Логика (радна верзија)

Небојша Икодиновић & Славко Моцоња

Садржај

Глава 1

Исказна логика

1. Синтакса исказне логике

Синтакса исказне логике одређена је *исказним језиком*, тј. колекцијом симбола који служе за запис *исказних формула*, као и правилима по којима се исказне формуле граде.

Дефиниција 1.1. (Исказни језик и исказна формула)

1. *Исказни језик* чине следећи симболи:
   - *исказна слова* или *исказне променљиве* које обично обележавамо са $p, q, r, \ldots, p_0, p_1, p_2, \ldots$ и слично,
   - *логичке константе* $\bot$ и $\top$,
   - *логички везници* $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ и $\veebar$, и
   - заграде као помоћни симболи.
2. *Исказне формуле* градимо рекурентно користећи следећа правила:
   - исказна слова и логичке константе су исказне формуле;
   - ако су φ и ψ исказне формуле, онда су и $\neg\varphi$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \leftrightarrow \psi)$ и $(\varphi \veebar \psi)$ такође исказне формуле;
   - свака исказна формула се гради у коначно много корака користећи претходна два правила.

Скуп исказних слова обележавамо са $\mathcal{P}$, и, ако није другачије наглашено, претпостављамо да је $\mathcal{P}$ бесконачно пребројив скуп. У овој глави ћемо уместо „исказна формула” обично говорити само „формула”. Формуле ћемо по правилу обележавати малим словима грчког алфабета. Скуп свих формула обележаваћемо са $\mathcal{F}$.

Задатак 3.1. 1° Ако је $\mathcal{P}$ највише пребројив (коначан или пребројив) скуп, доказати да је $\mathcal{F}$ пребројив скуп.
2° Ако је $|\mathcal{P}| = \kappa \geqslant \aleph_0$, доказати да је $|\mathcal{F}| = \kappa$.
(*Помоћ за део б): Ако је $\kappa \geqslant \aleph_0$, тада је $\kappa \cdot \kappa = \kappa$.*)

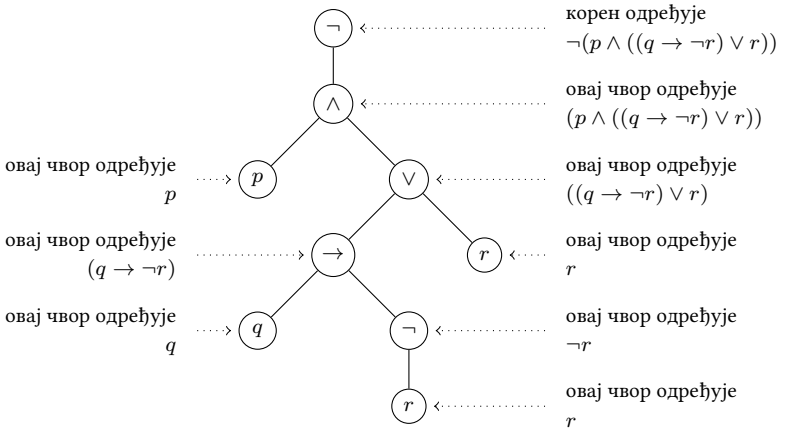
Интуитивно је јасно да је свака формула $\varphi \in \mathcal{F}$ записана користећи само коначно много симбола. Специјално, φ у свом запису има само коначно много исказних слова; *коначан* скуп исказних слова која се јављају у запису формуле φ обележавамо са $\mathcal{P}_\varphi$. Слично, у запису формуле φ логички везници појављују се само коначно много пута. Са $\mathrm{sl}(\varphi)$ обележавамо број појављивања исказних везника у φ. Дакле, $\mathrm{sl}(\varphi) \in \mathbb{N}$, и овај број називамо *сложеност формуле* φ. Приметимо да је $\mathrm{sl}(\varphi) = 0$ ако и само ако је φ исказно слово или логичка константа. Такође, $\mathrm{sl}(\neg\varphi) = 1 + \mathrm{sl}(\varphi)$, $\mathrm{sl}((\varphi \wedge \psi)) = \mathrm{sl}((\varphi \vee \psi)) = \mathrm{sl}((\varphi \to \psi)) = \mathrm{sl}((\varphi \leftrightarrow \psi)) = \mathrm{sl}((\varphi \veebar \psi)) = \mathrm{sl}(\varphi) + \mathrm{sl}(\psi) + 1$.

Задатак 3.2. Ако је $\mathrm{sl}(\varphi) = n$, доказати да је укупан број симбола у формули φ највише $4n + 1$. Такође, доказати да је број слова у формули φ највише $n + 1$.

Сваку формулу можемо представити њеним *дрветом изградње*. Дрво изградње формуле φ можемо описати рекурентно пратећи поступак изградње на следећи начин. Исказном слову и логичким константама одговара једночлано дрво: чвор који је означен датим исказним словом, односно логичком константом. Ако је $\varphi = \neg\psi$, дрво формуле φ има корен означен са $\neg$ на које се надовезује дрво формуле ψ. Ако је $\varphi = (\psi * \theta)$, где је $*$ неки од везника $\wedge$, $\vee$, $\to$, $\leftrightarrow$ и $\veebar$, дрво формуле φ има корен означен са $*$ на које се надовезују два поддрвета: редом дрва формуле ψ и формуле θ.

Формула ψ је *потформула* формуле φ ако је $\psi = \varphi$ или ако је формула ψ учествовала у изградњи формуле φ. Лако можемо да видимо да у дрвету изградње формуле φ сваком чвору дрвета одговара једна потформула од φ.

Пример 1.2. Дрво изградње формуле $\neg(p \wedge ((q \to \neg r) \vee r))$ приказано је на следећој слици.



Често ћемо, како бисмо растеретили запис, и ако је то могуће, испуштати да запишемо вишак заграда у формули. При томе водимо рачуна о познатом правилу о приоритету везника: $\neg$ је везник највишег приоритета, $\wedge$ и $\vee$ средњег, а $\to$, $\leftrightarrow$ и $\veebar$ су везници најнижег приоритета. Тако је нпр. $p \wedge q \to r \vee s$ формула $((p \wedge q) \to (r \vee s))$. Такође, због семантичке асоцијативности везника $\wedge$ и $\vee$, заграде у формулама $p \wedge q \wedge r$ или $p_1 \vee p_2 \vee \cdots \vee p_n$ нећемо записивати. Слично важи и за везник $\veebar$. Иако везник $\leftrightarrow$ има исту особину, није обичај да се у сличним формулама заграде бришу.

2. Семантика исказне логике

Скуп $\{0, 1\}$ је скуп истинитосних вредности, при чему 0 читамо као *нетачно*, а 1 као *тачно*.

Дефиниција 1.3. (Валуација) *Валуација* је било која функција $v : \mathcal{P} \to \{0, 1\}$.

Дакле, било које додељивање истинитосних вредности исказним словима назива се валуација. Када имамо задату валуацију v, на природан начин рачунамо истинитосне вредности свих формула.

Дефиниција 1.4. (Интерпретација формуле) Нека је $v : \mathcal{P} \to \{0, 1\}$ произвољна валуација. Дефинишемо *интерпретацију формуле* φ *при* v, у ознаци $\varphi[v]$, рекурзијом по изградњи формуле на следећи начин:

- $\top[v] \stackrel{def}{=} 1$ и $\bot[v] \stackrel{def}{=} 0$;
- за $p \in \mathcal{P}$, $p[v] \stackrel{def}{=} v(p)$;
- $(\neg\varphi)[v] \stackrel{def}{=} 1$ ако и само ако $\varphi[v] = 0$;
- $(\varphi \wedge \psi)[v] \stackrel{def}{=} 1$ ако и само ако важе оба $\varphi[v] = 1$ и $\psi[v] = 1$;
- $(\varphi \vee \psi)[v] \stackrel{def}{=} 1$ ако и само ако важи бар један од $\varphi[v] = 1$ и $\psi[v] = 1$;
- $(\varphi \to \psi)[v] \stackrel{def}{=} 1$ ако и само ако важи бар један од $\varphi[v] = 0$ и $\psi[v] = 1$;
- $(\varphi \leftrightarrow \psi)[v] \stackrel{def}{=} 1$ ако и само ако важи $\varphi[v] = \psi[v]$;
- $(\varphi \veebar \psi)[v] \stackrel{def}{=} 1$ ако и само ако важи $\varphi[v] \neq \psi[v]$.

Прва тачка претходне дефиниције каже да се формула $\top$ интерпретира као тачна, а формула $\bot$ као нетачна независно од валуације v. Друга тачка каже да слово p посматрано као формула има истинитосну вредност дату валуацијом v: $v(p)$. Трећа тачка каже да $\neg\varphi$ има супротну истинитосну вредност при v у односу на формулу φ; то можемо представити *истинитосном таблицом* на следећи начин:

$\varphi[v]$	$(\neg\varphi)[v]$
0	1
1	0

Коначно последње четири тачке кажу да се истинитосне вредности формула $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \to \psi$, $\varphi \leftrightarrow \psi$ и $\varphi \veebar \psi$ рачунају према следећим истинитосним таблицама:

$\varphi[v]$	$\psi[v]$	$(\varphi \wedge \psi)[v]$	$(\varphi \vee \psi)[v]$	$(\varphi \to \psi)[v]$	$(\varphi \leftrightarrow \psi)[v]$	$(\varphi \veebar \psi)[v]$
0	0	0	0	1	1	0
0	1	0	1	1	0	1
1	0	0	1	0	0	1
1	1	1	1	1	1	0

Интуитивно је јасно да интерпретација формуле φ при валуацији v зависи само од вредности валуације v у словима формуле φ:

Теорема 1.5. Вредност $\varphi[v]$ зависи само од $v|_{\mathcal{P}_\varphi}$[1].

Претходна теорема формално се доказује индукцијом по сложености формуле φ, што остављамо као задатак.

Задатак 3.3. Дати формалан доказ теореме 1.5.

Дефиниција 1.6. Нека су $\varphi, \psi_1, \ldots, \psi_n$ формуле.

1. Пишемо $\varphi = \varphi(p_1, p_2, \ldots, p_n)$ да нагласимо $\mathcal{P}_\varphi \subseteq \{p_1, p_2, \ldots, p_n\}$.
2. Ако је $\varphi = \varphi(p_1, \ldots, p_n)$, са $\varphi(\psi_1, \ldots, \psi_n)$ обележавамо формулу φ у којој смо свако појављивање слова p_i заменили са формулом ψ_i, за све $i \leqslant n$.

Лема 1.7. (Лема о замени)
Нека су $\varphi = \varphi(p_1, \ldots, p_n), \psi_1, \ldots, \psi_n$ формуле. Нека је v произвољна валуација, и нека је w валуација слова $p_1, \ldots, p_n$ дата са:

$$w(p_i) \overset{def}{=} \psi_i[v], \text{ за све } i \leqslant n.$$

Тада $\varphi(\psi_1, \ldots, \psi_n)[v] = \varphi[w]$.

Доказ. Изводимо доказ индукцијом по $\mathrm{sl}(\varphi)$. Ако је $\mathrm{sl}(\varphi) = 0$, имамо три случаја да размотримо.

1° φ је слово. Тада је обавезно $\varphi = p_i$ за неко $i \leqslant n$ (јер је $\varphi = \varphi(p_1, \ldots, p_n)$), па је $\varphi(\psi_1, \ldots, \psi_n) = \psi_i$. Сада је:

$$\varphi(\psi_1, \ldots, \psi_n)[v] = \psi_i[v] = w(p_i) = \varphi[w],$$

где друга једнакост важи по дефиницији валуације w.

2° $\varphi = \bot$. Тада је $\varphi(\psi_1, \ldots, \psi_n) = \bot$, Одакле $\varphi(\psi_1, \ldots, \psi_n)[v] = 0 = \varphi[w]$.

3° $\varphi = \top$. Овај случај се ради слично као 2°.

[1] $v|_{\mathcal{P}_\varphi}$ је ознака за рестрикцију валуације v на скуп $\mathcal{P}_\varphi$.

Претпоставимо сада да је $\mathrm{sl}(\varphi) = m > 0$. Имамо неколико случајева да размотимо.

1° $\varphi = \neg\theta$. Приметимо да је $\theta = \theta(p_1, \ldots, p_n)$, $\mathrm{sl}(\theta) = m - 1$ и $\varphi(\psi_1, \ldots, \psi_n) = \neg\theta(\psi_1, \ldots, \psi_n)$. Како је $\mathrm{sl}(\theta) < m$ по индукцијској хипотези је:

$$\theta(\psi_1, \ldots, \psi_n)[v] = \theta[w],$$

па је $\varphi(\psi_1, \ldots, \psi_n)[v] = 1$ акко $\theta(\psi_1, \ldots, \psi_n)[v] = 0$ акко $\theta[w] = 0$ акко $\varphi[v] = 1$; одавде, $\varphi(\psi_1, \ldots, \psi_n)[v] = \varphi[w]$.

2° $\varphi = \theta \wedge \sigma$. Приметимо да је $\theta = \theta(p_1, \ldots, p_n)$, $\sigma = \sigma(p_1, \ldots, p_n)$, $\mathrm{sl}(\theta), \mathrm{sl}(\sigma) < n$ и $\varphi(\psi_1, \ldots, \psi_n) = \theta(\psi_1, \ldots, \psi_n) \wedge \sigma(\psi_1, \ldots, \psi_n)$. Како је $\mathrm{sl}(\theta), \mathrm{sl}(\sigma) < m$, по индукцијској хипотези је:

$$\theta(\psi_1, \ldots, \psi_n)[v] = \theta[w] \text{ и } \sigma(\psi_1, \ldots, \psi_n)[v] = \sigma[w],$$

па је $\varphi(\psi_1, \ldots, \psi_n)[v] = 1$ акко $\theta(\psi_1, \ldots, \psi_n)[v] = 1$ и $\sigma(\psi_1, \ldots, \psi_n)[v] = 1$ акко $\theta[w] = 1$ и $\sigma[w] = 1$ акко $\varphi[v] = 1$; одавде, $\varphi(\psi_1, \ldots, \psi_n)[v] = \varphi[w]$.

Преостали случајеви раде се на сличан начин као 2°.

Дефиниција 1.8. Нека је φ формула и v валуација.

1. Кажемо да v *задовољава* φ или да је v *модел за* φ, у ознаци $v \models \varphi$, ако је $\varphi[v] = 1$; у супротном, ако $\varphi[v] = 0$, кажемо да је v *контрамодел* за φ и пишемо $v \not\models \varphi$.
2. Формула φ је *задовољива* ако постоји валуација v таква да $v \models \varphi$; у супротном, φ је *контрадикција*.
3. Формула φ је *порецива* ако постоји валуација v таква да $v \not\models \varphi$; у супротном, φ је *таутологија*, што записујемо са $\models \varphi$.

Приметимо да је φ таутологија ако за сваку валуацију v важи $v \models \varphi$ (тј. φ нема контрамодел). Слично, φ је контрадикција ако за сваку валуацију v важи $v \not\models \varphi$ (тј. φ нема модел). Очигледно, по дефиницији, $\top$ је пример једне таутологије, а $\bot$ је пример једне контрадикције.

