jedne formula $p \to q$ biti iskorištena.*

Imamo sve kako bismo definirali prvu proceduru koju ćemo koristiti u našoj metodi tableauxa za osnovni modalni jezik. Familija skupova koju dobivamo izvršavanjem te procedure sastoji se od potpuno proširenih Hintikkinih skupova. Ona formalno opisuje izgradnju lokalnog tableaua.

Procedura 3.3.6. *Za neku familiju skupova $\mathcal{F}$ i neki konačan skup $\Theta \in \mathcal{F}$ procedura* **PotpunoProširen**, *u oznaci $PP(\Theta)$, nedeterministički izvršava sljedeće:*

1. *Za još neiskorištenu konjunktivnu formulu $\varphi \in \Theta$ s konjunktivnim komponentama φ_1 i φ_2, zamijeni skup Θ skupom $\Theta \cup \{\varphi_1, \varphi_2\}$ i označi formulu φ kao iskorištenu.*

2. *Za još neiskorištenu disjunktivnu formulu $\varphi \in \Theta$ s disjunktivnim komponentama φ_1 i φ_2, zamijeni skup Θ skupovima $\Theta \cup \{\varphi_1\}$ i $\Theta \cup \{\varphi_2\}$ te označi φ kao iskorištenu.*

Tako je dobivena familija skupova $PP(\Theta)$. Pravila (1) i (2) kod zamjene provjeravaju još i hoće li izvršavanjem te zamjene u familiju $PP(\Theta)$ biti dodan uočljivo inkonzistentan skup, te će takve ukloniti iz $PP(\Theta)$.
Prestanak izvršavanja pravila (1) i (2) nastupa kada su sve konjunktivne, odnosno disjunktivne formule iz skupa Θ iskorištene. Ovo će se sigurno dogoditi jer je svako potpuno proširenje podskup konačnog skupa $ecl(\Theta)$.

Propozicija 3.3.7. *Za svaki konačan skup Γ vrijedi:*

$$\bigwedge \Gamma \equiv \bigvee \left\{ \bigwedge \Delta \mid \Delta \in PP(\Gamma) \right\}$$

Dokaz. Neka je dan skup Γ. Za familiju $PP(\Gamma)$ iz leme 2.1.3 slijedi kako svaka zamjena skupova u proceduri **PotpunoProširen** čuva istinitost formule $\vee \{\wedge \Delta \mid \Delta \in PP(\Gamma)\}$ do na logičku ekvivalenciju. U prvom koraku te procedure familija $PP(\Gamma)$ sadrži samo skup Γ. □

Primjer 3.3.8. *Odredimo familiju $PP(\Gamma)$ za sljedeće skupove:*

1. *Skup kojemu je jedini član formula $\varphi = \neg(\Box p \to \Diamond\Box q)$:*

   $\Theta_1 = \{\varphi, \Box p, \neg\Diamond\Box q\}$.

2. *Skup kojemu je jedini član formula $\varphi = \neg(\Diamond(p \to q) \vee (\Box p \to \Box q))$:*

   $\Theta_1 = \{\varphi, \neg\Diamond(p \to q), \neg(\Box p \to \Box q), \Box p, \neg\Box q\}$.

3. *Skup kojemu je jedini član formula $\varphi = \neg(\neg(\Box p \wedge \neg\Diamond q) \to \Diamond q)$:*

   $\Theta_1 = \{\varphi, \neg(\Box p \wedge \neg\Diamond q), \neg\Diamond q, \neg\Box p\}$.

   Procedurom se dobiva i skup $\Theta_2 = \{\varphi, \neg(\Box p \wedge \neg\Diamond q), \neg\Diamond q, \neg\neg\Diamond q\}$, no on je uočljivo inkonzistentan te je stoga uklonjen.

4. *Skup kojemu je jedini član formula $\varphi = (\Diamond p \to \Box q) \wedge \neg(\Box\Box p \wedge \Diamond q)$:*

   $\Theta_1 = \{\varphi, \Diamond p \to \Box q, \neg(\Box\Box p \wedge \Diamond q), \neg\Diamond p, \neg\Box\Box p\}$.

   $\Theta_2 = \{\varphi, \Diamond p \to \Box q, \neg(\Box\Box p \wedge \Diamond q), \Box q, \neg\Box\Box p\}$.

   $\Theta_3 = \{\varphi, \Diamond p \to \Box q, \neg(\Box\Box p \wedge \Diamond q), \neg\Diamond p, \neg\Diamond q\}$.

   $\Theta_4 = \{\varphi, \Diamond p \to \Box q, \neg(\Box\Box p \wedge \Diamond q), \Box q, \neg\Diamond q\}$.

3.3.2 Pretableau

Cilj nam je sada dati proceduru za izgradnju prve verzije tableaua. Prije nego li krenemo na proceduru, potrebni su nam još neki pojmovi.

Definicija 3.3.9. ***Usmjereni graf*** *je uređen par (V, E), gdje je V konačan neprazan skup, čije elemente zovemo* ***vrhovi****, a $E \subseteq V \times V$ irefleksivna relacija čije elemente nazivamo* ***bridovi****.*

Označeni graf *je usmjereni graf čijem je svakom vrhu pridružen skup formula. Bridovima možemo, ali i ne moramo pridružiti neku formulu.*

Uređen par (G, Γ), gdje je G označeni graf, a Γ istaknuti skup formula osnovnog modalnog jezika koji je pridružen nekom vrhu označenog grafa G, zovemo ***označeni graf s istaknutim skupom formula****, u oznaci G^{Γ}.*

Označeni graf s istaknutim skupom Γ dobiven izvršavanjem dvije procedure koje ćemo istaknuti malo niže, zvat ćemo **pretableau**, u oznaci $\mathcal{P}^{\Gamma}$. Istaknuti skup Γ je upravo onaj skup formula kojemu želimo ispitati ispunjivost. Vrhove pretableaua označene bilo kakvim podskupovima skupa $ecl(\Gamma)$ zovemo **predstanja**. Ako želimo istaknuti da je vrh označen potpuno proširenim skupom, zovemo ga **stanje**. Skup $ecl(\Gamma)$ je domena naših procedura jer sadrži sve moguće formule koje su vezane uz pitanje ispunjivosti skupa Γ. Ponekad ćemo oznake vrhova, a ne same vrhove, zvati stanja i predstanja. Svaki pretableau sadrži i konačnu verziju tableaua za našu metodu.

Procedura 3.3.10. *Za neko predstanje Γ, za koje ova procedura još nije izvršena,* **PredProširenje** *izvršava sljedeće:*

1. *Izgradi familiju $PP(\Gamma)$ potpunih proširenja skupa Γ i dodaj te skupove kao oznake novih vrhova pretableaua. Te vrhove zvat ćemo* ***stanja potomci****.*

2. *Za svako novo stanje Θ, dodaj neoznačeni brid (Γ, Θ).*

3. *Ako u pretableau već postoji stanje Θ, ne stvaraj novo već samo dodaj neoznačeni brid (Γ, Θ).*

Familiju stanja potomaka označavat ćemo sa ***potomci***(Γ).

Potrebna nam je i oznaka skupa nasljednih komponenata svih univerzalno nasljednih formula nekog skupa Θ.

$$X(\Theta) := \{\psi \mid \Box\psi \in \Theta\} \cup \{\neg\psi \mid \neg\Diamond\psi \in \Theta\}$$

Procedura 3.3.11. *Za neko stanje označeno skupom Θ, za koje ova procedura još nije izvršena,* **Sljedbenik** *izvršava sljedeće za svaku egzistencijalno nasljednu formulu $\varphi \in \Theta$:*

1. *Dodaj predstanje* Γ *od* Θ. *Skupu* $X(\Theta)$ *dodaj nasljednu komponentu formule* φ *i time označi predstanje* Γ.

2. *Za svako novo predstanje* Γ, *dodaj brid* (Θ, Γ) *označen formulom* φ.

3. *Ako u pretableau već postoji predstanje* Γ, *ne stvaraj novo već samo dodaj još jedan brid* (Θ, Γ) *označen formulom* φ.

Izgradnja pretableaua počinje istaknutim skupom Γ. Taj skup pridružimo polaznom vrhu označenog grafa i izvršimo proceduru **PredProširenje**, a zatim na dobivena stanja proceduru **Sljedbenik**. Ovo izmjenjivanje se nastavlja sve dok više nije moguće dodati novo predstanje ili stanje.

Važno je napomenuti kako pretableau ima dvije vrste grananja: grananje u traženju i strukturno grananje.