Пример 1.9. Важне таутологије:

1. $\models p \vee \neg p$ (*tertium non datur*)
2. $\models \neg(p \wedge \neg p)$ (закон неконтрадикције)
3. $\models \neg\neg p \leftrightarrow p$ (класични закон дупле негације)
4. $\models p \wedge p \leftrightarrow p$ (идемпотентност за $\wedge$)
5. $\models p \vee p \leftrightarrow p$ (идемпотентност за $\vee$)
6. $\models (p \to p) \leftrightarrow \top$, $\models (p \leftrightarrow p) \leftrightarrow \top$ **и** $\models (p \veebar p) \leftrightarrow \bot$
7. $\models p \wedge \top \leftrightarrow p$ **и** $\models p \wedge \bot \leftrightarrow \bot$
8. $\models p \vee \top \leftrightarrow \top$ **и** $\models p \vee \bot \leftrightarrow p$
9. $\models (p \leftrightarrow \top) \leftrightarrow p$ **и** $\models (p \leftrightarrow \bot) \leftrightarrow \neg p$
10. $\models (p \veebar \top) \leftrightarrow \neg p$ **и** $\models (p \veebar \bot) \leftrightarrow p$
11. $\models \bot \to p$ (*ex falso quodlibet*)
12. $\models (p \to \bot) \leftrightarrow \neg p$ (*reductio ad absurdum*)
13. $\models p \to \top$ **и** $\models (\top \to p) \leftrightarrow p$
14. $\models p \wedge q \leftrightarrow q \wedge p$ **и** $\models p \vee q \leftrightarrow q \vee p$ (комутативност за $\wedge$ и $\vee$)
15. $\models (p \leftrightarrow q) \leftrightarrow (q \leftrightarrow p)$ **и** $\models (p \veebar q) \leftrightarrow (q \veebar p)$
16. $\models p \wedge (p \vee q) \leftrightarrow p$ **и** $\models p \vee (p \wedge q) \leftrightarrow p$ (закони апсорбције)
17. $\models p \wedge (q \wedge r) \leftrightarrow (p \wedge q) \wedge r$ **и** $\models p \vee (q \vee r) \leftrightarrow (p \vee q) \vee r$
(асоцијативност за $\wedge$ и $\vee$)
18. $\models (p \leftrightarrow (q \leftrightarrow r)) \leftrightarrow ((p \leftrightarrow q) \leftrightarrow r)$ **и** $\models (p \veebar (q \veebar r)) \leftrightarrow ((p \veebar q) \veebar r)$
19. $\models p \wedge (q \vee r) \leftrightarrow (p \wedge q) \vee (p \wedge r)$ **и** $\models p \vee (q \wedge r) \leftrightarrow (p \vee q) \wedge (p \vee r)$
(дистрибутивни закони)
20. $\models \neg(p \wedge q) \leftrightarrow \neg p \vee \neg q$ **и** $\models \neg(p \vee q) \leftrightarrow \neg p \wedge \neg q$
(Де Морганови закони)
21. $\models \neg(p \leftrightarrow q) \leftrightarrow (\neg p \leftrightarrow q)$ **и** $\models \neg(p \veebar q) \leftrightarrow (\neg p \veebar q)$
22. $\models (p \to q) \leftrightarrow (\neg q \to \neg p)$ (закон контрапозиције)
23. $\models (p \veebar q) \leftrightarrow \neg(p \leftrightarrow q)$
24. $\models (p \leftrightarrow q) \leftrightarrow (p \to q) \wedge (q \to p)$
25. $\models (p \to q) \leftrightarrow \neg p \vee q$
26. $\models p \wedge (p \to q) \to q$ (*modus ponens*)
27. $\models (p \to q) \wedge \neg q \to \neg p$ (*modus tollens*)
28. $\models (p \vee q) \wedge \neg p \to q$ (дисјунктивни силогизам)
29. $\models (p \to q) \wedge (q \to r) \to (p \to r)$ (хипотетички силогизам)

Користећи лему о замени (лема 1.7) лако добијамо следеће тврђење:

Тврђење 1.10. Нека су $\varphi = \varphi(p_1, \ldots, p_n), \psi_1, \ldots, \psi_n$ формуле. Ако $\models \varphi$, онда и $\models \varphi(\psi_1, \ldots, \psi_n)$. (И слично, ако је φ контрадикција, онда је и $\varphi(\psi_1, \ldots, \psi_n)$ контрадикција.)

Доказ. Нека је v произвољна валуација, и нека је w валуација слова $p_1, \ldots, p_n$ дата са $w(p_i) = \psi_i[v]$. Према леми 1.7 је $\varphi(\psi_1, \ldots, \psi_n)[v] = \varphi[w] = 1$, где друга једнакост важи јер $\models \varphi$. Како је v била произвољна, закључујемо $\models \varphi(\psi_1, \ldots, \psi_n)$.

Дефиниција 1.11. За формулу φ са $\mathrm{Mod}(\varphi)$ обележавамо скуп свих валуација скупа $\mathcal{P}_\varphi$ које су модели за φ, тј. које задовољавају φ.

3. Логичка еквиваленција, нормалне форме

Дефиниција 1.12. Формуле φ и ψ су *логички еквивалентне*, $\varphi \equiv \psi$ (или $\varphi \Leftrightarrow \psi$), ако за све валуације v важи $\varphi[v] = \psi[v]$.

Према дефиницијама логичке еквивалентности и интерпретације везника $\leftrightarrow$, видимо да је $\varphi \equiv \psi$ ако и само ако $\models \varphi \leftrightarrow \psi$. С тим у вези, таутологије наведене у примеру 1.9 код којих је главни везник еквиваленција дају одговарајуће логички еквивалентне формуле; примера ради, $p \wedge p \equiv p$, $p \vee \top \equiv \top$, $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$, итд.

Релација $\equiv$ на скупу $\mathcal{F}$ је једна релација еквиваленције. Специјално, класа еквиваленције $[\top]_\equiv$ је скуп свих таутологија, док је класа $[\bot]_\equiv$ скуп свих контрадикција.

Задатак 3.4. За формулу φ кажемо да је *позитивна* ако је изграђена коришћењем исказних слова и везника $\wedge, \vee$ и $\rightarrow$. Нека је v_1 валуација дата са $v_1(p) = 1$ за све $p \in \mathcal{P}$. Доказати:

1° формула φ је еквивалентна некој позитивној формули ако и само ако $\varphi[v_1] = 1$;

2° за сваку формулу φ постоји позитивна формула ψ таква да је $\varphi \equiv \psi$ или $\varphi \equiv \neg\psi$.

Задатак 3.5. Израчунати $|\mathcal{F}/_{\equiv}|$ ако је $\mathcal{P}$: 1° коначан; 2° пребројив.

Теорема 1.13. (Теорема о еквивалентним заменама)
Нека су $\varphi = \varphi(p_1, \ldots, p_n)$, $\varphi' = \varphi'(p_1, \ldots, p_n)$, $\psi_1, \ldots, \psi_n, \theta_1, \ldots, \theta_n$ формуле такве да $\varphi \equiv \varphi'$ и $\psi_i \equiv \theta_i$ за све $i = 1, \ldots, n$. Тада је:

$$\varphi(\psi_1, \ldots, \psi_n) \equiv \varphi'(\theta_1, \ldots, \theta_n).$$

Доказ. Како $\models \varphi \leftrightarrow \varphi'$, према тврђењу 1.10 имамо $\models \varphi(\theta_1, \ldots, \theta_n) \leftrightarrow \varphi'(\theta_1, \ldots, \theta_n)$, тј. $\varphi(\theta_1, \ldots, \theta_n) \equiv \varphi'(\theta_1, \ldots, \theta_n)$, па је довољно доказати $\varphi(\psi_1, \ldots, \psi_n) \equiv \varphi(\theta_1, \ldots, \theta_n)$. Доказ ове еквивалентности изводимо индукцијом по $\mathrm{sl}(\varphi)$. Ако је $\varphi = \bot$ или $\varphi = \top$, резултат следи директно. Ако је φ слово, како је $\varphi = \varphi(p_1, \ldots, p_n)$, $\varphi = p_i$ за неко i, па по претпоставци имамо $\varphi(\psi_1, \ldots, \psi_n) = \psi_i \equiv \theta_i = \varphi(\theta_1, \ldots, \theta_n)$. Претпоставимо сада да је $\mathrm{sl}(\varphi) > 0$. Ако је $\varphi = \neg\sigma$, према индукцијској хипотези је $\sigma(\psi_1, \ldots, \psi_n) \equiv \sigma(\theta_1, \ldots, \theta_n)$, одакле лако следи да је и $\neg\sigma(\psi_1, \ldots, \psi_n) \equiv \neg\sigma(\theta_1, \ldots, \theta_n)$, тј. $\varphi(\psi_1, \ldots, \psi_n) \equiv \varphi(\theta_1, \ldots, \theta_n)$. Слично, ако је $\varphi = \sigma * \tau$ за неки бинарни везник $*$, по индукцијској хипотези је $\sigma(\psi_1, \ldots, \psi_n) \equiv \sigma(\theta_1, \ldots, \theta_n)$ и $\tau(\psi_1, \ldots, \psi_n) \equiv \tau(\theta_1, \ldots, \theta_n)$, па лако видимо да је и $\sigma(\psi_1, \ldots, \psi_n) * \tau(\psi_1, \ldots, \psi_n) \equiv \sigma(\theta_1, \ldots, \theta_n) * \tau(\theta_1, \ldots, \theta_n)$, тј. $\varphi(\psi_1, \ldots, \psi_n) \equiv \varphi(\theta_1, \ldots, \theta_n)$.

Пример 1.14. PRIMER PRIMENE

Дефиниција 1.15.
1. *Литерал* је слово (*позитиван литерал*) или негација слова (*негативан литерал*).
2. *Клауза* је дисјункција литерала.
3. *Конјунктивна клауза* је конјункција литерала.
4. Формула је у *конјунктивној нормалној форми (КНФ)* ако је конјункција клауза.
5. Формула је у *дисјунктивној нормалној форми (ДНФ)* ако је дисјункција конјунктивних клауза.

Теорема 1.16. Свака формула је еквивалентна некој у ДНФ и некој у КНФ.

Доказ. Произвољну формулу можемо да сведемо на еквивалентну формулу користећи теорему о еквивалентним заменама (теорема 1.13) на следећи начин:
1. корак: Елиминишемо појављивања редом везника $\veebar$, $\leftrightarrow$ и $\rightarrow$ користећи законе $\varphi \veebar \psi \equiv \neg\varphi \leftrightarrow \psi$, $\varphi \leftrightarrow \psi \equiv (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$ и $\varphi \rightarrow \psi \equiv \neg\varphi \vee \psi$.
После првог корака у запису формуле учествују слова, константе и везници $\neg$, $\wedge$ и $\vee$.
2. корак: Елиминишем појављивања везника $\neg$ „испред заграда" користећи де Морганове законе $\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi$ и $\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi$, елиминишемо $\neg\neg$ користећи закон $\neg\neg\varphi \equiv \varphi$, и елиминишемо појављивање $\neg$ испред константи користећи законе $\neg\top \equiv \bot$ и $\neg\bot \equiv \top$.
После другог корака везник $\neg$ појављује се само испред слова.
3. корак: Сваку потформулу облика $\varphi \wedge (\psi \vee \theta)$ еквивалентно заменимо са $(\varphi \wedge \psi) \vee (\varphi \wedge \theta)$ (и $(\varphi \vee \psi) \wedge \theta$ са $(\varphi \wedge \theta) \vee (\psi \wedge \theta)$).
После трећег корака добијена формула је дисјункција потформула које су конјункције литерала и константи.
4. корак: Користећи законе $\top \wedge \varphi \equiv \varphi \wedge \top \equiv \varphi$ и $\bot \wedge \varphi \equiv \varphi \wedge \bot \equiv \bot$ сводимо формулу на дисјункцију потформула од којих је свака или конјункција литерала или $\top$ или $\bot$.
5. корак: Користећи законе $\top \vee \varphi \equiv \varphi \vee \top \equiv \top$ и $\bot \vee \varphi \equiv \varphi \vee \bot \equiv \varphi$ сводимо формулу или на формулу у ДНФ или на $\top$ или на $\bot$.
Описаним поступком добијамо еквивалентну формулу у ДНФ под условом да полазна формула није таутологија ни контрадикција. Ако је полазна формула таутологија, она је еквивалентна са $p \vee \neg p$, што је формула у ДНФ, а ако је полазна формула контрадикција, она је еквивалентна са $p \wedge \neg p$, што је такође формула у ДНФ. Према томе, свака формула је еквивалентна некој у ДНФ.
Поступак свођења на КНФ је сличан, једино је потребно да у корацима 3 – 5 применимо дуалне законе. (Алтернативно, можемо негацију

полазне формуле да сведемо на ДНФ, који негирањем, применом Де Морганових закона и брисањем дуплих негација постаје КНФ полазне формуле.)

Пример 1.17. PRIMER

Дефиниција 1.18. 1. За слово p и валуацију v, p^v означава литерал:

$$p^v \overset{def}{=} \begin{cases} p & \text{ако } v(p) = 1 \\ \neg p & \text{ако } v(p) = 0 \end{cases}.$$

2. За литерал ℓ, $\overline{\ell}$ означава супротан литерал:

$$\overline{\ell} \overset{def}{=} \begin{cases} p & \text{ако } \ell = \neg p \\ \neg p & \text{ако } \ell = p \end{cases}.$$

Приметимо да је директно по дефиницији $p^v[v] = 1$, и општије, $p^v[w] = 1$ ако и само ако $v(p) = w(p)$. Такође, за сваки литерал ℓ, ℓ и $\overline{\ell}$ имају супротну тачност при свакој валуацији.

Теорема 1.19. (Теорема о савршеним нормалним формама)
Нека је φ формула.

1. $\varphi \equiv \bigvee_{\substack{v:\mathcal{P}_\varphi \to \{0,1\} \\ \varphi[v]=1}} \bigwedge_{p \in \mathcal{P}_\varphi} p^v$;

2. $\varphi \equiv \bigwedge_{\substack{v:\mathcal{P}_\varphi \to \{0,1\} \\ \varphi[v]=0}} \bigvee_{p \in \mathcal{P}_\varphi} \overline{p^v}$.

Доказ. Доказаћемо део 1. Означимо са θ формулу на десној страни еквиваленције. Претпоставимо најпре да је w валуација слова формуле φ таква да $w \models \varphi$ и докажимо $w \models \theta$. Како $w \models \varphi$, формула $\bigwedge_{p \in \mathcal{P}_\varphi} p^w$ је један од дисјунката у θ. Како смо већ рекли, $p^w[w] = 1$ за сва слова p, па је и $\left(\bigwedge_{p \in \mathcal{P}_\varphi} p^w\right)[w] = 1$, одакле изводимо жељени закључак: $w \models \theta$.
Претпоставимо сада $w \models \theta$ и докажимо $w \models \varphi$. Из $w \models \theta$ имамо да постоји валуација v слова $\mathcal{P}_\varphi$ за коју је $\varphi[v] = 1$, и за коју је

$\left(\bigwedge_{p\in\mathcal{P}_\varphi} p^v\right)[w] = 1$. Одавде је за све $p \in \mathcal{P}_\varphi$, $p^v[w] = 1$, одакле закључујемо $v(p) = w(p)$. Дакле, валуације v и w поклапају се на словима формуле φ, па је $\varphi[w] = \varphi[v] = 1$ према теореми 1.5. Завршили смо доказ.

Формула у делу 1. претходне теореме је *савршена дисјунктивна нормална форма (СДНФ)* формуле φ, а формула у делу 2. је *савршена конјунктивна нормална форма (СКНФ)* формуле φ. Приметимо да ако је φ контрадикција, онда формула у делу 1. није дефинисана јер не постоји валуација v слова формуле φ за коју је $\varphi[v] = 1$. У том случају ћемо рећи да је $\bot$ СДНФ формуле φ. Слично, за таутологије φ не можемо да формирамо формулу у делу 2. и у том случају кажемо да је $\top$ СКНФ формуле φ.

Пример 1.20. Користећи теорему 1.19 налазимо СДНФ и СКНФ формуле φ дате следећом таблицом:

p	q	r	φ	$p^v \wedge q^v \wedge r^v$	$\overline{p^v} \vee \overline{q^v} \vee \overline{r^v}$
0	0	0	0		$p \vee q \vee r$
0	0	1	1	$\neg p \wedge \neg q \wedge r$	
0	1	0	0		$p \vee \neg q \vee r$
0	1	1	1	$\neg p \wedge q \wedge r$	
1	0	0	1	$p \wedge \neg q \wedge \neg r$	
1	0	1	0		$\neg p \vee q \vee \neg r$
1	1	0	0		$\neg p \vee \neg q \vee r$
1	1	1	1	$p \wedge q \wedge r$	

СДНФ формуле φ је:

$$(\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (p \wedge \neg q \wedge \neg r) \vee (p \wedge q \wedge r),$$

СКНФ формуле φ је:

$$(p \vee q \vee r) \wedge (p \vee \neg q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r).$$

Задатак 3.6. Доказати део 2. у теореми 1.19.

Задатак 3.7. Нека су φ и ψ формуле такве да φ није контрадикција, ψ није таутологија и $\models \varphi \to \psi$. Доказати:
1° $\mathcal{P}_\varphi \cap \mathcal{P}_\psi \neq \emptyset$;
2° постоји формула θ таква да $\models \varphi \to \theta$, $\models \theta \to \psi$ и $\mathcal{P}_\theta \subseteq \mathcal{P}_\varphi \cap \mathcal{P}_\psi$.

Задатак 3.8. Нека су φ и ψ формуле такве да $\models \varphi \to \psi$ и $\not\models \psi \to \varphi$. Доказати да постоји формула θ таква да $\models \varphi \to \theta$, $\models \theta \to \psi$, $\not\models \psi \to \theta$ и $\not\models \theta \to \varphi$.

4. Проблем задовољивости

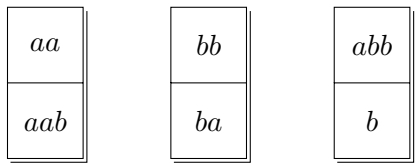
4.1. Проблеми одлучивања. Нека су $P \subseteq \Sigma$ пребројиви скупови. *Проблем одлучивања* (Σ, P), или само P ако је значење Σ јасно из контекста, је следећи проблем.

Проблем. (Проблем одлучивања (Σ, P))
За дато $x \in \Sigma$, одредити да ли $x \in P$.

Дефиниција 1.21. Проблем (Σ, P) је *одлучив* ако постоји алгоритам који се за произвољан улаз облика $x \in \Sigma$ зауставља и коректно даје одговор *да* или *не* на питање: „Да ли $x \in P$?”.

Претходна дефиниција је неформална, али је довољно добра за потребе овог текста.

Пример 1.22. Нека је S_n n-точлани скуп симбола. Реч над S_n је било који коначни низ симбола из S_n; нека је R_n скуп свих речи над S_n. *Домина* над S_n је било који уређен пар речи из R_n; означимо са $D_{\alpha,\beta}$ домину (α, β). Ако је $S_2 = \{a, b\}$, домине $D_{aa,aab}$, $D_{bb,ba}$ и $D_{abb,b}$ цртамо на следећи начин:



Проблем. (PCP_n – Post Correspondence Problem) За дато k и домине D_{α_i,β_i} над S_n, $i \leqslant k$, одредити да ли постоје m и i_j, $j \leqslant m$, такви да $i_j \leqslant k$ за све $j \leqslant m$, и речи $\alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_m}$ и $\beta_{i_1}\beta_{i_2}\ldots\beta_{i_m}$ су једнаке. (Другим речима, одредити да ли је могуће поређати (са евентуалним понављањем) дате домине тако да су речи добијене у горњем и доњем реду једнаке.)

Нагласимо да је n у датом проблему фиксирано. Горе представљена инстанца проблема PCP_2 има следеће решење:

aa	bb	aa	abb
aab	ba	aab	b

Дакле, ређањем домина $D_{aa,aab}, D_{bb,ba}, D_{aa,aab}, D_{abb,b}$ добијамо и у горњем и у доњем реду реч $aabbaaabb$. Са друге стране, инстанца дата са $D_{aa,aab}, D_{b,abb}$ нема решење јер је горња реч у обе домине краћа од доње, па било којим ређањем оваквих домина увек добијамо у горњем реду краћу реч.

Докажимо да је проблем PCP_1 одлучив. Нека је $S_1 = \{a\}$. Реч над S_1 је облика a^s – s понављања слова a, па домину D_{a^s,a^t} над S_1 можемо означити са $D_{s,t}$. Тврдимо да инстанца $D_{s_1,t_1}, \ldots, D_{s_k,t_k}$ има решење ако и само ако постоји $i \leqslant k$ такво да $s_i = t_i$, или постоје $i, j \leqslant k$ такви да $s_i < t_i$ и $s_j > t_j$. Ако постоји $i \leqslant k$ такво да $s_i = t_i$, решење инстанце је само једна домина D_i. Ако постоје $i, j \leqslant k$ тако да $s_i < t_i$ и $s_j > t_j$, онда је једно решење инстанце:

$$\underbrace{D_{s_i,t_i}\ldots D_{s_i,t_i}}_{s_j - t_j \text{ пута}}\underbrace{D_{s_j,t_j}\ldots D_{s_j,t_j}}_{t_i - s_i \text{ пута}}.$$

Заиста, горња реч је тада $(s_j - t_j)s_i + (t_i - s_i)s_j = t_i s_j - s_i t_j$ понављања слова a, а доња реч је $(s_j - t_j)t_i + (t_i - s_i)t_j = t_i s_j - s_i t_j$ понављања слова a; ове две речи су једнаке. Са друге стране, ако за све $i \leqslant k$ важи $s_i < t_i$, или за све $i \leqslant k$ важи $s_i > t_i$, решење не може да постоји јер било којим слагањем домина горња реч ће бити у првом случају краћа, а у другом дужа од доње речи. Према томе, алгоритам који одлучује PCP_1 је једноставан: за домине са улаза треба само да

провери да ли је горња реч у свим доминама краћа (односно, дужа) од доње.

Насупрот претходном резулату имамо следећи.

Теорема. (Пост) За $n \geqslant 2$, PCP_n није одлучив.

На крају приметимо да су одлучиви проблеми затворени за комплемент. Прецизније, ако је (Σ, P) одлучив проблем, онда је очиледно и (Σ, P^c) одлучив проблем. Наиме, алгоритам који одлучује P истовремено одлучује и P^c, довољно је само да обрнемо *да* и *не* на излазу.

4.2. SAT. Проблем задовољивости или SAT[2] је следећи проблем одлучивања.

Проблем. (Проблем задовољивости – SAT)
Одредити да ли је дата формула задовољива.

У складу са овде датом дефиницијом проблема одлучивања, SAT је проблем $(\mathcal{F}, \mathrm{SAT})$, где је $\mathrm{SAT} \subseteq \mathcal{F}$ подскуп задовољивих формула.

Теорема 1.23. Проблем SAT је одлучив.

Доказ. Један очигледан алгоритам који одлучује SAT је следећи: за формулу на улазу φ, израчунавамо вредности формуле φ у свим валијацијама скупа $\mathcal{P}_\varphi$. Ако је $\varphi[v] = 1$ за неку v враћамо *да*; у супротном, враћамо *не*. Како је скуп $\mathcal{P}_\varphi$ коначан, јасно је да се овај алгоритам зауставља за сваки улаз φ.

Према претходној теореми имамо да је и проблем: „За дату формулу φ одредити да ли је φ контрадикција.” одлучив као комплемент проблема SAT. Слично као у претходном доказу можемо да видимо и да је проблем порецивости: „За дату формулу φ одредити да ли је φ порецива.”, као и његов комплемент: „За дату формулу φ одредити да ли је φ таутологија.” одлучив.

[2]Скраћено од *satisfiability*.

4.3. Временска сложеност алгоритма. Неформално речено, временска сложеност алгоритма је величина која описује које је време потребно за његово извршавање, или, у колико корака ће се завршити извршавање алгоритма. Временску сложеност рачунамо у односу на величину улаза. Како израчунавања и за исту величину улаза могу да буду различитог трајања, узимамо најгоре могуће време.

...

4.4. Свођење на SAT.

Дефиниција 1.24. Нека су (Σ_1, P_1) и (Σ_2, P_2) два проблема. Проблем (Σ_1, P_1) *се своди на* (Σ_2, P_2), пишемо $(\Sigma_1, P_1) \leqslant (\Sigma_2, P_2)$, ако:

- постоји функција $f : \Sigma_1 \to \Sigma_2$ таква да за све $x \in \Sigma_1$ важи $x \in P_1 \Leftrightarrow f(x) \in P_2$;
- постоји алгоритам који се за произвољан улаз облика $x \in \Sigma_1$ зауставља и враћа вредност $f(x)$.

Проблем (Σ_1, P_1) *се своди на* (Σ_2, P_2) *у полиномијалном времену*, $(\Sigma_1, P_1) \leqslant_P (\Sigma_2, P_2)$, ако додатно у претходној дефиницији тражимо и да постоји полином p такав да се алгоритам за улаз $x \in \Sigma_1$ зауставља у највише $p(|x|)$ корака, где $|x|$ означава дужину улаза x.

Ако $(\Sigma_1, P_1) \leqslant (\Sigma_2, P_2)$ и (Σ_2, P_2) је одлучив, тада је и (Σ_1, P_1) одлучив простим надовезивањем алгоритама за свођење (Σ_1, P_1) на (Σ_2, P_2) и алгоритма за одлучивање (Σ_2, P_2). Штавише, ако је дато свођење у полиномијалном времену и (Σ_2, P_2) је у **P**, на исти начин видимо да је онда и (Σ_1, P_1) у **P**. Овде треба додатно нагласити следећу финесу: ако функција f и полином p сведоче $(\Sigma_1, P_1) \leqslant_P (\Sigma_2, P_2)$, онда је $|f(x)| \leqslant p(|x|)$ за све $x \in \Sigma_1$.

Дајемо неколико примера ефективног свођења проблема одлучивања на SAT.

Пример 1.25. Означимо са LP (логичка последица) следећи проблем.

Проблем. За дате формуле φ и ψ одредити да ли је $\varphi \models \psi$.

Како је $\varphi \not\models \psi$ акко $\not\models \varphi \to \psi$, акко $\neg(\varphi \to \psi) \equiv \varphi \wedge \neg\psi$ је задовољива, функција $(\varphi, \psi) \mapsto \varphi \wedge \neg\psi$ описује једно ефективно свођење горњег

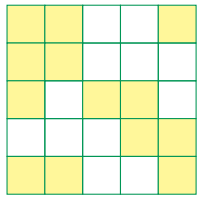
проблема на SAT. Формално, ако је LP скуп парова формула (φ, ψ) таквих да $\varphi \models \psi$, претходна функција показује $\neg\text{LP} \leqslant \text{SAT}$.

Задатак 3.9. Нека је LE (логичка еквивалентност) следећи проблем.

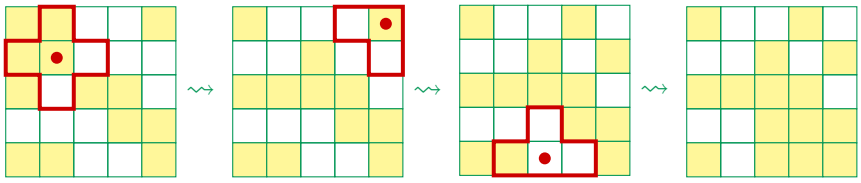
Проблем. За дате формуле φ и ψ одредити да ли је $\varphi \equiv \psi$.

Објаснити како се одлучивање LE може свести на SAT.

Пример 1.26. Lights Out је следећа игра. Дата је табла величине $m \times n$ код које је свако поље обојено бело или жуто; ако је поље жуто, замишљамо да је светло у њему упаљено, а ако је поље бело, светло у том пољу је угашено. Пример једне такве табле величине 5×5 дата је на следећој слици.



У сваком кораку игре можемо да упалимо или угасимо сијалицу у било ком пољу. Међутим, такав потез аутоматски мења и стање у свим суседним пољима (суседна поља су поља која су непосредно изнад, испод, лево и десно, ако постоје). Играње на пољима $(2, 2)$, $(1, 5)$ и $(5, 3)$ приказујемо на следећој слици.



Циљ саме игре је да угасимо сва светла на табли. Напоменимо да то није могуће за све почетне конфигурације. Овде нас занима следећи проблем.

Проблем. За дату таблу на улазу одредити да ли је могуће да се угасе сва светла.

Објаснићемо како се претходни проблем своди на SAT. Најпре уочимо неколико чињеница. Прво, сваки потез је инволуција. Друго, два потеза комутирују. Према томе, ако постоји решење, онда постоји решење у коме смо на сваком пољу играли највише једном; таква решења зовемо минимална. Уведимо исказна слова $p_{i,j}$, $i \leqslant m$ и $j \leqslant n$; замишљамо да слово $p_{i,j}$ има значење „Играмо на пољу (i, j).". Тада свака валуација ових слова одговара једној могућој минималној игри (код које не понављамо потезе). Имајући у виду ову кореспонденцију, описаћемо које формуле v треба да задовољи да би била решење табле.

Посматрајмо поље (i, j) које није у углу нити на ивици. Стање у пољу (i, j) може да промени играње на пољима $(i-1, j), (i, j-1), (i, j), (i, j+1)$ и $(i+1, j)$. Ако је поље (i, j) на почетку жуто, морамо да одиграмо на непарно много од датих пет поља како би на крају светло било угашено. Ако је, са друге стране, на почетку светло већ угашено у пољу (i, j) морамо да одиграмо на парно много од датих пет поља како би светло и на крају било угашено. Формула чија тачност зависи од тога да ли је број тачних слова у њој непаран или паран је њихова ексклузивна дисјункције. Дакле, нека је $\theta_{i,j}$ следећа формула:

$$\theta_{i,j} \overset{def}{=} p_{i-1,j} \veebar p_{i,j-1} \veebar p_{i,j} \veebar p_{i,j+1} \veebar p_{i+1,j}.$$

Према претходној анализи, ако је v решење полазне инстанце, мора бити $\theta_{i,j}[v] = 1$ ако је поље (i, j) на почетку жуто, а $(\neg\theta_{i,j})[v] = 1$ у супротном. Иста анализа важи и за поља у угловима и на страницама, осим што у одговарајућој формули $\theta_{i,j}$ имамо мање чланова.

Сада је лако да полазној табели доделимо формулу. За полазну табелу формирамо конјункцију φ на следећи начин. За свако поље (i, j), ако је светло на њему упаљено, ставимо да је $\theta_{i,j}$ један од конјунката формуле φ; у супротном, ставимо да је $\neg\theta_{i,j}$ један од конјунката формуле φ. Горња анализа нам каже да ако је v решење табеле, онда је $\varphi[v] = 1$, али лако видимо да важи и обрнуто. Према томе, улазна табела има решење ако и само ако је φ задовољива формула. Тиме смо очигледно свели полазни проблем на SAT.

Пример 1.27. *Усмерен граф* је структура $G = (V, E)$, где је V непразан скуп чије елементе зовемо *чворови графа*, а E је произвољна бинарна релација на скупу E чије елементе зовемо *стрелице графа*. За $a, b \in V$, уместо $E(a, b)$ обично пишемо $a \to b$. *Циклус* је низ елемената $a_0, a_1, \ldots, a_n$, $n \geqslant 0$, такав да $a_0 \to a_1 \to \cdots \to a_n \to a_0$.

Проблем. За дати граф $G = (V, E)$ одредити да ли у G постоји циклус.

Показаћемо како одлучивање претходног проблема можемо да сведемо на LP, па према примеру 1.25 и на SAT. *Усмерени пут* у графу је низ чворова $a_0, a_1, \cdots, a_n$, $n \geqslant 1$, такав да $a_0 \to a_1 \to \cdots \to a_n$. Уведимо слова $p_{a,b}$ за све $a, b \in V$. Интуитивно, слово $p_{a,b}$ има значење „од a до b постоји усмерени пут". Уочимо следеће формуле:

$$\varphi = \bigwedge_{\substack{a,b \in V \\ a \to b}} p_{a,b} \wedge \bigwedge_{a,b,c \in V} (p_{a,b} \wedge p_{b,c} \to p_{a,c}) \quad \text{и} \quad \psi = \bigvee_{a \in V} p_{a,a}.$$

Формуле прве конјункције у φ кажу да за сваку стрелицу $a \to b$ постоји усмерен пут од a до b, док формуле друге конјункције кажу да ако постоји усмерен пут од a до b и од b до c, онда постоји и усмерен пут од a до c. Дисјункти у формули ψ кажу да постоји усмерен пут од a до a (дакле, циклус кроз a).

Тврдимо да G има циклус ако и само ако $\varphi \models \psi$. Претпоставимо да G има циклус $a_0 \to a_1 \to \cdots \to a_n \to a_0$, $n \geqslant 0$, и претпоставимо $v \models \varphi$. Тада су слова $p_{a_0,a_1}, p_{a_1,a_2}, \ldots, p_{a_n,a_0}$ сва тачна у валацији v, (прва конјункција у φ), па директно видимо, користећи формуле из друге конјункције у φ, да и p_{a_0,a_0} мора бити тачно у v. Одатле, $v \models \psi$, и закључујемо $\varphi \models \psi$.

Обратно, претпоставимо G нема циклусе. Дефинишимо валуацију v на следећи начин. Ако постоји усмерен пут од a до b ставимо $v(p_{a,b}) = 1$; у супротном, ставимо $v(p_{a,b}) = 0$. Како G нема циклусе, за свако a је $v(p_{a,a}) = 0$, па $v \models \psi$. Да бисмо закључили $\varphi \not\models \psi$, довољно је да докажемо $v \models \varphi$. За све $a, b \in V$, ако $a \to b$, постоји усмерени пут од a до b, па је по дефиницији $v(p_{a,b}) = 1$. Дакле, v задовољава прву велику конјункцију у φ. Нека су сада $a, b, c \in V$ произвољни. Тврдимо $v \models p_{a,b} \wedge p_{b,c} \to p_{a,c}$. Ако $v \models p_{a,b} \wedge p_{b,c}$, по дефиницији

v постоје усмерени путеви од a до b и од b до c, па надовезивањем ова два пута постоји и усмерен пут од a до c, тј. по дефиницији v је $v(p_{a,c}) = 1$; закључујемо $v \models p_{a,b} \wedge p_{b,c} \to p_{a,c}$. Дакле, v задовољава и другу велику конјункцију у φ, па $v \models \varphi$.