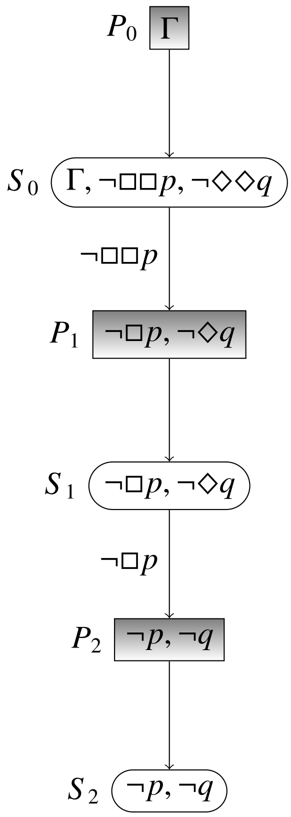
Grananje u traženju nastupa kada iz predstanja gradimo stanja potomke. Ovakvo grananje je disjunktivno i zapravo nam je potrebno samo jedno stanje koje kasnije preživi.

Strukturno grananje, međutim, nastupa kada iz stanja gradimo njegova nasljedna predstanja i ono je, kao što nagovješćuje i oznaka takvog brida, konjunktivno i moraju sva predstanja preživjeti.

Primjer 3.3.12. *Pokazat ćemo izgradnju pretableaua za više skupova:*

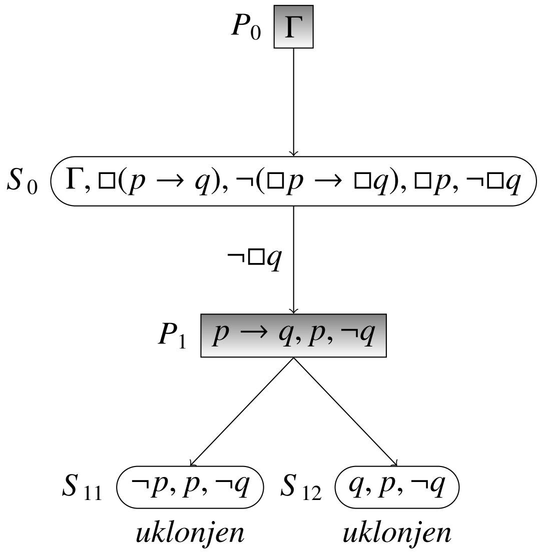
1. *Neka je skup* $\Gamma = \{\neg(\Box\Box p \lor \Diamond\Diamond q)\}$.

   Ovdje dajemo pripadi pretableau $\mathcal{P}^{\Gamma}$*:*

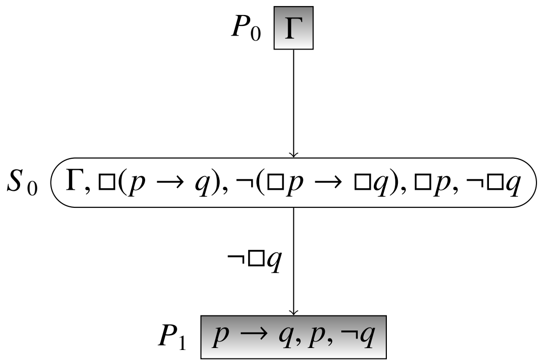


2. *Neka je skup* $\Gamma = \{\neg(\Box(p \to q) \to (\Box p \to \Box q))\}$. *Primjetimo da je formula iz danog skupa* Γ *negacija poznatog aksioma K.*

   Ovdje dajemo pretableau $\mathcal{P}^{\Gamma}$*:*

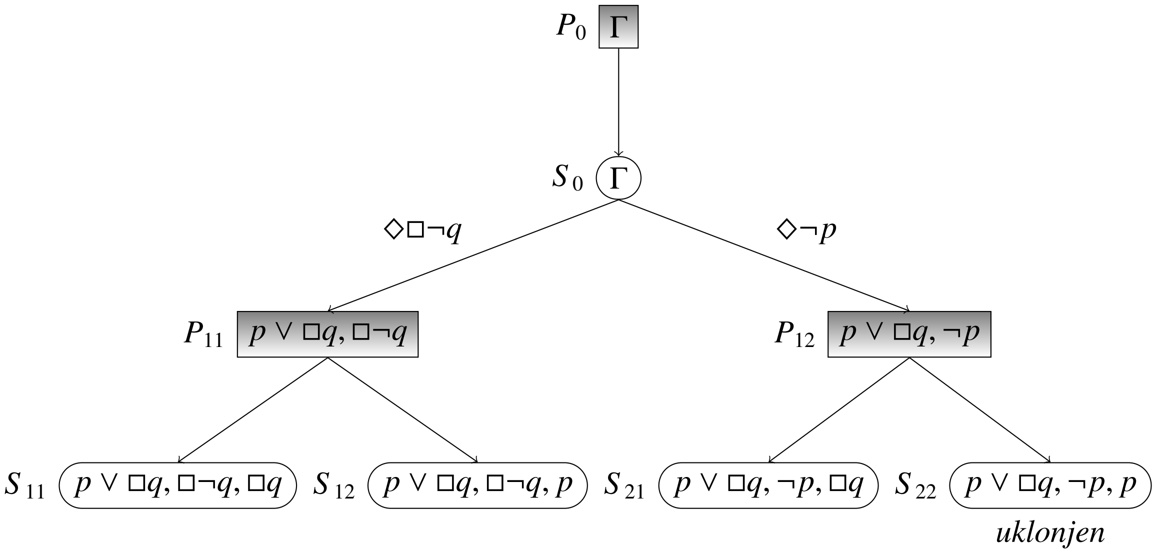


   Ovako izgleda pretableau $\mathcal{P}^{\Gamma}$ *nakon završetka izgradnje:*

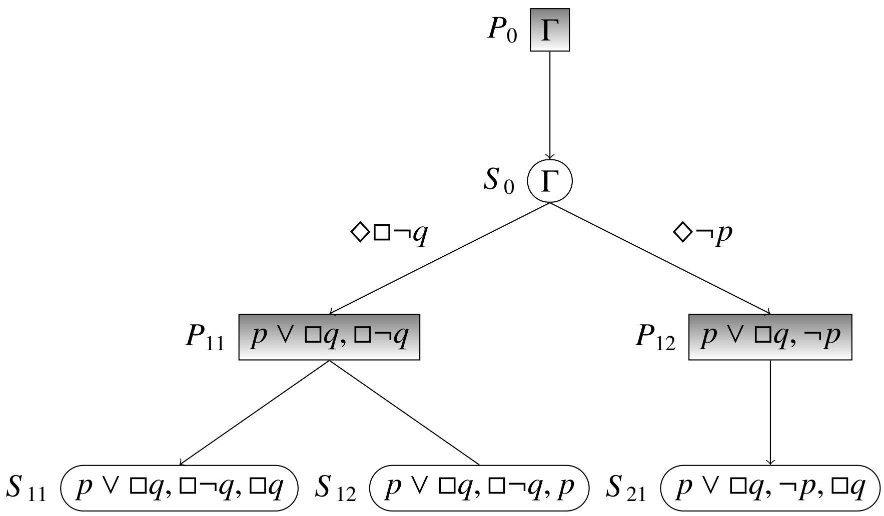


3. *Neka je skup* $\Gamma = \{\Box(p \vee \Box q), \Diamond\Box\neg q, \Diamond\neg p\}$.

Ovdje dajemo pripadi pretableau $\mathcal{P}^{\Gamma}$*:*



Ovako izgleda pretableau $\mathcal{P}^{\Gamma}$ *nakon završetka izgradnje:*



3.3.3 Uklanjanje predstanja i inicijalni tableau

Predstanja su samo pomoćni dio naše metode. U ovom dijelu cilj nam je ukloniti ih iz tableaua.

Procedura 3.3.13. *Za svako predstanje Γ u pretableau $\mathcal{P}^{\Gamma}$, procedura* UKLANJANJEPREDSTANJA *izvršava sljedeće:*

1. *Ukloni predstanje Γ iz pretableaua $\mathcal{P}^{\Gamma}$ zajedno sa svim bridovima (Γ, Y), $Y \in V$;*

2. *ako postoji stanje $\Delta \in \mathcal{P}^{\Gamma}$ i postoji brid $(\Delta, \Gamma) \in E$, onda za svako stanje $\Theta \in$ **potomci**(Γ) dodaj brid (Δ, Θ) s istom oznakom kao (Δ, Γ);*

3. *ukloni brid (Δ, Γ).*

Označeni graf dobiven ovom procedurom zovemo **inicijalni tableau**, uz oznaku $\mathcal{T}_0^{\Gamma}$.

Primjer 3.3.14. *Za svaki pretableau iz prošlog primjera dajemo njegov inicijalni tableau $\mathcal{T}_0^{\Gamma}$:*

1. *Za skup $\Gamma = \{\neg(\Box(p \to q) \to (\Box p \to \Box q))\}$ dajemo pripadni inicijalni tableau:*

$$\begin{tikzcd}
S_0\ \boxed{\Gamma, \neg\Box\Box p, \neg\Diamond\Diamond q} \arrow[d, "\neg\Box\Box p"'] \\
S_1\ \boxed{\neg\Box p, \neg\Diamond q} \arrow[d, "\neg\Box p"'] \\
S_2\ \boxed{\neg p, \neg q}
\end{tikzcd}$$

2. *Za skup $\Gamma = \{\neg(\Box\Box p \vee \Diamond\Diamond q)\}$ dajemo pripadni inicijalni tableau:*

$$S_0\ \boxed{\Gamma, \Box(p \to q), \neg(\Box p \to \Box q), \Box p, \neg\Box q}$$

3. *Za skup* $\Gamma = \{\Box(p \vee \Box q), \Diamond\Box\neg q, \Diamond\neg p\}$ *dajemo pripadni inicijalni tableau:*

$$\begin{tikzcd}[column sep=small] & S_0\ \Gamma \arrow[dl, "\Diamond\Box\neg q"'] \arrow[d, "\Diamond\Box\neg q"] \arrow[dr, "\Diamond\neg p"] & \\ S_{11}\ p \vee \Box q, \Box\neg q, \Box q & S_{12}\ p \vee \Box q, \Box\neg q, p & S_{21}\ p \vee \Box q, \neg p, \Box q \end{tikzcd}$$

3.3.4 Uklanjanje stanja i završni tableau

U ovom dijelu cilj nam je ukloniti sva stanja koja **nisu ostvariva**, pod čime smatramo ona stanja koja imaju egzistencijalnu nasljednu formulu ali ne i stanja nasljednike koji odgovaraju toj formuli.

Procedura 3.3.15. *Za inicijalni tableau* $\mathcal{T}_0^\Gamma$*, procedura* UklanjanjeStanja *izvršava sljede-će: ako stanje* Θ *sadrži formulu* $\Diamond\varphi$*, odnosno* $\neg\Box\varphi$*, a ne postoji stanje potomak koje sadrži formulu* φ*, odnosno* $\neg\varphi$*, ukloni stanje* Θ *iz inicijalnog tableaua* $\mathcal{T}_0^\Gamma$*.*

Primijetimo da je izvršavanjem te procedure moguće doći do novih stanja koja nisu ostvariva. Svaku novu iteraciju tableaua dobivenog tom procedurom zovemo **trenutni tableau**, uz oznaku $\mathcal{T}_n^\Gamma = \textsc{UklanjanjeStanja}(\mathcal{T}_{n-1}^\Gamma)$, $n \in \mathbb{N}$.

Ponavljanje staje ukoliko u trenutnom tableauau više ne postoje neostvariva stanja i tu iteraciju zovemo **završni tableau**, uz oznaku $\mathcal{T}^\Gamma = \mathcal{T}_n^\Gamma$, a skup stanja S^Γ. Ako u završnom tableauu postoji stanje Θ koje sadrži početni skup formula Γ, kažemo da je **otvoren**, dok u suprotnom kažemo da je završni tableau **zatvoren**.

Primjer 3.3.16. *Provjerimo sada kakvi su završni tableaui za skupove formula iz prethodnih primjera:*

1. *Završni tableau za skup* $\Gamma = \{\neg(\Box\Box p \vee \Diamond\Diamond q)\}$ *jednak je inicijalnom te kako je skup* Γ *član stanja* S_0*, tableau je otvoren.*

2. *Završni tableau za skup* $\Gamma = \{\neg(\Box(p \to q) \to (\Box p \to \Box q)\}$ *je prazan skup, jer je jedino preostalo stanje neostvarivo i uklonjeno. Dakle, tableau je zatvoren.*

3. *Završni tableau za skup* $\Gamma = \{\Box(p \vee \Box q), \Diamond\Box\neg q, \Diamond\neg p\}$ *jednak je inicijalnom te kako je skup* Γ *član stanja* S_0*, tableau je otvoren.*

3.4 Adekvatnost i potpunost tableauxa za osnovni modalni jezik

U ovom odjeljku cilj je povezati pojmove ispunjivosti i otvorenosti tableaua. U prošlom poglavlju to smo već napravili za metodu u logici sudova a sada ćemo na sličan način to učiniti i za osnovni modalni jezik.

U prošlom poglavlju smo kod iskaza teorema potpunosti govorili o korištenju obrata po kontrapoziciji. Slično ćemo ga u ovom odjeljku iskoristiti za teorem adekvatnosti.

Za dokazati adekvatnost želimo pokazati kako svaka procedura uklanjanja stanja i predstanja uklanja samo neostvarive vrhove grafa.

Lema 3.4.1. *Neka je $\mathfrak{M}$ jedan model i w jedan svijet tog modela. Ako za neko predstanje Θ pretableaua $\mathcal{P}^{\Gamma}$ vrijedi $\mathfrak{M}, w \Vdash \Theta$, tada za barem jedno od stanja $\Delta \in$ **potomci(Θ)** vrijedi $\mathfrak{M}, w \Vdash \Delta$.*

Dokaz. Iz procedure **PredProširenje**, znamo da se familija **potomci**(Θ) dobiva izvršavanjem procedure **PotpunoProširen**. Iz propozicije 3.3.7 slijedi da potpuno proširenje čuva ispunjivost. Slijedi tvrdnja. □

Lema 3.4.2. *Izvršavanjem procedure* **UklanjanjeStanja** *nije uklonjeno niti jedno ispunjivo stanje Δ iz inicijalnog tableaua $\mathcal{T}_0^{\Gamma}$.*

Dokaz. Kako bismo dokazali ovu lemu, dovoljno je pokazati indukcijom po n, da nijedno ispunjivo stanje nije uklonjeno izvršavanjem procedure **UklanjanjeStanja** na trenutni tableau $\mathcal{T}_n^{\Gamma}$. Stoga ćemo dokazati nešto jaču tvrdnju.
Za svaki $n \in \mathbb{N}$:

1. Ako je $\Delta \in \mathcal{T}_n^{\Gamma}$ ispunjiv, tada za svaku formulu $\Diamond\varphi \in \Delta$ postoji ispunjivo stanje $\Theta \in \mathcal{T}_n^{\Gamma}$ takvo da imamo brid $(\Delta, \Theta) \in E$ označen formulom $\Diamond\varphi$.

2. Ako je $\Delta \in \mathcal{T}_n^{\Gamma}$ ispunjiv, tada za svaku formulu $\neg\Box\varphi \in \Delta$ postoji ispunjivo stanje $\Theta \in \mathcal{T}_n^{\Gamma}$ takvo da imamo brid $(\Delta, \Theta) \in E$ označen formulom $\neg\Box\varphi$.

3. Sva ispunjiva stanja iz inicijalnog tableaua $\mathcal{T}_0^{\Gamma}$ su prisutna u trenutnom tableau $\mathcal{T}_n^{\Gamma}$.

Tvrdnja (3) direktno slijedi iz (1) i (2) te je dovoljno dokazati njih.

Baza:

Promatramo dva slučaja obzirom na način kako je označen određeni brid.

1. Neka je $\Delta \in \mathcal{T}_0^\Gamma$ bilo koje ispunjivo stanje. Neka su $\mathfrak{M}$ i w model i svijet u kojem je $\mathfrak{M}, w \Vdash \Delta$. Znamo iz procedure **Sljedbenik** kako su sva stanja $\Delta' \in \mathcal{T}_0^\Gamma$ za koja imamo brid $(\Delta, \Delta') \in E$ označen formulom $\Diamond\varphi$ dobivena kao potpuna proširenja predstanja $\Gamma = \{\varphi\} \cup X(\Delta)$.

   Nadalje, znamo da je $\Diamond\varphi \in \Delta$ pa je stoga i $\mathfrak{M}, w \Vdash \Diamond\varphi$. Iz definicije istinitosti za $\Diamond$ postoji svijet $w' \in \mathfrak{M}$ takav da $\mathfrak{M}, w' \Vdash \varphi$. Nadalje, po definiciji istinitosti za $\Box$, $X(\Delta)$ ostaje istinit u svakom dostiživom svijetu pa tako i u w'. Iz te dvije činjenice slijedi $\mathfrak{M}, w' \Vdash \Gamma$.

   Lema 3.4.1 povlači kako postoji barem jedno potpuno proširenje Θ predstanja Γ takvo da $\mathfrak{M}, w' \Vdash \Theta$. Po proceduri **PredProširenje** sigurno postoji stanje s oznakom $\Theta \in \mathcal{T}_0^\Gamma$ i brid (Δ, Θ) označen formulom $\Diamond\varphi$.

   Iz svega ovoga, stanje Δ nije moglo biti uklonjeno izvršavanjem procedure **UklanjanjeStanja** na $\mathcal{T}_0^\Gamma$.

2. Neka je $\Delta \in \mathcal{T}_0^\Gamma$ bilo koje ispunjivo stanje. Neka su $\mathfrak{M}$ i w model i svijet u kojem je $\mathfrak{M}, w \Vdash \Delta$. Znamo iz procedure **Sljedbenik** kako su sva stanja $\Delta' \in \mathcal{T}_0^\Gamma$ za koja imamo brid $(\Delta, \Delta') \in E$ označen formulom $\neg\Box\varphi$ dobivena kao potpuna proširenja predstanja $\Gamma = \{\neg\varphi\} \cup X(\Delta)$.