Пример 1.28. Ана и Бане играју следећу игру. Најпре Ана бира број n и $3 \times n$ матрицу A чији су сви улази облика x_i или $-x_i$ за $i \leqslant n$. У следећем кораку Бане свакој од променљивих x_i, $i \leqslant n$, додељује вредност 1 или -1 и рачуна матрицу одговарајућу матрицу A. На крају Ана сортира сваку колону у неопадајућем поретку. Бане побеђује ако су све вредности у средњој врсти сортиране матрице једнаке 1. Пример једне игре у којој Бане побеђује је следећи. Ана бира $n = 4$ и матрицу A:

$$A = \begin{bmatrix} -x_4 & -x_1 & x_2 & x_3 \\ x_1 & x_3 & x_3 & -x_4 \\ x_3 & -x_2 & x_4 & x_3 \end{bmatrix}.$$

Сада Бане бира вредности за x_i: $f = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ -1 & 1 & 1 & -1 \end{pmatrix}$, и рачуна дату матрицу за изабране вредности:

$$A_f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

Ана сортира све колоне и добија матрицу:

$$A_f^s = \begin{bmatrix} -1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Како су све вредности у другој врсти претходне матрице једнаке 1, Бане добија игру.

Један пример у коме Бане не може да добије је следећи. Ана бира $n = 2$ и матрицу A:

$$A = \begin{bmatrix} x_1 & -x_2 \\ -x_1 & -x_2 \\ x_2 & x_1 \end{bmatrix}.$$

Коју год вредност за x_1 Бане изабрао, у првој колони појавиће се и 1 и -1. Зато Бане мора да бира $x_2 = 1$ јер ће у супротном, након сортирања, на месту $(2, 1)$ бити -1. Али сада у другој колони имамо -1 два пута, па ће сигурно након сортирања на месту $(2, 2)$ бити -1.

Проблем. За дато n и матрицу A одредити да ли Бане може да победи.

Показаћемо како се претходни проблем своди на SAT. Уведимо слова p_i, $i \leqslant n$, и, неформално говорећи, уочимо смену $x_i = (-1)^{p_i}$. Дакле, уместо да Бане бира вредности ± 1 за x_i, може да бира вредности $0/1$ за p_i, тј. да бира једну валуацију ових слова. У складу са овом сменом важи $(-1)^e x_i = (-1)^{e+p_i} = (-1)^{\neg p_i^e}$, где последња једнакост има јасно значење.

Јасно је да Бане побеђује ако се у свакој колони, када је срачуната, -1 појављује највише једном. Уочимо произвољну, нпр. m-ту колону; она је следећег облика $[(-1)^a x_i \;\; (-1)^b x_j \;\; (-1)^c x_k]^T$. Међу њеним координатама -1 се појављује највише једном акко (имајући у виду горњу једнакост) у скупу $\{\neg p_i^a, \neg p_j^b, \neg p_k^c\}$ се 1 појављује највише једном, акко у скупу $\{p_i^a, p_j^b, p_k^c\}$ се 0 појављује највише једном. Међутим, ово је еквивалентно са чињеницом да изабрана валуација задовољава формулу:

$$\theta_m = (p_i^a \vee p_j^b) \wedge (p_i^a \vee p_k^c) \wedge (p_j^b \vee p_k^c).$$

Дакле, описали смо алгоритам који за дату m-ту колону формира формулу θ_m, па онда за дати унос формира и формулу $\varphi = \bigwedge_{m=1}^{n} \theta_m$. Према претходној анализи, валуација v задовољава φ акко даје решење проблема на улазу, па проблем на улазу има решење акко је φ задовољива. На овај начин сводимо полазни проблем на SAT.

Задатак 3.10. Свести на SAT проблем одлучивања да ли дата судоку таблица има решење.

Задатак 3.11. Свести на SAT проблем одлучивања да ли се чворови неусмереног графа могу обојити у k боја тако да суседни чворови нису обојени истом бојом. Овде је $k \geqslant 2$ фиксиран број.

4.5. Сложеност ДНФ и КНФ. Видели смо да се свака формула може трансформисати у еквивалентну формулу која је у некој нормалној форми. Природно можемо поставити питање да ли се таква трансформација може урадити у полиномијалном времену у односу на сложеност улазне формуле. Нажалост, одговор на ово питање је негативан. Проблем је што и најмања сложеност неке формуле у нормалној форми која је еквивалента полазној може да буде експоненцијална у односу на сложеност полазне формуле. Дајемо такав пример.

Пример 1.29. Нека су p_i^1, p_i^2, $i \leqslant n$, међусобно различита слова. Посматрајмо формулу φ (која је у ДНФ):

$$\varphi = \bigvee_{i=1}^{n} (p_i^1 \wedge p_i^2).$$

Није тешко видети да је $\mathrm{sl}(\varphi) = 2n - 1$. Такође, није тешко видети да је једна формула у КНФ еквивалентна са φ формула ψ:

$$\psi = \bigwedge_{\eta:[n]\to\{1,2\}} \bigvee_{i=1}^{n} p_i^{\eta(i)}.$$

(Овде $[n]$ означава скуп $\{1, \ldots, n\}$.) Формулу ψ добијамо тако што у формули φ „измножимо” заграде „сваки-са-сваким” користећи дистрибутивни закон. Сложеност формуле ψ једнака је $\mathrm{sl}(\psi) = 2^n(n-1) + 2^n - 1 = n2^n - 1$, што није ограничено полиномом по n. Показаћемо да не можемо да нађемо „бољу” КНФ формуле φ, тј. КНФ мање сложености од ψ.

Претпоставимо да је θ формула у КНФ које је најмање могуће сложености и еквивалентна је са φ. Приметимо да свака од клауза у θ не садржи супротстављене литерале (слово и његову негацију), јер

бисмо у супротном могли да заборавимо такву клаузу и да добијемо КНФ мање сложености.

Даље, приметимо да за свако $i \leqslant n$ и за сваку клаузу γ у θ, бар једно од слова p_i^1 и p_i^2 мора да има позитивно појављивање у γ. У супротном, нека је v валуација таква да $v \not\models \gamma$. Тада можемо коректно проширити валуацију v са $v(p_i^1) = v(p_i^2) = 1$ (ако је потребно, тј. ако се ова слова не појављују ни негативно у γ; ако се нпр. $\neg p_i^1$ појављује у γ, већ важи $v(p_i^1) = 1$, тако да не морамо да проширујемо v на слово p_i^1). Тада $v \models \varphi$, али $v \not\models \theta$, што је контрадикција.

Специјално, закључујемо да свака клауза γ у θ има бар n литерала. За сваку функцију $\eta : [n] \to \{1, 2\}$ уочимо валуацију v_η дату са $v_\eta(p_i^1) = 1$ ако $\eta(i) = 1$ и $v_\eta(p_i^2) = 1$ ако $\eta(i) = 2$; јасно је да $v_\eta \not\models \varphi$, па $v_\eta \not\models \theta$. То значи да постоји клауза γ_η у θ таква да $v_\eta \not\models \gamma_\eta$; специјално, одавде p_i^1 не учествује позитивно у γ_η ако $\eta(i) = 1$, а p_i^2 не учествује позитивно у γ_η ако $\eta(i) = 2$. Тврдимо, ако су ζ и η две различите функције, онда $\gamma_\zeta \neq \gamma_\eta$. Изаберимо i тако да је $\zeta(i) = 1$ и $\eta(i) = 2$ (или обрнуто, што анализирамо на сличан начин). Тада p_i^1 нема позитивно појављивање у γ_ζ, па према претходном пасусу p_i^2 има, али такође p_i^2 нема позитивно појављивање у γ_η; дакле, заиста $\gamma_\zeta \neq \gamma_\eta$.

Закључујемо да θ мора да има бар 2^n клауза, од којих свака има бар n литерала. Дакле, $\mathrm{sl}(\theta) \geqslant \mathrm{sl}(\psi)$, па према томе ψ заиста јесте формула у КНФ најмање могуће сложености која је еквивалентна са φ.

4.6. DNF-SAT. Неке варијанте проблема задовољивости јесу одлучиве у полиномијалном времену. Први такав пример је проблем DNF-SAT: Одредити да ли је дата формула у ДНФ задовољива. С обзиром да је формула у ДНФ задовољива ако и само ако је бар један њен дисјункт задовољив, а конјункција литерала је задовољива ако и само ако у њој не учествују *супротстављени* литерали (литерали облика p и $\neg p$ за исто слово p), довољно је само за сваки дисјункт улазне формуле проверити да не садрже супротстављене литерале. Ово се очигледно може урадити у полиномијалном времену у односу на сложеност улазне формуле.

4.7. XOR-SAT. *XOR-клауза* је ексклузивна дисјункција литерала. Имајући у виду законе комуникативности и асоцијативности ексклузивне дисјункције, као и закон:

$$\neg p \veebar q \;\equiv\; \neg(p \veebar q)$$

(негација се може „извући” испред ексклузивне дисјункције)

видимо да је свака XOR-клауза еквивалентна ексклузивној дисјункцији слова или негацији ексклузивне дисјункције слова. Кажемо да је формула у *XOR-KNF* ако је конјункција XOR-клауза. Проблем *XOR-SAT* је проблем одлучивања да ли је дата формула у XOR-KNF задовољива. Приметимо да смо у примеру 1.26, Lights Out проблем свели управо на XOR-SAT.

Нека је $\theta = p_1 \veebar p_2 \veebar \ldots \veebar p_n$. Ако скуп тачности $\{0, 1\}$ посматрамо као елементе поља $\mathbb{Z}_2$, ексклузивна дисјункција на њима одговара сабирању у $\mathbb{Z}_2$. При томе, за валацију v, имамо кореспонденцију:

$$\begin{aligned} v \models \theta \quad &\Leftrightarrow \quad v(p_1) + v(p_2) + \cdots + v(p_n) = 1 \\ v \not\models \theta \quad &\Leftrightarrow \quad v(p_1) + v(p_2) + \cdots + v(p_n) = 0. \end{aligned}$$

Према томе, инстанца XOR-KNF проблема своди се на систем линеарних једначина над пољем $\mathbb{Z}_2$, при чему је полазна инстанца задовољива ако и само ако придружени систем има решење. Приметимо да ово свођење можемо урадити у линеарном времену. Систем линеарних једначина решив је у полиномијалном (кубном) времену методом Гаусове елиминације. Према томе:

Теорема 1.30. Проблем XOR-SAT је у **P**.

4.8. HORN-SAT. *Хорнова клауза* је дисјункција литерала у којој учествује највише један позитиван литерал. Примери Хорнових клауза су p, $\neg p$, $\neg p \vee q$, $p \vee \neg q$, $\neg p \vee \neg q$, $\neg p \vee q \vee \neg r$, $\neg p \vee \neg q \vee r \vee \neg s$, итд. Формула је *Хорнова* ако је конјункција Хорнових клауза. Проблем *HORN-SAT* је проблем одлучивања да ли је дата Хорнова формула задовољива. У примеру 1.27 свели смо проблем постојања циклуса у усмереном графу на питање да ли $\varphi \models \psi$, тј. на питање да ли је $\varphi \wedge \neg\psi$ задовољива, где је последња

формула била:

$$\bigwedge_{\substack{a,b\in V\\ a\to b}} p_{a,b} \wedge \bigwedge_{a,b,c\in V} (p_{a,b} \wedge p_{b,c} \to p_{a,c}) \wedge \bigwedge_{a\in V} \neg p_{a,a}.$$

Имајући у виду да је $p \wedge q \to r \equiv \neg p \vee \neg q \vee r$, заправо видимо да смо проблем свели на инстанцу проблема HORN-SAT.

Свака Хорнова клауза може да се запише у *импликацијском облику*:

$$\begin{array}{rcl} \neg p_1 \vee \neg p_2 \vee \cdots \vee \neg p_n \vee q & \equiv & p_1 \wedge p_2 \wedge \cdots \wedge p_n \to q \\ \neg p_1 \vee \neg p_2 \vee \cdots \vee \neg p_n & \equiv & p_1 \wedge p_2 \wedge \cdots \wedge p_n \to \bot \\ q & \equiv & \top \to q \end{array}$$

где је $n \geqslant 1$. (Први случај одговара клаузама које имају и негативне и позитиван литерал, други одговара клаузама које имају само негативне литерале, и последњи клаузама које имају само позитиван литерал.) Доказаћемо да следећи алгоритам одлучује HORN-SAT у полиномијалном времену.

Алгоритам (HORN-SAT)
улаз: Хорнова формула φ
постави $v \equiv 0$ (v је идентички нетачна)
док $\varphi[v] = 0$:
  изаберемо клаузу $\theta = \pi \to \ell$ за коју $\theta[v] = 0$
  ако је ℓ слово:
    постави $v(\ell) = 1$
  иначе (ако је $l = \bot$):
    врати „φ није задовољива”
врати φ је „задовољива” (и $v \models \varphi$)

За две валуације v, v' слова формуле φ пишемо $v \leqslant v'$ ако за свако слово $p \in \mathcal{P}_\varphi$ важи $v(p) \leqslant v'(p)$ (где је, наравно, $0 \leqslant 1$); такође, $v < v'$ ако $v \leqslant v'$ и $v \neq v'$.

Фиксирајмо Хорнову формулу φ. Означимо са $v_0 \equiv 0$ идентички нетачну валуацију, а са v_i валуацију v коју алгоритам даје после i-тог проласка

кроз док-петљу (ако се док-петља не прекида враћањем „φ није задовољива" у i-том проласку, у ком случају v_i не дефинишемо).

Лема 1.31. Претпоставимо да је за неко $i \geqslant 1$, v_i дефинисана. Тада:
(i) $v_{i-1} < v_i$;
(ii) ако је φ задовољива и $w \models \varphi$, онда $v_i \leqslant w$.

Доказ. (i) Претпоставимо да је v_i дефинисана, тј. у i-том пролазу док-петља не враћа „φ није задовољива". То значи да смо изабрали клаузу $\theta = \pi \to q$ (где је π или конјункција слова или $\top$, а q је слово) такву да $\theta[v_{i-1}] = 0$; одатле је специјално $v_{i-1}(q) = 0$. Како је q слово, у следећем кораку смо предефинисали валуацију v_{i-1} (само) на слову q; тј. доделили смо $v_i(q) = 1$. Очигледно, $v_{i-1} < v_i$.
(ii) Доказ изводимо индукцијом по i. Јасно је да је $v_0 \leqslant w$. Претпоставимо да је $v_{i-1} \leqslant w$ и да је v_i дефинисана. Као у доказу (i), то значи да само уочили клаузу $\theta = \pi \to q$ такву да $\theta[v_{i-1}] = 0$ (па $\pi[v_{i-1}] = 1$ и $v_{i-1}(q) = 0$), и предефинисали смо валуацију на слову q: $v_i(q) = 1$. Како је $v_{i-1} \leqslant w$, и v_{i-1} и v_i се поклапају на свим словима осим q, да бисмо закључили да је $v_i \leqslant w$, довољно је да докажемо $w(q) = 1$. Претпоставимо супротно, $w(q) = 0$. Како је $\theta[w] = 1$ (јер $w \models \varphi$), то је и $\pi[w] = 0$, што специјално значи и да је $\pi = p_1 \wedge \cdots \wedge p_m$, одакле $w(p_j) = 0$ за неко $j \leqslant m$. Тада $v_{i-1} \leqslant w$ повлачи $v_{i-1}(p_j) = 0$, па је и $\pi[v_{i-1}] = 0$; контрадикција.

Докажимо сада коректност HORN-SAT алгоритма.

Теорема 1.32. HORN-SAT алгоритам се зауставља. При томе, φ је задовољива ако и само ако алгоритам враћа „φ је задовољива".

Доказ. Претпоставимо да φ има n слова. Према леми 1.31(i), низ $v_0, v_1, \ldots$ строго је растући низ валуација n слова, што значи да не може да има више од $n + 1$ члана. Дакле, најдаље валуација v_n је дефинисана, што значи да ћемо из док-петље изаћи у најдаље $(n + 1)$-ом пролазу.

Докажимо сада други део теореме. Смер ($\Leftarrow$) је очигледан: алгоритам враћа „φ је задовољива" ако претходно услов петље $\varphi[v] = 0$ није био испуњен, тј. ако је $\varphi[v] = 1$; дакле, φ је задовољива и v је једна валуација која је задовољава.
($\Rightarrow$) Претпоставимо да је φ задовољива, и нека $w \models \varphi$. Претпоставимо супротно, алгоритам се завршава тако што враћа „φ није задовољива" у i-том пролазу док-петље. То значи да смо у i-том пролазу изабрали клаузу $\theta = p_1 \wedge \cdots \wedge p_m \to \bot$ такву да $\theta[v_{i-1}] = 0$; специјално, $v_{i-1}(p_j) = 1$ за све $j \leqslant m$. Како је, према леми 1.31(ii), $v_{i-1} \leqslant w$, закључујемо $w(p_j) = 1$ за све $j \leqslant m$, па је и $\theta[w] = 0$; ово је контрадикција јер $w \models \varphi$.

С обзиром да је према доказу претходне теореме број пролаза кроз док-петљу у HORN-SAT алгоритму ограничен са бројем слова у улазној формули φ, као и да се рачунање $\varphi[v]$ може урадити у линеарном времену, јасно је да је алгоритам полиномијалне сложености.

Теорема 1.33. HORN-SAT је у **P**.

Коментар 1.34. Хорнова формула φ, ако је задовољива, има најмање решење (најмању валуацију која је задовољава) у односу на уведено уређење $\leqslant$. Наиме, ако су v, v' два решења, онда и њихов инфимум $v \wedge v'$ у односу на $\leqslant$, а за који лако видимо да је дефинисан са $(v \wedge v')(p) = v(p) \wedge v'(p)$ за све $p \in \mathcal{P}_\varphi$, такође задовољава φ. Ову чињеницу можемо да проверимо директним рачуном посматрајући посебно сва три типа Хорнових клауза. Према леми 1.31(ii), валуација v коју враћа HORN-SAT алгоритам, а која задовољава φ, мања је или једнака од било које валуације која задовољава φ, што значи да HORN-SAT алгоритам управо враћа најмање решење.

Задатак 3.12. Доказати да није свака формула еквивалентна некој Хорновој формули.

4.9. 2SAT. *2-клауза* је клауза у којој учествују највише два литерала. Проблем *2SAT* је проблем одлучивања да ли је дата конјункција 2-клауза

задовољива. У примеру 1.28 дати проблем свели смо управо на једну инстанцу проблема 2SAT. У овом делу ћемо доказати да је и проблем 2SAT у **P**.

Ако у некој 2-клаузи учествује само један литерал ℓ, записаћемо такву клаузу као $\ell \vee \ell$ користећи $\ell \equiv \ell \vee \ell$. Дакле, можемо да претпоставимо да је дата формула конјункција 2-клауза у којима учествују по тачно два литерала.

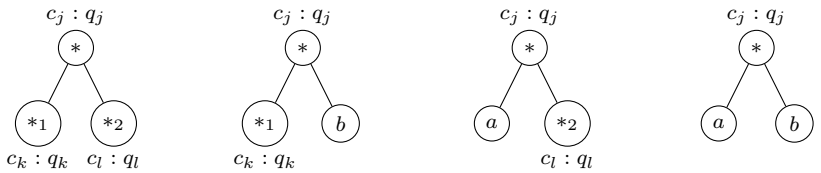
4.10. Цејтинова трансформација. Већ смо напоменули да дужина КНФ произвољне формуле φ може да буде експоненцијална у односу на дужину формуле φ. Међутим, постоји поступак којим се формули φ може придружити *еквизадовољива* формула φ_C у КНФ чија је дужина линеарна у односу на дужину формуле φ.

Дефиниција 1.35. Формуле φ и ψ су *еквизадовољиве* ако важи:

φ је задовољива ако и само ако ψ је задовољива.

Нека је φ формула сложености бар један (φ није слово и није константа) над словима $\mathcal{P}_\varphi = \{p_0, p_1, \ldots, p_n\}$. Уочимо њено дрво изградње. Нека су $c_0, c_1, \ldots, c_m$ чворови дрвета који нису лишће (тј. сваки од c_j је означен логичким везником), где је c_0 корен дрвета. Сваком c_j доделимо *ново* слово q_j. Формули φ придружујемо њену *Цејтинову формулу* φ_C над словима $\{p_0, p_1, \ldots, p_n, q_0, q_1, \ldots, q_m\}$ на следећи начин. Уочимо чвор c_j. Ако је c_j означен негацијом ($\neg$) имамо да је његов наследник или неки од преосталих чворова c_k (означен неким везником $*$) или је лист означен са a, где је a неко од слова p_l или нека константа $\bot$ или $\top$; на слици:

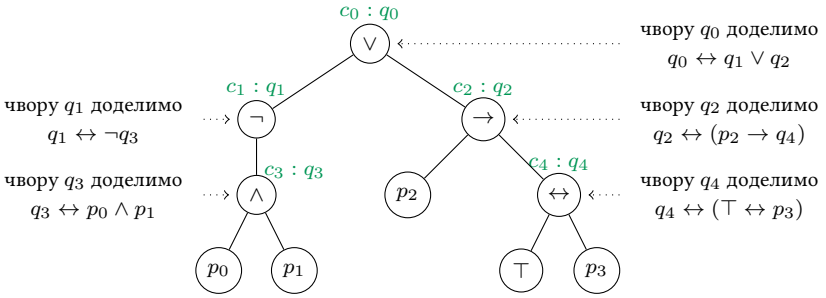
У првом случају чвору c_j доделимо формулу $\gamma_j \overset{def}{=} q_j \leftrightarrow \neg q_k$, а у другом случају формулу $\gamma_j \overset{def}{=} q_j \leftrightarrow \neg a$. Ако је c_j означен неким бинарним везником $*$, онда c_j има два наследника од којих сваки може да буде или неки од чворова c_k или неки лист; на слици су приказане све могућности:



У првом случају чвору c_j доделимо формулу $\gamma_j \overset{def}{=} q_j \leftrightarrow (q_k * q_l)$, у другом $\gamma_j \overset{def}{=} q_j \leftrightarrow (q_k * b)$, у трећем $\gamma_j \overset{def}{=} q_j \leftrightarrow (a * q_l)$, и у последњем $\gamma_j \overset{def}{=} q_j \leftrightarrow (a * b)$. Сада Цејтинову формулу φ_C дефинишемо да буде:

$$\varphi_C \overset{def}{=} q_0 \wedge \bigwedge_{j=0}^{m} \gamma_j.$$

Пример 1.36. Посматрајмо дрво изградње формуле $\varphi = \neg(p_0 \wedge p_1) \vee (p_2 \rightarrow (\top \leftrightarrow p_3))$:



Придружена Цејтинова формула φ_C сад је конјункција формула q_0, $q_0 \leftrightarrow q_1 \vee q_2$, $q_1 \leftrightarrow \neg q_3$, $q_2 \leftrightarrow (p_2 \rightarrow q_4)$, $q_3 \leftrightarrow p_0 \wedge p_1$ и $q_4 \leftrightarrow (\top \leftrightarrow p_3)$.

Теорема 1.37. Нека је φ формула над $\mathcal{P}_\varphi$, и нека је φ_C Цејтинова формула придружена φ над словима $\mathcal{P}_{\varphi_C} = \mathcal{P}_\varphi \cup \mathcal{Q}$. Пресликавање:

$$r : \mathrm{Mod}(\varphi_C) \rightarrow \mathrm{Mod}(\varphi)$$

дато рестрикцијом (за $v \in \mathrm{Mod}(\varphi_C)$, $r(v) = v|_{\mathcal{P}_\varphi}$) је добро дефинисана бијекција.

Доказ. Означимо са ψ_j потформулу формуле φ одређену чвором c_j (ком смо доделили исказно слово q_j и одговарајући коњункт γ_j у

φ_C). Дакле, $\psi_0 = \varphi$ јер смо корен дрвета означили са c_0. Најпре ћемо доказати *лему*: ако валуација v слова $\mathcal{P}_{\varphi_C}$ задовољава све конјункте γ_j формуле φ_C, онда $\psi_j[v] = v(q_j)$ за све j.

Доказ изводимо индукцијом по сложености формуле ψ_j. Како је формула ψ_j сложености бар један (јер ψ_j одговара чвору c_j који није лист), базни случај је $\mathrm{sl}(\psi_j) = 1$. Ако је c_j означен са $\neg$, његов наследник је лист означен са a, где је a неко слово p_i, константа $\bot$ или константа $\top$. Формула ψ_j је $\neg a$, а формула γ_j је $q_j \leftrightarrow \neg a$. Како $v \models \gamma_j$, директно имамо $\psi_j[v] = (\neg a)[v] = v(q_j)$. Ако је c_j означен са бинарним везником $*$, његови наследници су листови означени са a и b, где је сваки од a и b неко од слова p_i или нека од константи. Тада је ψ_j формула $a * b$, а γ_j је формула $q_j \leftrightarrow a * b$. Поново, како $v \models \gamma_j$, $\psi_k j[v] = (a * b)[v] = v(q_j)$. Тиме смо завршили доказ базе индукције.

Нека је ψ_j формула сложености $n \geqslant 2$. Ако је c_j означен са $\neg$, и његов наследник је чвор c_k, тада је ψ_j формула $\neg\psi_k$, γ_j је $q_j \leftrightarrow \neg q_k$, а по индукцијској хипотези је $\psi_k[v] = v(q_k)$. Одатле је директно $\psi_j[v] = (\neg\psi_k)[v] = (\neg q_k)[v] = v(q_j)$, где последња једнакост важи јер $v \models \gamma_j$. Нека је сада c_j означен са неким бинарним везником $*$. Ако су наследници чвора c_j чворови c_k и c_l (који нису листови), поступамо слично: тада је $\psi_j = \psi_k * \psi_l$, $\gamma_j = q_j \leftrightarrow (q_k * q_l)$, по индукцијској хипотези је $\psi_k[v] = v(q_k)$ и $\psi_l[v] = v(q_l)$, и, коначно, $v \models \gamma_j$; одатле, $\psi_j[v] = (\psi_k * \psi_l)[v] = (q_k * q_l)[v] = v(q_j)$. Врло слично изводимо закључак и ако је један од наследника чвора c_j лист. Овиме смо завршили доказ леме.

Према леми, за $v \in \mathrm{Mod}(\varphi_C)$ важи $\varphi[v] = \psi_0[v] = v(q_0) = 1$, где последња једнакост важи јер је q_0 један од конјуката формуле ψ_C; према томе, $r(v) \in \mathrm{Mod}(\varphi)$, тј. r јесте добро дефинисана функција. Да бисмо видели да је r бијекција, довољно је да докажемо да свака валиација $u \in \mathrm{Mod}(\varphi)$ има јединствено проширење $v \in \mathrm{Mod}(\varphi_C)$. Јединственост следи из леме: ако су v_1 и v_2 два проширења од u, онда је $v_1(q_j) = \psi_j[v_1] = \psi_j[u] = \psi_j[v_2] = v_2(q_j)$. Остаје да докажемо егзистенцију. Дефинишемо проширење од u са $v(q_j) \overset{def}{=} \psi_j[u]$. Слично као у доказу леме, индукцијом по сложености формуле

ψ_j можемо да видимо да $v \models \gamma_j$ за све j, а одатле је специјално и $v(q_0) = \psi_0[v] = \varphi[v] = \varphi[u] = 1$, па v задовољава све конјункте формуле φ_c, тј. $v \in \mathrm{Mod}(\varphi_C)$.

Претходна теорема директно нам каже да је формула φ задовољива ако и само ако је њена Цејтинова формула φ_C задовољива, тј. ове две формуле су еквизадовољиве. Штавише, ако израчунамо неку валуацију која задовољава φ_C, теорема нам каже да њена рестрикција задовољава φ. Приметимо такође да је сложеност формуле φ_C линеарна у односу на сложеност формуле φ (конкретно, није тешко видети да је $\mathrm{sl}(\varphi_C) = 3\,\mathrm{sl}(\varphi)$), као и да конструкцију формуле φ_C можемо спровести у полиномијалном времену по сложености формуле φ. Предност формуле φ_C је што конструкцију њеног КНФ-а можемо спровести у линеарном времену по њеној сложености (па, дакле, и по сложености формуле φ) користећи следеће логичке еквиваленције:

$$\begin{aligned}
p \leftrightarrow \neg q &\equiv (p \vee q) \wedge (\neg p \vee \neg q);\\
p \leftrightarrow q \wedge r &\equiv (p \vee \neg q \vee \neg r) \wedge (\neg p \vee q) \wedge (\neg p \vee r);\\
p \leftrightarrow q \vee r &\equiv (p \vee \neg q) \wedge (p \vee \neg r) \wedge (\neg p \vee q \vee r);\\
p \leftrightarrow (q \to r) &\equiv (p \vee q) \wedge (p \vee \neg r) \wedge (\neg p \vee \neg q \vee r);\\
p \leftrightarrow (q \leftrightarrow r) &\equiv (p \vee q \vee r) \wedge (p \vee \neg q \vee \neg r) \wedge\\
&\quad\ (\neg p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r);\\
p \leftrightarrow (q \veebar r) &\equiv (\neg p \vee q \vee r) \wedge (p \vee \neg q \vee r) \wedge\\
&\quad\ (p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee \neg r).
\end{aligned}$$

Приметимо да, иако формуле φ и φ_C јесу еквизадовољиве, оне *нису* логички еквивалентне. Примера ради, за $\varphi = \neg p$, $\varphi_C = q \wedge (q \leftrightarrow \neg p)$, и за валуацију $v = \left(\begin{smallmatrix} p & q \\ 0 & 0 \end{smallmatrix}\right)$ имамо $\varphi[v] = 1$, али $\varphi_C[v] = 0$.

5. Резолуција

5.1. Правило резолуције и доказ у резолуцији. И даље се бавимо формулама у КНФ. Злоупотребом нотације, клаузе ћемо записивати као скупове литерала, док ћемо формуле посматрати као скупове клаузе. Такође, ако су C и C' клаузе и ℓ литерал, ознака $C = C'\ell$ је краћи запис за $C = C' \cup \{\ell\}$, $C' = C \setminus \ell$ за $C' = C \setminus \{\ell\}$, а CC' за $C \cup C'$. Слично, ако су

φ и φ' формуле и C клауза, $\varphi = \varphi' C$ значи $\varphi = \varphi' \cup \{C\}$, а $\varphi' = \varphi \setminus C$ за $\varphi' = \varphi \setminus \{C\}$.

С обзиром да за клаузу C (посматрану као скуп литерала) имамо да $v \models C$ ако и само ако $(\exists \ell \in C)\ v \models \ell$, за $C = \emptyset$ имамо $v \models \emptyset$ ако и само ако $(\exists \ell \in \emptyset)\ v \models \ell$, што је логички нетачно[3], па дакле празна клаузу представља контрадикцију; у будућности ћемо празну клаузу обележавати са $\square$. Слично, што ће нам бити мање важно, за формулу φ (посматрану као скуп клауза) имамо да $v \models \varphi$ ако и само ако $(\forall C \in \varphi)\ v \models C$, па за $\varphi = \emptyset$, имамо $v \models \emptyset$ ако и само ако $(\forall C \in \emptyset)\ v \models C$, што је логички тачно[4]. Дакле, $\emptyset$ као скуп клауза представља таутологију.

Резолуција је формални систем који служи за дедуктивни рачун клауза. Резолуција има само једно правило.

Дефиниција 1.38. 1. Нека су C_1, C_2 клаузе и p слово. *Резолвента* клауза C_1 и C_2 *у односу на* слово p је клауза:

$$\mathrm{Res}(C_1, C_2; p) \overset{def}{=} (C_1 \setminus p)(C_2 \setminus \neg p).$$

2. *Правило резолуције* је:

$$\frac{C_1 \quad C_2}{\mathrm{Res}(C_1, C_2; p)},$$

где су C_1, C_2 произвољне клаузе и p произвољно слово.

Лема 1.39. (Лема о сагласности правила резолуције) Ако $v \models C_1$ и $v \models C_2$, онда $v \models \mathrm{Res}(C_1, C_2; p)$.