   Nadalje, znamo da je $\neg\Box\varphi \in \Delta$ pa je stoga i $\mathfrak{M}, w \Vdash \neg\Box\varphi$. Iz veze između operatora $\Box$ i $\Diamond$ znamo da $\neg\Box\varphi \equiv \Diamond\neg\varphi$. Iz definicije istinitosti za $\Diamond$ postoji svijet $w' \in \mathfrak{M}$ takav da $\mathfrak{M}, w' \Vdash \neg\varphi$. Nadalje, po definiciji istinitosti za $\Box$, $X(\Delta)$ ostaje istinit u svakom dostiživom svijetu pa tako i u w'. Iz te dvije činjenice slijedi $\mathfrak{M}, w' \Vdash \Gamma$.

   Lema 3.4.1 povlači kako postoji barem jedno potpuno proširenje Θ predstanja Γ takvo da $\mathfrak{M}, w' \Vdash \Theta$. Iz procedure **PredProširenje** slijedi da sigurno postoji stanje s oznakom $\Theta \in \mathcal{T}_0^\Gamma$ i brid (Δ, Θ) označen formulom $\neg\Box\varphi$.

   Iz svega ovoga, stanje Δ nije moglo biti uklonjeno izvršavanjem procedure **UklanjanjeStanja** na $\mathcal{T}_0^\Gamma$.

Pretpostavka: Tvrdnje (1) i (2) vrijede za sve $n < m$.

Korak:

Promatramo dva slučaja obzirom na način kako je označen određeni brid.

1. Uzmimo neko ispunjivo stanje $\Delta \in \mathcal{T}_m^\Gamma$. Za bilo koju formulu $\Diamond\varphi \in \Delta$, po bazi indukcije znamo da postoji stanje s oznakom $\Theta \in \mathcal{T}_0^\Gamma$ i brid (Δ, Θ) označen formulom $\Diamond\varphi$. Po pretpostavci indukcije, Θ je preživjelo n izvršavanja procedure **UklanjanjeStanja** te se nalazi u $\mathcal{T}_n^\Gamma$. Kako za m vrijedi $m = n + 1$, procedura **UklanjanjeStanja** se u tom koraku mora izvršiti na $\mathcal{T}_n^\Gamma$. Upravo smo pokazali kako u $\mathcal{T}_n^\Gamma$ imamo stanje Θ i brid (Δ, Θ), pa ih stoga procedura **UklanjanjeStanja** ne može ukloniti. Slijedi da su oni pristuni i u $\mathcal{T}_m^\Gamma$.

2. Uzmimo neko ispunjivo stanje $\Delta \in \mathcal{T}_m^\Gamma$. Za bilo koju formulu $\neg\Box\varphi \in \Delta$, po bazi indukcije znamo da postoji stanje s oznakom $\Theta \in \mathcal{T}_0^\Gamma$ i brid (Δ, Θ) označen formulom $\neg\Box\varphi$. Po pretpostavci indukcije, Θ je preživjelo n izvršavanja procedure **UklanjanjeStanja** te se nalazi u $\mathcal{T}_n^\Gamma$. Kako za m vrijedi $m = n + 1$, procedura **UklanjanjeStanja** se u tom koraku mora izvršiti na $\mathcal{T}_n^\Gamma$. Upravo smo pokazali kako u $\mathcal{T}_n^\Gamma$ imamo stanje Θ i brid (Δ, Θ), pa ih stoga procedura **UklanjanjeStanja** ne može ukloniti. Slijedi da su oni pristuni i u $\mathcal{T}_m^\Gamma$.

Dokazali smo matematičkom indukcijom tvrdnje (1) i (2), te time i cijelu snažniju tvrdnju. Sada tvrdnja leme odmah slijedi. □

Imamo sve što nam je potrebno za dokaz teorema adekvatnosti.

Teorem 3.4.3. *Neka je Γ ispunjiv skup osnovnog modalnog jezika. Tada je završni tableau $\mathcal{T}^\Gamma$ otvoren.*

Dokaz. Tvrdnja teorema odmah slijedi iz leme 3.4.2. □

Sada nas zanima i obratni smjer, no kao i za logiku sudova, prije nego li krenemo na dokazivanje, potrebni su nam još neki pojmovi.

Definicija 3.4.4. *Neka je dan zatvoren skup Θ formula osnovnog modalnog jezika.* ***Hintikkina struktura za skup*** *Θ je uređeni par $\mathcal{H} = (\mathfrak{F}, H)$, gdje je $\mathfrak{F} = (W, R)$ Kripkeov okvir a $H : W \to \mathcal{P}(\Theta)$ funkcija koja svakom svijetu $w \in W$ pridružuje skup formula iz Θ i za svaki svijet w vrijedi sljedeće:*

1. *$H(w)$ je Hintikkin skup;*

2. *ako je $\varphi \in H(w)$ egzistencijalna nasljedna formula, tada je njezina nasljedna komponenta element skupa $H(w')$ za neki svijet $w' \in W$ i vrijedi wRw';*

3. *ako je $\varphi \in H(w)$ univerzalna nasljedna formula, tada je njezina nasljedna komponenta element skupa $H(w')$ za svaki svijet $w' \in W$ takav da vrijedi wRw'.*

Hintikkina struktura je analogon Hintikkinog skupa kojeg smo koristili u dokazu potpunosti u drugom poglavlju. Uočimo da je vrlo slična Kripkeovom modelu. Naime, iako je valuacija u Kripkeovom modelu definirana kao funkcija koja svakoj propozicionalnoj varijabli pridružuje skup svjetova u kojima je ona istinita, možemo je definirati i dualno, tako da svakom svijetu pridružuje skup propozicionalnih varijabli koje su u njemu istinite. Valuaciju onda možemo prirodno generalizirati tako da svakom svijetu pridružimo skup svih formula koje su u njemu istinite. Razlika između Kripkeovog modela i Hintikkine strukture je u tome što se u Hintikkinoj strukturi svijetu ne pridružuje skup bilo kakvih formula, nego samo onih koje pripadaju danom skupu Θ.

Definicija 3.4.5. *Za formulu osnovnog modalnog jezika φ kažemo da je* ***ispunjiva u Hintikkinoj strukturi*** *$\mathcal{H} = (W, R, H)$ za skup Θ ako je $\varphi \in H(w)$ za neki $w \in W$. Skup formula $\Delta \subseteq \Theta$ je ispunjiv u H ako $\Delta \subseteq H(w)$ za neki $w \in W$.*

Uočimo kako je ispunjivost u Hintikkinoj strukturi vrlo slična ispunjivosti u modelu. Model nam daje istinosne vrijednosti za sve formule osnovnog modalnog jezika, dok Hintikkina struktura daje samo dovoljno saznanja kako bi se odredila ispunjivost početne formule ili skupa formula. Vidimo kako prelazak s modela na Hintikkinu strukturu za neki skup nije težak.

Lema 3.4.6. *Neka je dan zatvoren skup Θ i model $\mathfrak{M} = (W, R, V)$. Struktura $\mathcal{H} = (W, R, H)$ za koju je $H(w) = \{\varphi \in \Theta \mid \mathfrak{M}, w \Vdash \varphi\}$ za svaki $w \in W$, je jedna Hintikkina struktura za Θ.*

Sljedeći rezultat dat će nam vezu ispunjivosti formula osnovnog modalnog jezika u Kripkeovim modelima i Hintikkinim strukturama.

Teorem 3.4.7. *Formula φ osnovnog modalnog jezika je ispunjiva ako i samo ako je ispunjiva u nekoj Hintikkinoj strukturi za skup $ecl(\varphi)$.*

Dokaz. Jedan smjer ($\Rightarrow$) odmah slijedi iz leme 3.4.6 za skup $\Theta = ecl(\varphi)$.

Dokažimo obrat ($\Leftarrow$).

Pretpostavimo da je $\varphi \in H(w_0)$ za neki svijet w_0 u nekoj Hintikkinoj strukturi $\mathcal{H} = (W, R, H)$. Definiramo model $\mathfrak{M} = (W, R, V)$ tako da samo prenesemo skupove W i R a valuaciju V definiramo tako da za svaku propozicionalnu varijablu p i svaki svijet w stavimo $w \in V(p)$ ako i samo ako $p \in H(w)$.

Sada indukcijom po strukturi formule dokazujemo sljedeću tvrdnju:

Ako je $\varphi \in H(w)$ za neki $w \in W$, onda je $\mathfrak{M}, w \Vdash \varphi$.

Baza

Promatramo sljedeća četiri slučaja.

1. Za $\varphi = p$ izravno iz definicije V vrijedi $\mathfrak{M}, w \Vdash \varphi$.
2. Za $\varphi = \top$ izravno iz definicije istinitosti vrijedi $\mathfrak{M}, w \Vdash \varphi$.
3. Za $\varphi = \neg p$ vrijedi da kako je skup $H(w)$ Hintikkin skup, a pritom znamo da nije uočljivo inkonzistentan, nije moguće da je onda i $p \in H(w)$. Slijedi da je $w \notin V(p)$ a to je po definiciji istinitosti ekvivalentno $\mathfrak{M}, w \nVdash p$, odnosno $\mathfrak{M}, w \Vdash \varphi$.
4. Za $\varphi = \neg\bot$ izravno iz definicije istinitosti vrijedi $\mathfrak{M}, w \nVdash \bot$ što je uz definiciju istinitosti za $\neg$ ekvivalentno $\mathfrak{M}, w \Vdash \varphi$.