Доказ. Претпоставимо $v \models C_1$ и $v \models C_2$. То значи да имамо литерале $\ell_1 \in C_1$ и $\ell_2 \in C_2$ такве да $v \models \ell_1$ и $v \models \ell_2$; приметимо да ℓ_1 и ℓ_2 нису супротни литерали. Ако $\ell_1 \neq p$, онда $\ell_1 \in \mathrm{Res}(C_1, C_2; p)$, па зато $v \models \mathrm{Res}(C_1, C_2; p)$. Ако $\ell_1 = p$, онда $\ell_2 \neq \neg p$, па $\ell_2 \in \mathrm{Res}(C_1, C_2; p)$, и зато опет $v \models \mathrm{Res}(C_1, C_2; p)$.

[3] Подсетимо се да $(\exists x \in \emptyset)\ P(x)$ значи $(\exists x)(x \in \emptyset \wedge P(x))$; како је $x \in \emptyset \equiv \bot$, то је и $x \in \emptyset \wedge P(x) \equiv \bot$, па и $(\exists x)(x \in \emptyset \wedge P(x)) \equiv \bot$.

[4] Подсетимо се да $(\forall x \in \emptyset)\ P(x)$ значи $(\forall x)(x \in \emptyset \to P(x))$; како је $x \in \emptyset \equiv \bot$, то је $x \in \emptyset \to P(x) \equiv \top$, па је и $(\forall x)(x \in \emptyset \to P(x)) \equiv \top$.

У литератури се правило резолуције обично наводи са додатним условом. Наиме, тражи се да су клаузе C_1, C_2 и слово p такви да $p \in C_1$ и $\neg p \in C_2$. Иако у складу са датом дефиницијом ово није обавезно, имајући у виду претходну лему, приметимо да је у пракси једино разумно примењивати правило резолуције на клаузе које задовољавају наведени услов. Разлог за то је што, ако је слово p такво да $p \notin C_1$ или $\neg p \notin C_2$, онда $C_1 \subseteq \mathrm{Res}(C_1, C_2; p)$ или $C_2 \subseteq \mathrm{Res}(C_1, C_2; p)$ (или оба), што у сваком случају даје слабији закључак у односу на дате премисе.

Дефиниција 1.40. Нека је φ формула и C клауза.

1. *Доказ у резолуцији* клаузе C из *премиса* φ је коначан низ клауза $C_1, C_2, \ldots, C_n$ такав да је $C_n = C$ и за свако $i \leqslant n$ важи:
   - $C_i \in \varphi$ (C_i је *премиса*), или
   - $C_i = \mathrm{Res}(C_j, C_k; p)$ за неке $j, k < i$ и неко слово p.
2. Клауза C је *доказива у резолуцији* из премиса φ, у ознаци $\varphi \vdash_{\mathrm{Res}} C$, ако постоји доказ у резолуцији за C из премиса φ.

Пример 1.41. Један доказ за $\neg pq, \neg q\neg r, p\neg r \vdash_{\mathrm{Res}} \neg r$ је низ:

$$\neg pq, \neg q\neg r, p\neg r, \neg p\neg r, \neg r,$$

где су прве три клаузе у низу премисе, четврта је $\mathrm{Res}(\neg pq, \neg q\neg r; q)$, и последња је $\mathrm{Res}(p\neg r, \neg p\neg r; p)$. Доказ је читљивији ако га запишемо на следећи начин:

1.	$\neg pq$	премиса
2.	$\neg q\neg r$	премиса
3.	$p\neg r$	премиса
4.	$\neg p\neg r$	$\mathrm{Res}(1, 2; q)$
5.	$\neg r$	$\mathrm{Res}(3, 4; p)$

где у последњој колони наводимо образложење датог корака у доказу, и где $\mathrm{Res}(1, 2; q)$ значи да смо применили правило резолуција на клаузе из корака 1 и 2 (у односу на слово q). Још један начин да запишемо доказ био би у форми дрвета:

$$\dfrac{p\neg r \qquad \dfrac{\neg pq \qquad \neg q\neg r}{\neg p\neg r}\,(q)}{\neg r}\,(p)$$

где су листови дате премисе, а са стране у сваком дедуктивном кораку нагласимо у односу на које слово је примењено правило резолуције.

Запишимо и доказ за $p\neg q, q\neg r, \neg p, qrs, \neg s \vdash_{\text{Res}} \square$.

1. $p\neg q$ премиса
2. $q\neg r$ премиса
3. $\neg p$ премиса
4. qrs премиса
5. $\neg s$ премиса
6. $\neg q$ $\text{Res}(1, 3; p)$
7. qs $\text{Res}(4, 2; r)$
8. q $\text{Res}(7, 5; s)$
9. $\square$ $\text{Res}(8, 6; q)$

$$\dfrac{\dfrac{p\neg q \quad \neg p}{\neg q}\,(p) \qquad \dfrac{\dfrac{qrs \quad q\neg r}{qs}\,(r) \qquad \neg s}{q}\,(s)}{\square}\,(q)$$

5.2. Теорема потпуности.

Лема 1.42. Нека $\varphi, C \vdash_{\text{Res}} D$, и нека је ℓ литерал. Тада $\varphi, C\ell \vdash_{\text{Res}} D$ или $\varphi, C\ell \vdash_{\text{Res}} D\ell$.

Доказ. Доказ изводимо потпуном индукцијом по дужини (рецимо најкраћег) доказа за $\varphi, C \vdash_{\text{Res}} D$. Ако $\varphi, C \vdash_{\text{Res}} D$ има доказ дужине један, он је сачињен од само једне клаузе, и та клауза мора бити D. По дефиницији доказа, D такође мора бити премиса. Ако је $D \in \varphi$, онда очигледно $\varphi, C\ell \vdash_{\text{Res}} D$, а ако је $D = C$, онда очигледно $\varphi, C\ell \vdash_{\text{Res}} D\ell$.

Претпоставимо сада да је најкраћи доказ за $\varphi, C \vdash_{\text{Res}} D$ дужине $n > 1$; нека је $C_1, C_2, \ldots, C_n = D$ тај доказ. Приметимо да формула D на крају тог доказа не може бити премиса, јер бисмо имали краћи доказ (доказ дужине један). Дакле, $D = \text{Res}(C_i, C_j; p)$ за неке $i, j < n$ и неко слово p. Како су $C_1, \ldots, C_i$ и $C_1, \ldots, C_j$ докази редом за $\varphi, C \vdash_{\text{Res}} C_i$ и $\varphi, C \vdash_{\text{Res}} C_j$, и како су оба краћа од n, по индукцијској хипотези важи $\varphi, C\ell \vdash_{\text{Res}} C_i$ или $\varphi, C\ell \vdash_{\text{Res}} C_i\ell$, и $\varphi, C\ell \vdash_{\text{Res}} C_j$ или $\varphi, C\ell \vdash_{\text{Res}} C_j\ell$. Имамо четири случаја да размотримо.

Ако $\varphi, C\ell \vdash_{\text{Res}} C_i$ и $\varphi, C\ell \vdash_{\text{Res}} C_j$, како је $D = \text{Res}(C_i, C_j; p)$, директно закључујемо $\varphi, C\ell \vdash_{\text{Res}} D$.

Претпоставимо $\varphi, C\ell \vdash_{\text{Res}} C_i$ и $\varphi, C\ell \vdash_{\text{Res}} C_j\ell$. Ако је $\ell \neq \neg p$, онда је $\text{Res}(C_i, C_j\ell; p) = D\ell$, па закључујемо $\varphi, C\ell \vdash_{\text{Res}} D\ell$. Ако је $\ell = \neg p$, онда је $\text{Res}(C_i, C_j\ell; p) = D$, па закључујемо $\varphi, C\ell \vdash_{\text{Res}} D$.
Преостала два случаја испитујемо на сличан начин.

Ако је $D = \square$, директно имамо следећу последицу.

Последица 1.43. Нека $\varphi, C \vdash_{\text{Res}} \square$, и нека је ℓ литерал. Тада $\varphi, C\ell \vdash_{\text{Res}} \square$ или $\varphi, C\ell \vdash_{\text{Res}} \ell$.

Редефинишемо за тренутак појам сложености клаузе и формуле. За клаузу C ставимо $\text{sl}(C) \overset{def}{=} |C| - 1$ (што је број знакова дисјункције у стандардном запису клаузе); специјално, $\text{sl}(\square) = -1$, $\text{sl}(\{\ell\}) = 0$ за било који литерал ℓ, и $\text{sl}(C) \geqslant 1$ ако је C бар двочлана. За формулу φ ставимо:

$$\text{sl}(\varphi) \overset{def}{=} \sum_{C \in \varphi} \text{sl}(C).$$

(Приметимо да је $\text{sl}(\varphi) \geqslant -1$.)

Теорема 1.44. (Теорема потпуности за резолуцију)
Формула φ је задовољива ако и само ако $\varphi \nvdash_{\text{Res}} \square$.

Доказ. ($\Rightarrow$) Претпоставимо $v \models \varphi$, и, свођењем на противречност, претпоставимо $\varphi \vdash_{\text{Res}} \square$; нека је $C_1, C_2, \ldots, C_n = \square$. Индукцијом по $i \leqslant n$ доказујемо $v \models C_i$. Ако је C_i премиса, очигледно $v \models C_i$ јер $v \models \varphi$. Ако $C_i = \text{Res}(C_j, C_k; p)$ за неке $j, k < i$ и неко слово p, онда $v \models C_i$ према леми 1.39 јер $v \models C_j$ и $v \models C_k$ по индукцијској хипотези.
Специјално, $v \models C_n = \square$, што је контрадикција.
($\Leftarrow$) Претпоставимо, за доказ контрапозиције, φ није задовољива. Ако $\square \in \varphi$, очигледно $\varphi \vdash_{\text{Res}} \square$, па можемо да претпоставимо $\square \notin \varphi$. Тада је $\text{sl}(\varphi) \geqslant 0$, и доказ да $\varphi \vdash_{\text{Res}} \square$ изводимо индукцијом по $\text{sl}(\varphi)$. Ако је $\text{sl}(\varphi) = 0$, како $\emptyset \notin \varphi$, мора бити да је φ скуп једночланих клауза. Како φ није задовољива, две клаузе морају бити супротстављне,

тј. постоји слово p тако да $C_1 = \{p\}$ и $C_2 = \{\neg p\}$ припадају φ. Тада $\square = \mathrm{Res}(C_1, C_2; p)$, што доказује $\varphi \vdash_{\mathrm{Res}} \square$.
Претпоставимо сада $\mathrm{sl}(\varphi) \geqslant 1$. То значи да постоји $C \in \varphi$ таква да $\mathrm{sl}(C) \geqslant 1$, тј. $|C| \geqslant 2$. Нека је $\ell \in C$, и нека је $C' = C \smallsetminus \ell$; приметимо C' је непразна, и обе клаузе C' и $\{\ell\}$ су сложености мање од $\mathrm{sl}(C)$. Нека је $\varphi' = \varphi \smallsetminus C$; приметимо $\mathrm{sl}(\varphi' C'), \mathrm{sl}(\varphi'\{\ell\}) < n$, па по индукцијској хипотези $\varphi', C' \vdash_{\mathrm{Res}} \square$ и $\varphi', \{\ell\} \vdash_{\mathrm{Res}} \square$. Из $\varphi', C' \vdash_{\mathrm{Res}} \square$, према последици 1.43, $\varphi', C \vdash_{\mathrm{Res}} \square$ или $\varphi', C \vdash_{\mathrm{Res}} \ell$, јер је, сетимо се, $C = C'\ell$. Ако $\varphi', C \vdash_{\mathrm{Res}} \square$, завршили смо јер је $\varphi = \varphi' C$. Ако $\varphi', C \vdash_{\mathrm{Res}} \ell$, онда, користећи $\varphi', \{\ell\} \vdash_{\mathrm{Res}} \square$, опет закључујемо $\varphi', C \vdash_{\mathrm{Res}} \square$, тј. $\varphi \vdash_{\mathrm{Res}} \square$.
Завршили смо доказ.

5.3. Дужина доказа у резолуцији. Нека је $[n] = \{1, 2, \ldots, n\}$. Нека су $p_{i,j}$, $1 \in [n]$ и $j \in [n-1]$, исказна слова. Свака валуација v ових слова одређује једну релацију $R_v \subseteq [n] \times [n-1]$ на следећи начин:

$$R_v(i,j) \overset{def}{\Leftrightarrow} v(p_{i,j}) = 1,$$

али и обрнуто, свака релација $R \subseteq [n] \times [n-1]$ одређује једну слова v_R са:

$$v_R(p_{i,j}) = 1 \overset{def}{\Leftrightarrow} R(i,j).$$

Очигледно је $v_{R_v} = v$ и $R_{v_R} = R$, тј. на овај начин смо утврдили једну бијективну кореспонденцију између свих евалуација датих слова и свих релација из $[n]$ у $[n-1]$.

Посматрајмо сада формулу:

$$\varphi_n \overset{def}{=} \bigwedge_{i=1}^{n} \bigwedge_{1 \leqslant j_1 < j_2 < n} (\neg p_{i,j_1} \vee \neg p_{i,j_2}).$$

Приметимо да $v \models \varphi_n$ ако и само ако је придружена релација R_v парцијална функција, тј. за свако $i \in [n]$ постоји највише једно $j \in [n-1]$ тако да $R_v(i,j)$; као што је и обичај, у овом случају ћемо да пишемо $j = R_v(i)$ ако за $i \in [n]$ постоји (обавезно јединствено) $j \in [n-1]$ тако да $R_v(i,j)$; ако такво j не постоји, пишемо $R_v(i) \uparrow$ (што читамо као $R_v(i)$ *је недефинисано*).

Уочимо и формулу:

$$\iota_n \overset{def}{=} \bigwedge_{1 \leqslant i_1 < i_2 \leqslant n} \bigwedge_{j=1}^{n} (\neg p_{i_1,j} \wedge \neg p_{i_2,j});$$

приметимо $v \models \varphi_n \wedge \iota_n$ ако и само ако је парцијална функција R_v и 1-1. Такође, за:

$$\nu_n \overset{def}{=} \bigwedge_{j=1}^{n-1} \bigvee_{i=1}^{n} p_{i,j},$$

$v \models \varphi_n \wedge \nu_n$ ако и само ако је парцијална функција R_v и на, тј. $v \models \varphi_n \wedge \iota_n \wedge \nu_n$ ако и само ако је R_v парцијална бијекција. Таква R_v просто је бијекција између неког $(n-1)$-точланог подскупа од $[n]$ и $[n-1]$, и за тачно једно $i \in [n]$ имамо $R_v(i) \uparrow$. Формулу $\varphi_n \wedge \iota_n \wedge \nu_n$ краће ћемо обележавати са κ_n; приметимо да је κ_n формула у КНФ.

Посматрајмо формуле:

$$C_{n,i} \overset{def}{=} \bigvee_{j=1}^{n-1} p_{i,j}, \text{ за } i \in [n], \quad \text{и} \quad \psi_{n,I} \overset{def}{=} \bigwedge_{i \in I} C_{n,i}, \text{ за } I \subseteq [n].$$

Формуле $C_{n,i}$ очигледно су клаузе. За $v \models \kappa_n$, $v \models C_{n,i}$ ако и само ако $R_v(i) \downarrow$ (слика $R_v(i)$ је дефинисана). Како смо раније напоменули, пошто не постоји бијекција $[n] \to [n-1]$, $\kappa_n \wedge \psi_{n,[n]}$ није задовољива формула, па према теореми 1.44, $\kappa_n, \psi_{n,[n]} \vdash_{\text{Res}} \square$. Наш циљ у овом одељку је да докажемо да доказ у резолуцији за $\kappa_n, \psi_{n,[n]} \vdash_{\text{Res}} \square$ не може бити кратак; прецизније, доказаћемо:

Теорема 1.45. Постоји константа $c > 0$ (независна од n) таква да је сваки доказ у резолуцији за $\kappa_n, \psi_{n,[n]} \vdash_{\text{Res}} \square$ дужине бар 2^{cn}.

За доказ претходне теореме нам је потребна кратка припрема. С обзиром да је $\kappa_n \wedge \psi_{n,[n]}$ контрадикција, за сваку клаузу D имамо $\kappa_n, \psi_{n,[n]} \models D$. Означимо са $w(D) = \min\{|I| \colon I \subseteq [n],\ \kappa_n, \psi_{n,I} \models D\}$. За број $w(D)$ кажемо да је *ширина* клаузе D, и према претходној напомени је јасно да је $w(D) \leqslant n$. Такође, приметимо да је $w(\square) = n$ јер $\kappa_n, \psi_{n,I} \not\models \square$ за $I \subsetneq [n]$.

За низ *позитивних* клауза (клаузе у којима су сви литерали позитивни) $D_1, \ldots, D_m$ кажемо да је *псеудодоказ* за $\kappa_n, \psi_{n,[n]} \models \square$ ако је $D_m = \square$ и за све $k \leqslant m$ важи:

- $\kappa_n, C_{n,i} \models D_k$ за неко $i \in [n]$; или
- $\kappa_n, D_s, D_t \models D_k$ за неке $s, t < k$.

За клаузу D, означимо са $\widehat{D}$ позитивну клаузу добијену од D тако што сваки литерал $\neg p_{i,j}$ у D заменимо са $\bigvee_{i' \neq i} p_{i',j}$.

Лема 1.46. 1. За све клаузе D важи $\kappa_n \models D \leftrightarrow \widehat{D}$.
2. Ако је $D_1, \ldots, D_m$ доказ за $\kappa_n, \psi_{n,[n]} \vdash_{\text{Res}} \square$, онда је $\widehat{D}_1, \ldots, \widehat{D}_m$ псеудодоказ за $\kappa_n, \psi_{n,[n]} \models \square$.

Доказ. 1. Ако $v \models \kappa_n$, R_v је парцијална бијекција $[n] \to [n-1]$ таква да за тачно једно $i_0 \in [n]$ важи $R_v(i) \uparrow$, па лако можемо да видимо да $v \models \neg p_{i,j} \leftrightarrow \bigvee_{i' \neq i} p_{i',j}$ за све i и j. Одавде директно следи $v \models D \leftrightarrow \widehat{D}$.
2. Јасно је $\widehat{D}_m = \square$. За $k \leqslant m$ знамо да је D_k или премиса или $D_k = \text{Res}(D_s, D_t; p)$ за неке $s, t < k$ и неко слово p. У првом случају је D_k или клауза из κ_n или нека од клауза $C_{n,i}$, па у оба случаја имамо $\kappa_n, C_{n,i} \models D_k$, па према 1. и $\kappa_n, C_{n,i} \models \widehat{D}_k$.
У другом случају, $D_k = \text{Res}(D_s, D_t; p)$, према леми 1.39 знамо $\models D_s \wedge D_t \to D_k$, па према 1. $\kappa_n \models \widehat{D}_s \wedge \widehat{D}_t \to \widehat{D}_k$, односно $\kappa_n, \widehat{D}_s, \widehat{D}_t \models \widehat{D}_k$.
Завршили смо доказ.

Према претходној леми, сваки доказ за $\kappa_n, \psi_{n,[n]} \vdash_{\text{Res}} \square$ трансформише се у псеудодоказ једнаке дужине, па за доказ теореме 1.45 је довољно наћи доњу границу за дужине псеудодоказа.

Лема 1.47. Претпоставимо $n > 4$. У сваком псеудодоказу за $\kappa_n, \psi_{n,[n]} \models \square$ постоји клауза са бар $\frac{n^2}{8}$ литерала.

Доказ. Посматрајмо псеудодоказ $D_1, \ldots, D_m = \square$. Како је $w(D_m) = n$, постоји најмањи број $k \leqslant m$ за који је $w(D_k) \geqslant \frac{n}{4}$. Ако $\kappa_n, C_{n,i} \models D_k$ за неко $i \in [n]$, онда је очигледно $w(D_k) \leqslant 1$,

па је $n \leqslant 4$, што је контрадикција. Дакле, постоје $t, s < k$ такви да $\kappa_n, D_t, D_s \models D_k$. По избору k, $w(D_t), w(D_s) < \frac{n}{4}$, па постоје $I_t, I_s \subseteq [n]$ такви да $|I_t|, |I_s| < \frac{n}{4}$ и $\kappa_n, \psi_{n,I_t} \models D_t$ и $\kappa_n, \psi_{n,I_s} \models D_s$, па $\kappa_n, \psi_{m,I_t \cup I_s} \models D_k$. Одатле $w(D_k) \leqslant |I_t \cup I_s| \leqslant |I_t| + |I_s| < \frac{n}{2}$. Дакле, $\frac{n}{4} \leqslant w(D_k) < \frac{n}{2}$.
Нека је $I_k \subseteq [n]$ такав да $|I_k| = w(D_k)$ и $\kappa_n, \psi_{n,I_k} \models D_k$; важи $\frac{n}{4} \leqslant |I_k| < \frac{n}{2}$. Доказаћемо да за свако $i \in I_k$ постоји 1-1 функција $f_i : [n] \setminus I_k \to [n-1]$ таква да за све $i' \notin I_k$, слово $p_{i,f_i(i')}$ је један од литерала у D_k; одавде D_k садржи бар $|I_k| \cdot |[n] \setminus I_k| > \frac{n}{4} \cdot (n - \frac{n}{2}) = \frac{n^2}{8}$ литерала, чиме завршавамо доказ.
Фиксирајмо $i \in I_k$. По избору I_k, $\kappa_n, \psi_{n,I_k \setminus \{i\}} \not\models D_k$, па постоји валуација v_i таква да $v_i \models \kappa_n \wedge \psi_{n,I_k \setminus \{i\}}$ и $v_i \not\models D_k$; приметимо $C_{n,i}[v_i] = 0$ јер $\kappa_n, \psi_{n,I_k} \models D_k$. Одатле $R_{v_i}(i) \uparrow$, па је $[n] \setminus I_k$ садржан у домену функције R_{v_i}. Нека је f_i рестрикција R_{v_i} на $[n] \setminus I_k$; јасно је да је f_i 1-1. Нека $i' \notin I_k$. Уочимо валуацију v' дату са: $v'(p_{i,f_i(i')}) = 1$, $v'(p_{i',f_i(i')}) = 0$ и v' се поклапа са v на преосталим словима. Функција $R_{v'}$ скоро је једнака R_v, осим што $R_{v'}(i') \uparrow$ и $R_{v'}(i) = f_i(i')$. Тада $v' \models \kappa_n \wedge \psi_{n,I_k}$, па одатле $v' \models D_k$. Из $v \not\models D_k$, $v' \models D_k$ и v' је једино $p_{i,f_i(i')}$ прогласила за ново тачно слово, закључујемо да ово слово мора да се налази у D_k.

Лема 1.48. Нека је $D_1, \ldots, D_m$ псеудодоказ за $\kappa_n, \psi_{n,[n]} \models \square$. Трансформишемо овај псеудодоказ на следећи начин:

- ако се слово $p_{n,n-1}$ налази у клаузи D_k, обришемо D_k;
- обришемо сва појављивања слова $p_{i,n-1}$, $i < n$, и слова $p_{n,j}$, $j < n-1$, из преосталих клауза.

Добијени низ је псеудодоказ за $\kappa_{n-1}, \psi_{n-1,[n-1]} \models \square$.

Доказ. Нека је $D'_1, \ldots, D'_{m'}$ низ добијен описаним поступком. Јасно је да је $D'_{m'} = \square$. Фиксирајмо $k' \leqslant m'$, посматрајмо клаузу $D'_{k'}$. Тада постоји $k \leqslant m$ и клауза D_k у којој се слово $p_{n,n-1}$ не појављује, и чијом смо трансформацијом добили $D'_{k'}$. Посматрамо два случаја.

1° Претпоставимо $\kappa_n, C_{n,i} \models D_k$ за неко $i \in [n]$. Ако је $i < n$, тврдимо да $\kappa_{n-1}, C_{n-1,i} \models D'_{k'}$. Нека је v валуација слова $p_{i,j}$, $i < n$ и $j < n-1$, таква да $v \models \kappa_{n-1}, C_{n-1,i}$; треба да докажемо $v \models D'_{k'}$. Приметимо, како $v \models \kappa_{n-1}$, да је R_v парцијална бијекција између $[n-1]$ и $[n-2]$. Додефинишемо је до парцијалне бијекције између $[n]$ и $[n-1]$ стављајући $R_v(n) = n-1$. (Формално, проширујемо валуацију v на слова $p_{n,n-1}$, $p_{i,n-1}$ и $p_{n,j}$, где $i < n$ и $j < n-1$, стављајући $v(p_{n,n-1}) = 1$, и $v(p_{i,n-1}) = v(p_{n,j}) = 0$.) За овако дефинисано проширење очигледно имамо $v \models \kappa_n$. Такође, како је $C_{n,i} = C_{n-1,i} \vee p_{i,n-1}$, јасно је да $v \models C_{n,i}$, па закључујемо $v \models D_k$. Међутим, како се $p_{n,n-1}$ не појављује у D_k, и сва слова $p_{i,n-1}$ и $p_{n,j}$ која бришемо из D_k су нетачна у v, закључујемо $v \models D'_{k'}$.

Нека је сада $i = n$, тј. $\kappa_n, C_{n,n} \models D_k$. Тада тврдимо да само $\kappa_{n-1} \models D'_{k'}$ (па и нпр. $\kappa_{n-1}, C_{n-1,1} \models D'_{k'}$). Нека је v валуација таква да $v \models \kappa_{n-1}$, и проширимо је на исти начин као и у претходном подслучају. Поново је јасно да $v \models \kappa_n$, али приметимо и да $v \models C_{n,n}$ јер слово $p_{n,n-1}$ учествује у $C_{n,n}$. Закључујемо $v \models D_k$, и на исти начин као у претходном подслучају добијамо $v \models D'_{k'}$.

2° Претпоставимо $\kappa_n, D_s, D_t \models D_k$ за неке $s, t < k$. Ако се $p_{n,n-1}$ појављује у обе клаузе D_s и D_t, на сличан начин као у другом подслучају у 1° видимо $\kappa_{n-1} \models D'_{k'}$, па нпр. и $\kappa_{n-1}, C_{n-1,1} \models D'_{k'}$. Претпоставимо да се $p_{n,n-1}$ појављује само у једној од D_s и D_t, нпр. у D_s. Тада се D_t трансформише у клаузу $D'_{t'}$, где $t' < k'$, и тврдимо $\kappa_{n-1}, D'_{t'} \models D'_{k'}$. Поступамо као и до сада, нека $v \models \kappa_{n-1} \wedge D'_{t'}$, и проширимо v на исти начин као и раније. Тада $v \models \kappa_n$, $v \models D_t$ јер $D'_{t'} \subseteq D_t$, и $v \models D_s$ јер се $p_{n,n-1}$ појављује у D_s. Закључујемо, $v \models D_k$, и поново као и раније, $v \models D'_{k'}$. Коначно, ако се $p_{n,n-1}$ не појављује ни у једној од клауза D_s и D_t, обе се трансформишу у клаузе $D'_{s'}$ и $D'_{t'}$ где $s', t' < k'$, и тврдимо $\kappa_{n-1}, D'_{s'}, D'_{t'} \models D'_{k'}$. Доказ изводимо на сличан начин као у претходном подслучају.

Доказ. (*Доказ теореме 1.45*) Као што смо већ напоменули, довољно је да нађемо доњу границу за дужине псеудодоказа за $\kappa_n, \psi_{n,[n]} \models \square$. Изаберимо произвољан псеудодоказ. За клаузу у псеудодоказу ћемо

рећи да је *дугачка* ако садржи бар $\frac{n^2}{32}$ слова. Нека је N број дугачких клауза у нашем псеудодоказу, нека је M_p број дугачких клауза у којима се појављује слово p, и нека је M највећи међу свим M_p. Посматрајмо скуп:

$$S = \{(D, p) \mid D \text{ је дугачка клауза и слово } p \in D\}.$$

Са једне стране, $|S| \geqslant N\frac{n^2}{32}$ јер свака од N дугачких клауза садржи бар $\frac{n^2}{32}$ слова, док са друге стране $|S| \leqslant n(n-1)M$ јер се свако од $n(n-1)$ слова појављује у највише M дугачких клауза. Одавде је:

$$M \geqslant \frac{n^2}{n(n-1)} \cdot \frac{N}{32} > \frac{N}{32}.$$

Дакле, постоји слово које се појављује у бар $\frac{N}{32}$ дугачких клауза. После преименовања индекса слова, можемо да претпоставимо да је уочено слово $p_{n,n-1}$. Примењујући трансформацију из леме 1.48 добијамо псеудодоказ за $\kappa_{n-1}, \psi_{n-1,[n-1]} \models \square$. У току ове трансформације, избрисали смо бар $\frac{N}{32}$ дугачких клауза, па добијени псеудодоказ има највише $\frac{31}{32}N$ дугачких клауза. (Нагласимо да „дугачка клауза" и даље значи клауза са бар $\frac{n^2}{32}$ слова.) Сада ћемо наставити поступак. Нека је N_1 број дугачких клауза у трансформисаном псеудодоказу, дакле $N_1 \leqslant \frac{31}{32}N$, нека је M_1 највећи број дугачких клауза у којима се појављује неко слово, и нека је скуп:

$$S_1 = \{(D, p) \mid D \text{ је дугачка клауза и слово } p \in D\}.$$

Поново је $|S_1| \geqslant N_1\frac{n^2}{32}$, и сада $|S_1| \leqslant (n-1)(n-2)M_1$, па је $M_1 \geqslant \frac{n^2}{(n-1)(n-2)} \cdot \frac{N_1}{32} > \frac{N_1}{32}$. Поново закључујемо да постоји слово које се појављује у бар $\frac{N_1}{32}$ дугачких клауза, и до на преименовање индекса, можемо да претпоставимо да говоримо о слову $p_{n-1,n-2}$. Примењујући трансформацију из леме 1.48, добијамо псеудодоказ за $\kappa_{n-2}, \psi_{n-2,[n-2]} \models \square$ у којем је број дугачких клауза N_2 највише $\frac{31}{32}N_1 \leqslant (\frac{31}{32})^2 N$.

Поступак ћемо урадити $\frac{n}{2}$ пута. Добијамо псеудодоказ за $\kappa_{\frac{n}{2}}, \psi_{\frac{n}{2},[\frac{n}{2}]} \models \square$ у којем је број дугачких клауза $N_{\frac{n}{2}}$ највише $(\frac{31}{32})^{\frac{n}{2}} N$. Међутим, према леми 1.47 добијени псеудодоказ има клаузу са бар

$\frac{1}{8}(\frac{n}{2})^2 = \frac{n^2}{32}$ слова, тј. $N_{\frac{n}{2}} \geqslant 1$. Дакле:

$$1 \leqslant N_{\frac{n}{2}} \leqslant \left(\frac{31}{32}\right)^{\frac{n}{2}} N, \text{ одакле } N \geqslant \left(\frac{32}{31}\right)^{\frac{n}{2}} = 2^{cn},$$

где је $c = \log_2 \sqrt{\frac{32}{31}} > 0$. Према томе, полазни псеудодоказ има бар 2^{cn} дугачких клауза, па има бар толико укупно клауза. Доказ је завршен.

6. DPLL алгоритам

Нека је φ формула у КНФ и нека је ℓ литерал. Са $\varphi|_\ell$ означавамо формулу добијену од φ на следећи начин:

- ако клауза $C \in \varphi$ садржи литерал ℓ, обришемо целу клаузу C;
- ако клауза $C \in \varphi$ садржи супротни литерал $\overline{\ell}$, обришемо литерал $\overline{\ell}$ из C;
- у осталим случајевима ништа не радимо.

Приметимо да се литерали ℓ и $\overline{\ell}$ не појављују у формули $\varphi|_\ell$. Такође, ако се неки од литерала ℓ и $\overline{\ell}$ појављује у φ, формула $\varphi|_\ell$ обавезно је мање сложености у односу на φ.

Лема 1.49. Нека је φ формула у КНФ и ℓ литерал. Нека је v валуација таква да $v \models \ell$. Тада $v \models \varphi$ ако и само ако $v \models \varphi|_\ell$.

Доказ. Ако $\ell \in C$, јасно $v \models C$. Ако $\overline{\ell} \in C$ и $C' = C \setminus \overline{\ell}$, јасно $v \models C$ ако и само ако $v \models C'$. Сада тврђење леме следи директно.

Последица 1.50. Нека је φ формула у КНФ и ℓ литерал. Ако је $\varphi|_\ell$ задовољива, онда је и φ задовољива.

Доказ. Нека $v \models \varphi|_\ell$, и нека је w валуација таква да $w \models \ell$ и $w(p) = v(p)$ за сва слова која нису ℓ, односно $\overline{\ell}$. Како се ℓ и $\overline{\ell}$ не појављују у формули $\varphi|_\ell$, $w \models \varphi|_\ell$. Према претходној леми, $w \models \varphi$. Дакле, φ је задовољива.