Pretpostavka: Pretpostavimo da tvrdnja indukcije vrijedi za svaku formulu složenosti manje od φ.

Korak

Promatramo slučajeve obzirom na oblik formule φ.

1. Za $\varphi = \varphi_1 \wedge \varphi_2$ po (1) iz definicije Hintikkine strukture znamo da su φ_1 i φ_2 elementi skupa $H(w)$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w \Vdash \varphi_1$ i $\mathfrak{M}, w \Vdash \varphi_2$ i po definiciji istinitosti za $\wedge$ vrijedi $\mathfrak{M}, w \Vdash \varphi$.
2. Za $\varphi = \neg(\varphi_1 \vee \varphi_2)$ po (1) iz definicije Hintikkine strukture znamo da su $\neg\varphi_1$ i $\neg\varphi_2$ elementi skupa $H(w)$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w \Vdash \neg\varphi_1$ i $\mathfrak{M}, w \Vdash \neg\varphi_2$. Po lemi 2.1.3 vrijedi $\mathfrak{M}, w \Vdash \varphi$.
3. Za $\varphi = \neg(\varphi_1 \to \varphi_2)$ po (1) iz definicije Hintikkine strukture znamo da su φ_1 i $\neg\varphi_2$ elementi skupa $H(w)$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w \Vdash \varphi_1$ i $\mathfrak{M}, w \Vdash \neg\varphi_2$. Po lemi 2.1.3 vrijedi $\mathfrak{M}, w \Vdash \varphi$.
4. Za $\varphi = \neg\neg\psi$ po (1) iz definicije Hintikkine strukture znamo da je ψ element skupa $H(w)$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w \Vdash \psi$. Po lemi 2.1.3 vrijedi $\mathfrak{M}, w \Vdash \varphi$.

5. Za $\varphi = \neg(\varphi_1 \wedge \varphi_2)$ po (1) iz definicije Hintikkine strukture znamo da je $\neg\varphi_1$ ili $\neg\varphi_2$ element skupa $H(w)$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w \Vdash \neg\varphi_1$ ili $\mathfrak{M}, w \Vdash \neg\varphi_2$. Po lemi 2.1.3 vrijedi $\mathfrak{M}, w \Vdash \varphi$.

6. Za $\varphi = \varphi_1 \vee \varphi_2$ po (1) iz definicije Hintikkine strukture znamo da je φ_1 ili φ_2 element skupa $H(w)$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w \Vdash \varphi_1$ ili $\mathfrak{M}, w \Vdash \varphi_2$ i po definiciji istinitosti za $\vee$ vrijedi $\mathfrak{M}, w \Vdash \varphi$.

7. Za $\varphi = \varphi_1 \rightarrow \varphi_2$ po (1) iz definicije Hintikkine strukture znamo da je $\neg\varphi_1$ ili φ_2 element skupa $H(w)$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w \Vdash \neg\varphi_1$ ili $\mathfrak{M}, w \Vdash \varphi_2$ i po definiciji istinitosti za $\rightarrow$ vrijedi $\mathfrak{M}, w \Vdash \varphi$.

8. Za $\varphi = \Diamond\psi$ po (2) iz definicije Hintikkine strukture znamo da postoji svijet w' takav da vrijedi wRw' i $\psi \in H(w')$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w' \Vdash \psi$. Iz definicije istinitosti za $\Diamond$ slijedi $\mathfrak{M}, w \Vdash \varphi$.

9. Za $\varphi = \neg\Box\psi$ po (2) iz definicije Hintikkine strukture znamo da postoji svijet w' takav da vrijedi wRw' i $\neg\psi \in H(w')$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w' \Vdash \neg\psi$. Iz definicije istinitosti za $\Box$ i $\neg$ slijedi $\mathfrak{M}, w \Vdash \varphi$.

10. Za $\varphi = \Box\psi$ po (3) iz definicije Hintikkine strukture znamo da za svaki svijet w' takav da je wRw' vrijedi $\psi \in H(w')$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w' \Vdash \psi$ za svaki takav w'. Iz definicije istinitosti za $\Box$ slijedi $\mathfrak{M}, w \Vdash \varphi$.

11. Za $\varphi = \neg\Diamond\psi$ po (3) iz definicije Hintikkine strukture znamo da za svaki svijet w' takav da je wRw' vrijedi $\neg\psi \in H(w')$. Po pretpostavci indukcije vrijedi $\mathfrak{M}, w' \Vdash \neg\psi$ za svaki takav w'. Iz definicije istinitosti za $\Diamond$ i $\neg$ slijedi $\mathfrak{M}, w \Vdash \varphi$.

Po principu matematičke indukcije dokazali smo da su sve formule Hintikkine strukture $\mathcal{H}$ ispunjive u modelu $\mathfrak{M}$. Kako je $\varphi \in H(w_0)$ posebno je i ona ispunjiva u $\mathfrak{M}$. Time je dokazan i ($\Leftarrow$) smjer. $\Box$

Korolar 3.4.8. *Konačan skup Θ osnovnog modalnog jezika je ispunjiv u modelu ako i samo ako je ispunjiv u Hintikkinoj strukturi za skup $ecl(\Theta)$.*

Sada imamo vezu ispunjivosti tih dviju struktura. Veoma smo blizu dokaza potpunosti. Jedino što nedostaje je veza Hintikkine strukture i završnog tableaua iz naše metode.

Teorem 3.4.9. *Neka je dan skup formula Θ osnovnog modalnog jezika. Tada za svaki otvoren završni tableau $\mathcal{T}^\Theta = (V, E)$ i njegov skup stanja S^Θ izgradimo strukturu $\mathcal{H}^\Theta = (W, R, H)$ ovako:*

1. *Svakom elementu skupa V maknemo oznaku i dodamo ga skupu W;*

2. *svakom elementu skupa E maknemo oznaku i dodamo ga skupu R;*

3. *za svaki vrh $w \in W$ dodajemo oznaku tog vrha iz skupa S^Θ skupu $H(w)$.*

Tada je struktura $\mathcal{H}^\Theta$ Hintikkina struktura za skup $ecl(\Theta)$.

Dokaz. Kako bismo dokazali da je $\mathcal{H}^\Theta$ zbilja Hintikkina struktura potrebno je pokazati da vrijede sve tri tvrdnje iz definicije 3.4.4. U dokazu tih tvrdnji implicitno se koristi i činjenica da je struktura nastala iz otvorenog završnog tableaua i preživjeli su svi potrebni svjetovi.

Prvo pokazujemo tvdnju (1):

Za svaki $w \in W$ vrijedi po proceduri **PredProširenje** kako je skup $H(w)$ jedno potpuno proširenje. Svako potpuno proširenje je ujedno i Hintikkin skup te je i podskup skupa $ecl(\Theta)$. Time odmah slijedi tvrdnja (1).

Zatim pokazujemo tvrdnju (2):

Za svaki $w \in W$ i $\varphi \in H(w)$ egzistencijalnu nasljednu formulu vrijedi: Iz procedure **Sljedbenik** znamo da je postojalo predstanje Γ čiji element je bila nasljedna komponenta formule φ. Također iz procedure **PredProširenje** znamo da sigurno postoji $w' \in W$ za koji je ta nasljedna komponenta element skupa $H(w')$. Naposljetku, procedura **UklanjanjePredstanja** nam daje kako za taj w' vrijedi još i wRw'. Iz svega navedenog odmah slijedi tvrdnja (2).

Naposljetku, pokazujemo tvrdnju (3):

Za svaki $w \in W$ i $\varphi \in H(w)$ univerzalnu nasljednu formulu vrijedi: Iz definicije skupa X i procedure **Sljedbenik** znamo da je postojalo predstanje Γ čiji element je bila nasljedna komponenta formule φ. Također iz procedure **PredProširenje** znamo da sigurno postoje svjetovi $w' \in W$ za koje je ta nasljedna komponenta element skupa $H(w')$. Naposljetku, procedura **UklanjanjePredstanja** nam daje kako za svaki takav w' vrijedi još i wRw'. Iz svega navedenog odmah slijedi tvrdnja (3).

Slijedi da je struktura $\mathcal{H}^\Theta$ jedna Hintikkina struktura. □

Imamo sve što nam je potrebno kako bismo dokazali teorem potpunosti.

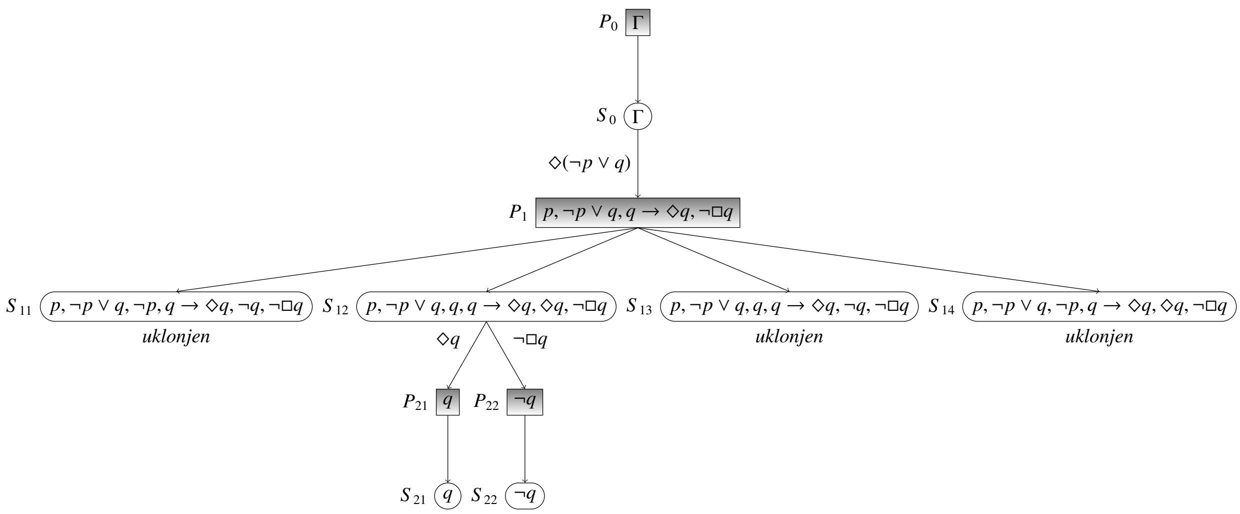
Teorem 3.4.10. *Neka je dan konačan skup formula Θ osnovnog modalnog jezika i otvoren završni tableau $\mathcal{T}^\Theta$. Tada je skup Θ ispunjiv.*

Dokaz. Za dani završni tableau $\mathcal{T}^\Theta$ iz teorema 3.4.9 slijedi da imamo Hintikkinu strukturu $\mathcal{H}$ za skup $ecl(\Theta)$. Znamo da je zbog otvorenosti završnog tableaua $\mathcal{T}^\Theta$ skup Θ sigurno oznaka nekog svijeta iz strukture $\mathcal{H}$. Time smo iz definicije 3.4.5 dobili da je skup Θ ispunjiv u strukturi $\mathcal{H}$. Iz korolara 3.4.8 slijedi tvrdnja. $\square$

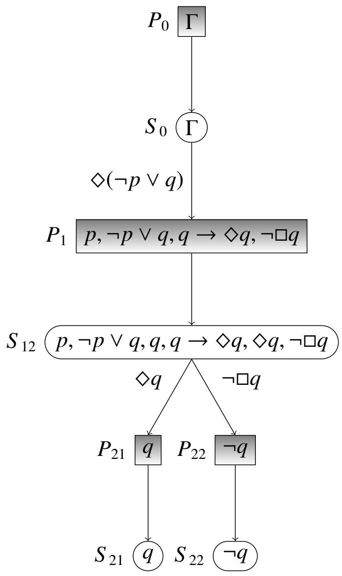
Dokazi ova dva teorema nam otvaraju mogućnost korištenja naše tableaux metode za pronalazak odgovora u sljedećim primjerima.

Primjer 3.4.11. *U ovom primjeru ćemo primjenom metode tableauxa ispitati vrijedi li* $\{\Box p, \Diamond(\neg p \vee q), \Box(q \to \Diamond q)\} \Vdash \Diamond\Box q$*?*
Nalazimo završni tableau za skup $\Gamma = \{\Box p, \Diamond(\neg p \vee q), \Box(q \to \Diamond q), \neg\Diamond\Box q\}$.

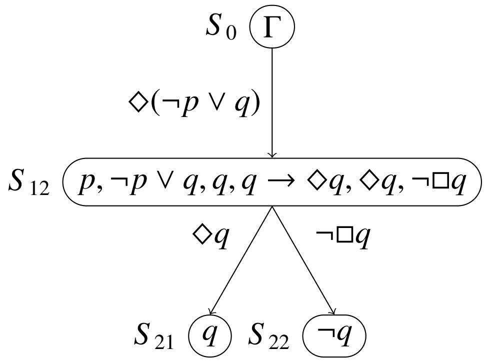
Ovdje dajemo pripadni pretableau $\mathcal{P}^\Gamma$*:*



Ovako izgleda pretableau $\mathcal{P}^\Gamma$ *nakon završetka izgradnje:*



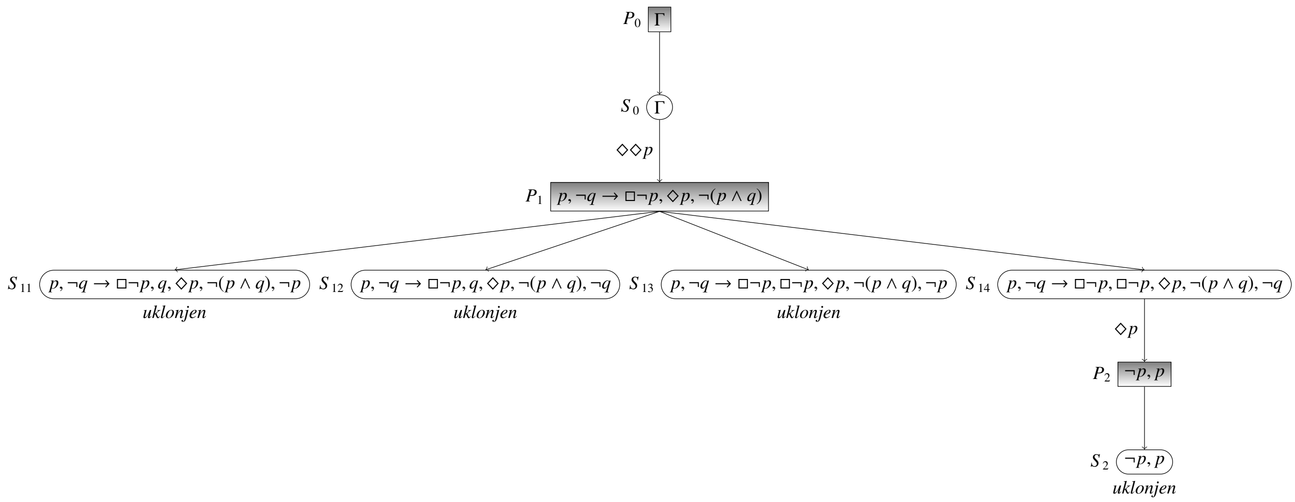
Sada dajemo pripadni inicijalni tableau:



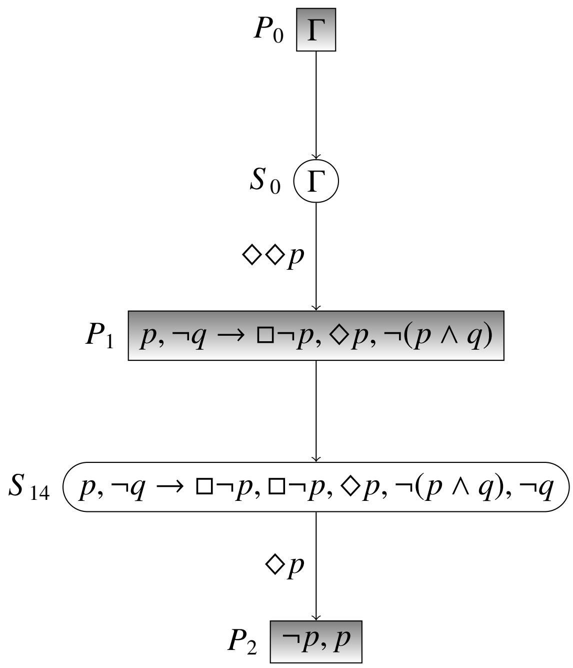
Završni tableau jednak je inicijalnom te kako je skup Γ član stanja S_0, tableau je otvoren. Slijedi da je skup Γ ispunjiv i vrijedi $\{\Box p, \Diamond(\neg p \vee q), \Box(q \rightarrow \Diamond q)\} \nvDash \Diamond\Box q$.

Primjer 3.4.12. *U ovom primjeru ćemo primjenom metode tableauxa ispitati vrijedi li $\{\Box p, \Box(\neg q \rightarrow \Box\neg p), \Diamond\Diamond p\} \vDash \Diamond(p \wedge q)$?*
Nalazimo završni tableau za skup $\Gamma = \{\Box p, \Box(\neg q \rightarrow \Box\neg p), \Diamond\Diamond p, \neg\Diamond(p \wedge q)\}$.