Обратна импликација у претходној последици не мора да важи у општем случају. На пример, ако је $\{p\}$ једина клауза у φ, онда $\varphi|_{\neg p}$ садржи само празну клаузу; φ је задовољива, док $\varphi|_{\neg p}$ није.

Последица 1.51. Нека је φ формула у КНФ и ℓ литерал. Тада је φ задовољива ако и само ако је бар једна од $\varphi|_\ell$ и $\varphi|_{\overline{\ell}}$ задовољива.

Доказ. Смер ($\Leftarrow$) важи према последици 1.50. За обртну импликацију, претпоставимо да је φ задовољива, и нека $v \models \varphi$. Тада важи или $v \models \ell$ или $v \models \overline{\ell}$. Према леми 1.49, у првом случају $v \models \varphi|_\ell$, а у другом $v \models \varphi|_{\overline{\ell}}$. Дакле, у првом случају је бар $\varphi|_\ell$ задовољива, а у другом је бар $\varphi|_{\overline{\ell}}$ задовољива.

У неким специјалним случајевима формуле φ и $\varphi|_\ell$ су еквизадовољиве.

Лема 1.52. (Лема о пропагирању једночлане клаузе) Ако формула φ садржи једночлану клаузу $\{\ell\}$, онда су φ и $\varphi|_\ell$ еквизадовољиве.

Доказ. Према последици 1.50, довољно је да докажемо да φ је задовољива повлачи $\varphi|_\ell$ је задовољива. Нека $v \models \varphi$. Како је $\{\ell\}$ једна од клауза у φ, $v \models \ell$. Према леми 1.49, $v \models \varphi|_\ell$; дакле, $\varphi|_\ell$ је задовољива.

Лема 1.53. (Лема о пропагирању монотоног литерала) Ако формула φ садржи монотони литерал ℓ, што значи да се литерал ℓ појављује, док се $\overline{\ell}$ не појављује у φ, онда су φ и $\varphi|_\ell$ еквизадовољиве.

Доказ. Према последици 1.50, довољно је да докажемо да φ је задовољива повлачи $\varphi|_\ell$ је задовољива. Нека $v \models \varphi$. Како ниједна клауза не садржи литерал $\overline{\ell}$, произвољна клауза $C \in \varphi$ је или обрисана у $\varphi|_\ell$ (ако $\ell \in C$) или је остала иста у $\varphi|_\ell$ (ако $\ell \notin C$). Према томе, $\varphi|_\ell \subseteq \varphi$, па $v \models \varphi|_\ell$; $\varphi|_\ell$ је задовољива.

Алгоритам (DPLL)
функција DPLL(φ), где је φ у КНФ
ако $\varphi = \emptyset$:

врати: „φ је задовољива”
ако $\square \in \varphi$:
врати: „φ није задовољива”
ако φ садржи једночлану клаузу $\{\ell\}$:
врати: DPLL$(\varphi|_\ell)$
ако φ садржи монотон литерал ℓ:
врати: DPLL$(\varphi|_\ell)$
изабери литерал ℓ који се појављује у φ
врати: DPLL$(\varphi|_\ell)$ **или** DPLL$(\varphi|_{\overline{\ell}})$

Теорема 1.54. Претходни алгоритам се зауставља и коректно враћа одговор на питање да ли је формула φ задовољива.

Доказ. Редефинишемо сложеност формуле φ у КНФ, $\mathrm{sl}(\varphi)$, да буде укупан број литерала у φ. Доказ теореме ћемо извести индукцијом по $\mathrm{sl}(\varphi)$. Претпоставимо да је $\mathrm{sl}(\varphi) = 0$. То је могуће ако је $\varphi = \emptyset$, или ако $\varphi \neq \emptyset$ и $\square \in \varphi$. У првом случају φ је задовољива, док у другом није, и приметимо да ће прве две ако наредбе у овом случају коректно вратити одговор на питање да ли је φ задовољива.
Ако је ℓ литерал који се појављује у φ, приметимо да је обавезно $\mathrm{sl}(\varphi|_\ell) < \mathrm{sl}(\varphi)$ и $\mathrm{sl}(\varphi|_{\overline{\ell}}) < \mathrm{sl}(\varphi)$. Претпоставимо сада да је $\mathrm{sl}(\varphi) > 0$. Тада $\varphi \neq \emptyset$, па прва ако наредба неће прекинути извршавање алгоритма. Могуће је да $\square \in \varphi$, у ком случају φ није задовољива, и друга ако наредба ће коректно вратити одговор.
Претпоставимо $\square \notin \varphi$. У том случају, алгоритам ће испитати трећу ако наредбу. Ако је тачно да φ садржи једночлану клаузу $\{\ell\}$, алгоритам ће започети извршавање функције DPLL са улазом $\varphi|_\ell$. Овај позив ће се по индукцијској хипотези (јер $\mathrm{sl}(\varphi|_\ell) < \mathrm{sl}(\varphi)$) завршити и коректно ће вратити одговор на питање да ли је $\varphi|_\ell$ задовољива. Исти одговор ће вратити и извршавање функције DPLL за улаз φ, што јесте коректан одговор на питање да ли је φ задовољива према леми 1.52.
Претпоставимо сада да φ не садржи једночлану клаузу. У том случају, алгоритам испитује четврту ако наредбу. Ако је тачно да φ

садржи монотони литерал ℓ, као у претходном пасусу, користећи индукцијску хипотезу и лему 1.53, закључујемо да се алгоритам завршава са коректним излазом.
Коначно, претпоставимо да φ не садржи монотони литерал. У том случају, алгоритам ће да изабере произвољан литерал ℓ који се појављује у φ (бар један такав постоји јер $\mathrm{sl}(\varphi) > 0$) и позваће функцију DPLL са улазима $\varphi|_\ell$ и $\varphi|_{\overline{\ell}}$. Оба позива ће се по индукцијској хипотези завршити и коректно ће вратити одговоре на питања да ли су формуле $\varphi|_\ell$ и $\varphi|_{\overline{\ell}}$ задовољиве. Алгоритам ће као одговор на питање да ли је φ задовољива вратити дисјункцију добијена два одговора, што јесте коректан излаз према последици 1.51.
Завршили смо доказ.

7. Теорема компактности

Дефиниција 1.55. Скуп формула Σ је:

1. *задовољив* ако постоји валуација v таква да $v \models \Sigma$;
2. *коначно задовољив (к.з.)* ако је сваки коначан подскуп Σ_0 од Σ задовољив;
3. *затворен за слова* ако за свако слово p важи $p \in \Sigma$ или $\neg p \in \Sigma$.

Лема 1.56. Нека је Σ к.з. скуп формула и φ једна формула. Тада је бар један од скупова $\Sigma \cup \{\varphi\}$ и $\Sigma \cup \{\neg\varphi\}$ к.з.

Доказ. Претпоставимо супротно, $\Sigma \cup \{\varphi\}$ и $\Sigma \cup \{\neg\varphi\}$ нису к.з. Тада имамо коначне подскупове $\Sigma_1, \Sigma_2 \subseteq \Sigma$ такве да $\Sigma_1 \cup \{\varphi\}$ и $\Sigma_2 \cup \{\neg\varphi\}$ нису задовољиви. Како је $\Sigma_1 \cup \Sigma_2$ коначан подскуп од Σ и Σ је к.з., $\Sigma_1 \cup \Sigma_2$ је задовољив; нека је v валуација таква да $v \models \Sigma_1 \cup \Sigma_2$. Како $v \models \Sigma_1$ и $\Sigma_1 \cup \{\varphi\}$ није задовољив, имамо да $v \not\models \varphi$. Слично, како $v \models \Sigma_2$ и $\Sigma_2 \cup \{\neg\varphi\}$ није задовољив, имамо да $v \not\models \neg\varphi$, тј. $v \models \varphi$. Контрадикција.

Лема 1.57. Нека је $(\Sigma_i)_{i \in I}$ ланац к.з. скупова. (Дакле, за све $i, j \in I$ важи $\Sigma_i \subseteq \Sigma_j$ или $\Sigma_j \subseteq \Sigma_i$.) Тада је и $\Sigma^* = \bigcup_{i \in I} \Sigma_i$ к.з.

Доказ. Нека је Γ коначан подскуп од Σ^*; запишимо $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$. Тада за свако $j \leqslant n$ имамо $\gamma_j \in \Sigma_{i_j}$ за неко $i_j \in I$. Како су $\Sigma_{i_1}, \ldots, \Sigma_{i_n}$ међусобно упоредиви, неки од њих, рецимо Σ_{i_k}, је највећи међу њима. Тада $\Gamma \subseteq \Sigma_{i_k}$, па како је Σ_{i_k} к.з., Γ је задовољив. Завршили смо доказ.

Надаље ће нам бити потребна следећа варијанта *аксиоме избора.*

Аксиома 1.58. (Хауздорфов принцип максималности)
Свако парцијално уређење има максималан ланац.

Лема 1.59. Ако је Σ к.з. скуп, онда постоји к.з. скуп Σ^* такав да $\Sigma \subseteq \Sigma^*$ и Σ^* је затворен за слова.

Доказ. Уочимо фамилију $\mathcal{S} = \{\Gamma \mid \Sigma \subseteq \Gamma,\ \Gamma \text{ је к.з.}\}$. Фамилија $\mathcal{S}$ је непразна јер $\Sigma \in \mathcal{S}$, и фамилија $\mathcal{S}$ јесте парцијално уређење у односу на релацију подскупа. По Хауздорфовом принципу максималности постоји максималан ланац $\mathcal{L} \subseteq \mathcal{S}$. Према леми 1.57, $\Sigma^* = \bigcup \mathcal{L}$ је к.з. скуп. Како сваки елемент ланца $\mathcal{L}$ садржи Σ то $\Sigma \subseteq \Sigma^*$. Дакле, треба још да докажемо да је Σ^* затворен за слова.
Нека је p слово. Према леми 1.56, бар један од скупова $\Sigma^* \cup \{p\}$ и $\Sigma^* \cup \{\neg p\}$ је к.з., тј. бар један од њих је у фамилија $\mathcal{S}$. Без умањења општости, нека је $\Sigma^* \cup \{p\}$ к.з., тј. у $\mathcal{S}$. Приметимо да је тада и $\mathcal{L} \cup \{\Sigma^* \cup \{p\}\}$ ланац у $\mathcal{L}$, па због максималности ланца $\mathcal{L}$ мора бити $\Sigma^* \cup \{p\} \in \mathcal{L}$. Тада је $\Sigma^* \cup \{p\} \subseteq \bigcup \mathcal{L} = \Sigma^*$, и закључујемо $p \in \Sigma^*$. Дакле, Σ^* је затворен за слова.

Ако је скуп слова $\mathcal{P}$ пребројив, претходну лему можемо да докажемо без коришћења аксиоме избора.

Задатак 3.13. Ако је $\mathcal{P}$ пребројив, доказати лему 1.59 без коришћења аксиоме избора.

Лема 1.60. Ако је Σ к.з. и затворен за слова, онда је Σ задовољив.

Доказ. Нека је Σ к.з. и затворен за слова. Како је $\{p, \neg p\}$ незадовољив скуп, за свако слово p важи тачно једно од $p \in \Sigma$ и $\neg p \in \Sigma$. Дефинишемо валуацију v са:

$$v(p) \overset{def}{=} \begin{cases} 1 & \text{ако } p \in \Sigma \\ 0 & \text{ако } p \notin \Sigma \ (\Leftrightarrow \neg p \in \Sigma) \end{cases}.$$

Подсетимо да са p^v обележавамо слово p ако је $v(p) = 1$, односно $\neg p$ ако је $v(p) = 0$. Према дефиницији v видимо да за свако слово p важи $p^v \in \Sigma$.
Тврдимо $v \models \Sigma$, што доказује да је Σ задовољив. Нека је $\varphi \in \Sigma$ произвољна формула и нека је $\mathcal{P}_\varphi = \{p_1, \ldots, p_n\}$. Тада је $\{\varphi, p_1^v, \ldots, p_n^v\}$ коначан подскуп од Σ, па како је Σ к.з. постоји валуација w таква да $w \models \varphi$, и $w \models p_i^v$ за све $i \leqslant n$. Из $w \models p_i^v$ следи $w(p) = v(p)$ за све $i \leqslant n$, тј. v и w се поклапају на словима формуле φ, па је $\varphi[v] = \psi[w]$. Како $w \models \varphi$, закључујемо $v \models \varphi$, чиме завршавамо доказ.

Теорема 1.61. (Теорема компактности) Скуп формула Σ је задовољив ако и само ако је Σ к.з.

Доказ. Смер $(\Rightarrow)$ је очигледан. За $(\Leftarrow)$ претпоставимо да је Σ к.з. Према леми 1.59 постоји $\Sigma^* \supseteq \Sigma$ који је к.з. и затворен за слова. Према леми 1.60, Σ^*, па самим тим и Σ, је задовољив.

Пример 1.62. Нека је $(S, \prec)$ (строго) парцијално уређење. Доказаћемо да се $\prec$ може проширити до линеарног уређења $<$ на S. Размотрићемо два случаја.
1° S је коначан. Егзистенцију жељеног проширења можемо доказати индукцијом по $|S|$. Ако је $|S| = 1$, $\prec$ је формално већ линеарно, тако да немамо шта да докажемо. Претпоставимо да је $|S| = n > 1$. Тада S, као коначан скуп, има минимални елемент у односу на $\prec$; нека је $a \in S$ неки минималан елемент. По индукцијској хипотези рестрикција уређења $\prec$ на $S \setminus \{a\}$ може се проширити до линеарног уређења $<$ на $S \setminus \{a\}$. Сада лако видимо да $<$ можемо проширити до линеарног уређења на S стављајући $a < x$ за све $x \in S \setminus \{a\}$. Такође,

лако видимо да је добијено линеарно уређење $<$ на S проширење полазног уређења $\prec$ на S.

2° S је бесконачан. Посматрајмо скуп слова $\mathcal{P} = \{p_{a,b} \mid a, b \in S\}$ и следеће скупове формула над $\mathcal{P}$:

- $\Sigma_\prec = \{p_{a,b} \mid a, b \in S,\ a \prec b\}$;
- $\Sigma_I = \{\neg p_{a,a} \mid a \in S\}$;
- $\Sigma_T = \{p_{a,b} \wedge p_{b,c} \to p_{a,c} \mid a, b, c \in S\}$;
- $\Sigma_L = \{p_{a,b} \vee p_{b,a} \mid a, b \in S\ a \neq b\}$;
- $\Sigma = \Sigma_\prec \cup \Sigma_I \cup \Sigma_T \cup \Sigma_L$.

Претпоставимо за тренутак да је Σ задовољив скуп и да је v валуација која га задовољава. Дефинишемо $<$ на S са: $a < b \overset{def}{\Leftrightarrow} v(p_{a,b}) = 1$. Како $v \models \Sigma_I \cup \Sigma_T$ видимо да је $<$ ирефлексивна и транзитивна релација, тј. строго парцијално уређење на S. Како $v \models \Sigma_L$, $<$ је такође и линеарно уређење. Коначно, како $v \models \Sigma_\prec$, $<$ је проширење уређења $\prec$.

Дакле, довољно је да докажемо да је Σ задовољив скуп. Заправо, према теореми компактности, довољно је да докажемо да је к.з. Нека је Σ_0 коначан подскуп од Σ, и нека је $S_0 \subseteq S$ коначан подскуп елемената из S који се јављају као индекси слова у скупу Σ_0. Уочимо и Σ_1 – скуп свих формула из Σ над словима $\mathcal{P}_0 = \{p_{a,b} \mid a, b \in S_0\}$; приметимо $\Sigma_0 \subseteq \Sigma_1 \subseteq \Sigma$, па је довољно да докажемо да је Σ_1 задовољив. Према 1°, рестрикција $\prec$ на S_0 има проширење до линеарног уређења $<$ на S_0. Дефинишемо валуацију v слова $\mathcal{P}_0$ са: $v(p_{a,b}) \overset{def}{\Leftrightarrow} a < b$. На сличан начин као у претходном пасусу, лако видимо да $v \models \Sigma_1$. Дакле, Σ_1 је задовољив, што завршава доказ.

Задатак 3.14. Доказати да је неусмерен граф k-обојив ако и само ако је сваки његов коначан подграф k-обојив.

Задатак 3.15. Доказати да постоји линеарно уређење $\prec$ на $\mathbb{Q}$ такво да ниједан аритметички низ дужине бар 3 није строго растући.

Глава 2

Логика првог реда