Ovdje dajemo pripadni pretableau $\mathcal{P}^\Gamma$:



Ovako izgleda pretableau $\mathcal{P}^\Gamma$ nakon završetka izgradnje:



Sada dajemo pripadni inicijalni tableau:

$$\begin{tikzcd}
S_0 \; (\Gamma) \arrow[d, "\Diamond\Diamond p"'] \\
S_{14} \; (p, \neg q \to \Box\neg p, \Box\neg p, \Diamond p, \neg(p \wedge q), \neg q)
\end{tikzcd}$$

Završni tableau za skup Γ *je prazan skup, jer su jedina preostala stanja neostvariva i uklonjena. Dakle, tableau je zatvoren. Slijedi da skup* Γ *nije ispunjiv i vrijedi* $\{\Box p, \Box(\neg q \to \Box\neg p), \Diamond\Diamond p\} \vDash \Diamond(p \wedge q)$.

Poglavlje 4

Odlučivost

U ovom nam je poglavlju cilj definirati što je to odlučivost i odgovoriti na pitanje je li problem ispunjivosti skupa formula odlučiv u logici sudova, odnosno u osnovnom modalnom jeziku.

4.1 Osnove

Pojam odlučivosti često se veže uz neki problem. U našem radu bavimo se problemom ispunjivosti skupa formula. Odgovor na ovaj problem za svaki pojedini skup formula je uvijek da ili ne.

Ovakvim problemima ali i drugim, bavi se teorija izračunljivosti.

Glavni pojmovi te teorije su **algoritam**, **izračunljivost** i **model izračunavanja**.

Teorije se dijele prema modelu izračunavanja kojeg koriste. Neki od poznatijih su **Turingov stroj**, **λ-račun**, **parcijalno rekurzivne funkcije** i **RAM strojevi**. Najstariji i najšire prihvaćen je Turingov stroj. Jedan od glavnih rezultata u teorijama koje koriste ostale modele je upravo dokazati ekvivalenciju s Turingovim strojem. Mi ćemo se u našim razmatranjima opredijeliti za njega.

Za neki alfabet Γ, **jezik**, u oznaci L, je bilo koji podskup skupa Γ^*.

Kada malo preciziramo, naši se problemi mogu zadati kao sljedeća dva jezika.

$SAT_{PL} = \{\Theta \mid \Theta$ je ispunjiv skup formula logike sudova.$\}$
$SAT_{BML} = \{\Theta \mid \Theta$ je ispunjiv skup formula osnovnog modalnog jezika.$\}$

Definicija 4.1.1. ***Turingov stroj*** *je uređena sedmorka $(Q, \Sigma, \Gamma, \delta, q_0, q_{DA}, q_{NE})$, gdje je redom:*

- *Q konačan skup čije elemente nazivamo* ***stanja****;*

- Σ *konačan skup, čije elemente nazivamo* ***ulazni simboli****. Pretpostavljamo da* Σ *ne sadrži "prazan simbol", kojeg označavamo s* $\sqcup$*;*

- Γ *konačan skup kojeg nazivamo* ***alfabet Turingovog stroja****. Pretpostavljamo* $\sqcup \in \Gamma$ *i* $\Sigma \subseteq \Gamma$*;*

- $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, D, S\}$ *je proizvoljna funkcija koju nazivamo* ***funkcija prijelaza****;*

- $q_0 \in Q$ *i nazivamo ga* ***početno stanje****;*

- $q_{DA} \in Q$ *i nazivamo ga* ***stanje prihvaćanja****;*

- $q_{NE} \in Q$ *i nazivamo ga* ***stanje odbijanja****.*

Turingov stroj sadrži glavu i beskonačnu traku za pisanje i čitanje. Glava u svakom trenutku može zapisati ili pročitati točno jedan simbol alfabeta. Skup $\{L, D, S\}$ označava pomak glave u određenom koraku izvršavanja stroja: L je pomak za jedno mjesto ulijevo, D za jedno mjesto udesno i S bez pomaka taj korak. Istaknuli smo dva završna stanja stroja. Ovakav tip stroja se uobičajeno naziva **odlučitelj**. Kako je za nas najbitnija klasa odlučivih jezika, ovakva definicija je dovoljna. Ukoliko se želi promatrati opće izračunljive funkcije, potrebno je prilagoditi gornju definiciju.

Definicija 4.1.2. *Kažemo da* ***Turingov stroj*** $T = (Q, \Sigma, \Gamma, \delta, q_0, q_{DA}, q_{NE})$ ***prepoznaje neku riječ*** $w \in \Gamma^*$ *ako postoji konačan niz parova* $(r_0, s_0)...(r_m, s_m) \in Q \times \Gamma$*, te konačan niz simbola* $I_1, ..., I_m \in \{L, D, S\}$ *tako da vrijedi:*

- $r_0 = q_0$ *i* s_0 *je prvi slijeva simbol riječi* w*;*

- *za svaki* $j \in 0, ..., m-1$ *imamo* $\delta(r_j, s_j) = (r_{j+1}, s_{j+1}, I_{j+1})$ *i* $r_j \notin \{q_{DA}, q_{NE}\}$*;*

- $r_m = q_{DA}$*.*

Za proizvoljan Turingov stroj T *s* $L(T)$ *označavamo skup svih riječi koje* T *prepoznaje. Kažemo da neki Turingov stroj* T ***prepoznaje jezik*** L *ako vrijedi* $L = L(T)$*.*

Prepoznavanje jezika je vrlo jako svojstvo, ali nije dovoljno kako bi za svaku riječ jednoznačno odgovorilo je li član tog jezika.

Kažemo da **Turingov stroj staje** s nekim danim ulazom, ako postoje nizovi kao u prethodnoj definiciji, pri čemu je $r_m = q_{DA}$ ili $r_m = q_{NE}$. Ako takvi nizovi ne postoje, kažemo da **Turingov stroj radi vječno**.

Primijetimo da ako Turingov stroj prepoznaje neki jezik L i dobije kao ulaz neki $w \in \Gamma^* \setminus L$, tada on uopće ne mora stati.

Definicija 4.1.3. *Za neki jezik $L \subseteq \Gamma^*$ kažemo da je **Turing-odlučiv**, ili kratko **odlučiv**, ako postoji Turingov stroj T koji ga prepoznaje, te za svaku riječ $w \in \Gamma^* \setminus L$ stroj T s ulazom w staje u završnom stanju q_{NE}. Ako neki jezik nije odlučiv tada još kažemo da je **neodlučiv**.*

Detaljniji pregled pitanja odlučivosti može se vidjeti u [11].

U daljnjim razmatranjima Turingove strojeve nećemo formalno zadavati, već ćemo samo dati njihove opise.

4.2 Odlučivost naših problema

Cilj ovog odjeljka je razmotriti pitanje odlučivosti naša dva problema, odnosno jezika.

Teorem 4.2.1. *Jezik SAT_{PL} jest odlučiv.*

Dokaz. Iz teorema adekvatnosti i potpunosti slijedi da je za svaki skup formula logike sudova činjenica da smo našom metodom dobili otvoreni tableau ekvivalentna odgovoru da je taj skup ispunjiv, odnosno da je činjenica da smo dobili zatvoren tableau ekvivalentna odgovoru da nije ispunjiv.

Za jezik SAT_{PL} naša metoda daje jednoznačan odgovor za svaki skup, i može se konstruirati Turingov stroj T koji simulira izgradnju odgovarajućeg tableaua. Slijedi da će takav Turingov stroj T za svaki skup kao ulazni podatak stati i vratiti jednoznačan odgovor. Po definiciji 4.1.3, jezik SAT_{PL} je odlučiv, što povlači da je problem ispunjivosti nekog skupa formula logike sudova odlučiv. □

Teorem 4.2.2. *Jezik SAT_{BML} jest odlučiv.*

Dokaz. Iz teorema adekvatnosti i potpunosti slijedi da je za svaki skup formula osnovnog modalnog jezika činjenica da smo našom metodom dobili otvoreni tableau ekvivalentna odgovoru da je taj skup ispunjiv, odnosno da je činjenica da smo dobili zatvoren tableau ekvivalentna odgovoru da nije ispunjiv.

Za jezik SAT_{BML} naša metoda daje jednoznačan odgovor za svaki skup, i može se konstruirati Turingov stroj T koji simulira izgradnju odgovarajućeg tableaua. Slijedi da će takav Turingov stroj T za svaki skup kao ulazni podatak stati i vratiti jednoznačan odgovor. Po definiciji 4.1.3, jezik SAT_{BML} je odlučiv, što povlači da je problem ispunjivosti nekog skupa formula osnovnog modalnog jezika odlučiv. □

Vrijedi napomenuti kako za logiku prvog reda ovakav teorem ne vrijedi. Ovo su prvi formalno dokazali nezavisno jedan od drugoga upravo Turing (Vidi [9]) i Church[1] (Vidi [2]).

[1] Alonzo Church (1903. – 1995.). Američki matematičar i logičar.

Bibliografija

[1] P. Blackburn, M. de Rijke i Y. Venema, *Modal Logic*, Cambridge University Press, 2001.

[2] A. Church, *An Unsolvable Problem of Elementary Number Theory*, American Journal of Mathematics **58** (1936), br. 2, 345–363.

[3] M. D'Agostino, D. M. Gabbay, R. Hähnle i J. Posegga, *Handbook of Tableau Methods*, Springer, 1999.

[4] S. Demri, V. Goranko i M. Lange, *Temporal Logics in Computer Science: Finite-State Systems*, Cambridge University Press, 2016.

[5] K. Gödel, *Collected Works: Volume I: Publications 1929-1936*, Oxford University Press, 1986.

[6] S. A. Kripke, *A Completeness Theorem in Modal Logic*, Journal of Symbolic Logic **24** (1959), 1–14.

[7] C. I. Lewis, *A Survey of Symbolic Logic*, University of California Press, 1918.

[8] R. M. Smullyan, *First-order Logic*, Dover, 1995.

[9] A. M. Turing, *On Computable Numbers, with an Application to the Entscheidungsproblem*, Proceedings of the London Mathematical Society **s2-42** (1937), br. 1, 230–265.

[10] M. Vuković, *Matematička logika*, Element, 2009.

[11] M. Vuković, *Složenost algoritama, skripta*, PMF - Matematički odsjek, Zagreb, 2015.

Sažetak

Ovaj rad proučava jednu od metoda za dobivanje modela logičkih formula koju nazivamo metoda tableauxa ili glavni test. Zanimljiva nam je zbog algoritamskog pristupa kojim je možemo opisati i time vrlo lako prenijeti na računalo. Od posebnog značaja je opis te metode za vrstu logike znanu kao modalna logika. Rad je podijeljen na četiri poglavlja.

Prvo poglavlje upoznaje nas sa sintaksom i semantikom osnovnog modalnog jezika. Ističemo sličnosti i razlike s logikom sudova, ponajprije u pojmu istinitosti formula. Ovdje veliku ulogu imaju modalni operatori koji su detaljno obrađeni.

Kraj prvog poglavlja iznosi cilj rada: dobivanje efikasnog postupka (algoritma) za pronalazak modela u kojem je neka formula osnovnog modalnog jezika ispunjiva. Također, navodimo vezu ispunjivosti i valjanosti formula i zaključujemo kako je naš algoritam moguće primijeniti i na ispitivanje ispunjivosti i na ispitivanje valjanosti dane formule.

Drugo poglavlje započinje klasifikacijom formula u nama zanimljive skupine. Ovakav pristup prvi je upotrijebio Smullyan u [8]. On spominje α i β klase formula koje mi malo drugačije nazivamo, ali ideja je ista.

Nastavlja se uvođenjem pravila metode tableauxa za logiku sudova i osnovnih pojmova potrebnih za njeno shvaćanje. Bitno je spomenuti kako su ova pravila sintaktičkog oblika, a semantičko značenje dajemo na kraju pogavlja dokazujući dva vrlo važna teorema: adekvatnost i potpunost.

Adekvatnost metode tableauxa povezuje pojmove zatvorenog stabla i neispunjivosti formule za koju je to semantičko stablo izgrađeno, štoviše, ako za formulu imamo zatvoreno stablo, tada je ona sigurno neispunjiva.

Potpunost ide obrnutim smjerom, za neispunjivu formulu iz ovog teorema slijedi da se sva semantička stabla zatvore.

Dobiveni rezultat nam daje slobodu u primjeni sintaktičkog algoritma, recimo na računalu, a semantička interpretacija ne izostaje.

Treće poglavlje proširuje naše klase još jednom, a to su nasljedne formule. Ova klasa sadrži sve formule dobivene primjenom modalnog operatora na neku formulu osnovnog modalnog jezika i njihove negacije. Uvodi se pojam proširenog zatvorenja i dokazuje njegova konačnost.

U nastavku iznosimo prilagođenu metodu tableauxa korištenjem one iz drugog poglavlja i pokazujemo njezine nedostatke.

Navodimo pravila nove metode tableauxa kojom se rješavamo tih nedostataka. Metoda iz drugog poglavlja važan je dio ove naše nove metode te razumijevanje drugog poglavlja uvelike pomaže u razumijevanju trećeg. Kao i u drugom poglavlju, ova pravila su sintaktičkog oblika i lako primjenjiva na računalu.

Nastavak trećeg poglavlja sadrži dokaz adekvatnosti i potpunosti naše nove metode za osnovni modalni jezik. Od posebnog značaja su pojmovi Hintikkine strukture i označenog grafa. Dokazom tih dvaju teorema ponovno je dobivena veza sintakse i semantike, ali za kompliciraniji osnovni modalni jezik.

Četvrto poglavlje proučava pitanje odlučivosti. Definiramo Turingov stroj. U nastavku se definira Turing-odlučivost i pokazuje kako su problemi koje rješavaju obje metode predstavljene u ovom radu odlučivi, za razliku od analognog problema u logici prvog reda.

Posebno, želio bih se zahvaliti svojem mentoru, doc. dr. sc. Tinu Perkovu na iznimnom strpljenju i ne malom broju sugestija bez kojih ovaj rad ne bi bio moguć. Također, zahvaljujem se i suvoditelju, izv. prof. dr. sc. Mladenu Vukoviću na vrlo zanimljivim predavanjima kojima je i potaknuo moju želju za proširenjem znanja o logici bez kojih ne bi bilo ovog rada.

Summary

In this thesis we examine one of the methods of acquiring a model for logical formulae known as tableaux. We are interested in it because of an algorithmic approach with which it can be described and easily transferred to a computer. Special attention is given to describing said method for a particular type of logic known as modal logic. The thesis consists of four chapters.

In the first chapter we are introduced to the syntax and semantics of basic modal language. We highlight the similarities and differences with propositional logic, especially in the definition of truth. Great care is taken to explain modal operators as they play an essential part of the basic modal language.

The end of first chapter states the goal of this thesis: providing an effective algorithm for finding a model in which a formula of the basic modal language is satisfiable. Moreover, we give a link between the concept of satisfiability and validity of formulae and conclude the effectiveness of our algorithm for finding both.

Second chapter begins with a classification of formulae to relevant groups. This approach was first introduced by Smullyan in [8]. He uses α and β classes of formulae that we denote slightly different, albeit the main concept stays the same.

We continue by introducing the rules of the tableaux method for propositional logic and other basic concepts needed. We emphasize that these rules are strictly syntactic in nature. The semantic meaning follows at the end of the chapter from the proof of two very important theorems: soundness and completeness.

Soundness of the tableaux method links the concept of closed tree and unsatisfiability of formulae of which said tree is built from, moreover, it follows that if a formula has a closed tree, then it is unsatisfiable.

Completeness gives us the converse: for an unsatisfiable formula all trees must close.

This result gives us the freedom to use the syntactic rules of the algorithm, for example on a computer, and the implied semantic meaning stays with it.

Third chapter expands our classification by one more class, and that is successor formulae. This class consists of all formulae built by applying a modal operator on some formula of the basic modal language and their negations. We introduce the concept of expanded closure and we prove that it is finite.

Continuing on, we state a modified tableaux method from chapter two and show its drawbacks.

We state the rules of a new tableaux method that will strive to fix the drawbacks mentioned earlier. The tableaux method from chapter two is an integral part of our new method. As such, understanding of concepts in chapter two is imperative for the proper understanding in chapter three. As in chapter two, these rules are syntactic in nature and easily transferable to a computer.

The continuation of chapter three consists of the proof of soundness and completeness of our new method for basic modal language. Important concepts to remember are Hintikka structures and labelled graphs. By concluding these proofs we have given a link between the syntactic rules and semantic meaning, but for a more complicated, basic modal language.

Chapter four deals with the question of decidability. We state what a Turing machine and Turing-decidability are. We then proceed to show that both of the problems solved by the methods examined here are decidable in contrast to the same problem in first order logic which is not decidable.

Finally, I would like to thank my mentor, assist. prof. Tin Perkov for his boundless patience and a variety of suggestions that have made this thesis possible. Furthermore, I would also like to thank my co-mentor, assoc. prof. Mladen Vuković, for very interesting lectures that made me want to expand my knowledge of logic, without which this thesis would not be possible.

Životopis

Rođen sam 07. 03. 1993. godine u Zagrebu. Osnovnu školu pohađam od 1999. godine upisujući Osnovnu školu Julija Klovića u Zagrebu. Istu završavam 2007. godine i upisujem V. Gimnaziju u Zagrebu. Preddiplomski studij matematike, smjer nastavnički, upisujem 2011. godine na Matematičkom odsjeku Prirodoslovno-matematičkog fakulteta u Zagrebu. Akademski naziv prvostupnika edukacije matematike stječem 2015. godine i upisujem diplomski studij Računarstvo i matematika na istom odsjeku. Tijekom treće i četvrte godine volontiram u studentskoj udruzi eSTUDENT. Na petoj godini zapošljavam se preko student servisa u firmi SysKit d.o.o. kao praktikant programer i u istoj trenutno radim.