LAUSELOGIIKKA

1. Lauselogiikan kieli
2. Lauselogiikan semantiikka
3. Semanttiset peruskäsitteet
4. Semanttinen taulu
5. Vaihtoehtoisia todistusjärjestelmiä
6. Normaalimuodot
7. Resoluutio
8. Laskennallisesta vaativuudesta

1 Lauselogiikan kieli

- Lauselogiikan aakkosto
- Kielen määritelmä
- Lauseiden muodostamisesta
- Sopimukset sulkujen käytöstä
- Esimerkki: rakenteinen induktio

1.1 Lauselogiikan aakkosto

- atomiset lauseet: $A, A_1, A_2, \ldots, B, B_1, \ldots, C, \ldots$
- negaatiosymboli: $\neg$ (ei)
- konjunktiosymboli: $\wedge$ (ja)
- disjunktiosymboli: $\vee$ (tai)
- implikaatiosymboli: $\rightarrow$ (jos ... niin)
- ekvivalenssisymboli: $\leftrightarrow$ (jos ja vain jos)
- sulut: ()

Symboleja $\neg, \vee, \wedge, \rightarrow$ ja $\leftrightarrow$ kutsutaan *konnektiiveiksi*, koska niiden avulla kytketään yksinkertaisempia lausekkeita (lauseita) monimutkaisemmiksi.

1.2 Kielen märitelmä

Olkoon $\mathcal{P}$ ei-tyhjä joukko atomisia lauseita.

Lauselogiikan lauseenmuodostussäännöt:

1. Jokainen atominen lause $A \in \mathcal{P}$ on *lause*.
2. Jos α ja β ovat lauseita, niin myös $(\neg\alpha)$, $(\alpha \vee \beta)$, $(\alpha \wedge \beta)$, $(\alpha \rightarrow \beta)$, $(\alpha \leftrightarrow \beta)$ ovat *lauseita*.
3. vain edellä olevien sääntöjen perusteella muodostetut merkkijonot ovat *lauseita*.

Näiden sääntöjen nojalla muodostettavissa olevien lauseiden joukkoa kutsutaan (atomisten lauseiden joukkoon $\mathcal{P}$ perustuvaksi) lauselogiikan kieleksi $\mathcal{L}$.

Vaihtoehtoinen määritelmä

Olkoon $\mathcal{P}$ ei-tyhjä joukko atomisia lauseita.

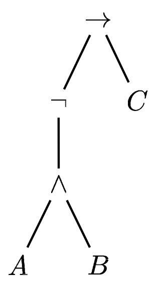
Määritelmä. Atomisten lauseiden joukkoon $\mathcal{P}$ perustuva *lauselogiikan kieli* $\mathcal{L}$ on merkkijonojen joukon $(\mathcal{P} \cup \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow, (,)\})^*$ *pienin* osajoukko, joka on suljettu seuraavien vaatimusten suhteen:

1. Jos $A \in \mathcal{P}$, niin $A \in \mathcal{L}$.
2. Jos $\alpha \in \mathcal{L}$ ja $\beta \in \mathcal{L}$, niin $(\neg\alpha) \in \mathcal{L}$, $(\alpha \vee \beta) \in \mathcal{L}$, $(\alpha \wedge \beta) \in \mathcal{L}$, $(\alpha \rightarrow \beta) \in \mathcal{L}$ ja $(\alpha \leftrightarrow \beta) \in \mathcal{L}$.

Esimerkki. Jos $\mathcal{P} = \{A, B\}$, niin esimerkiksi $A, B, (\neg A), ((\neg A) \vee B)$ ja $(((\neg A) \vee B) \rightarrow A)$ ovat kielen $\mathcal{L}$ lauseita. Sen sijaan merkkijonot $(\neg())$ ja $(A \vee C)$ eivät ole $\mathcal{L}$:n lauseita.

Jokaisella lauselogiikan lauseella on yksikäsitteinen *jäsennyspuu*.

Esimerkki. Lauseen $((\neg(A \wedge B)) \rightarrow C)$ jäsennyspuu on seuraava:



Jäsennyspuun juuressa oleva konnektiivi $\rightarrow$ määrää, että annettu lause on *muodoltaan* implikaatio (tai *implikaatio* lyhyesti sanottuna).

Vastaavasti määritellään lauseet, jotka ovat *negaatioita*, *konjunktioita*, *disjunktioita* ja *ekvivalensseja*.

Lauseiden ominaisuuksia voidaan todistaa induktiolla yli lauseen (tai sitä vastaavan jäsennyspuun) rakenteen.

1.3 Lauseiden muodostaminen

Jos lähtökohtana on joukko luonnollisen kielen lauseita,

- tunnistetaan atomiset lauseet eli väittämät, joita ei voida enää loogisessa mielessä pilkkoa osiin ja
- tunnistetaan konnektiivit ja muodostetaan vastaavat logiikan lauseet.

Tavoitteena voi olla myös jonkin järjestelmän määrittely suoraan logiikan lausein. Tällöin

- valitaan sopiva joukko järjestelmän ominaisuuksia kuvaavia atomisia lauseita ja
- määritellään näiden väliset suhteet/riippuvuudet logiikan lausein.

Esimerkki. Muotoillaan seuraava luonnollisen kielen lause lauselogiikan lauseena.

Jos tiedosto on liian suuri, niin se tiivistetään tai poistetaan.

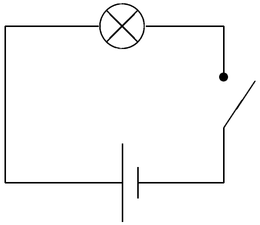
Valitaan atomiset lauseet

A = "Tiedosto on liian suuri",
B = "Tiedosto tiivistetään" ja
C = "Tiedosto poistetaan".

Saadaan: jos A, niin B tai C.

Tunnistetaan konnektiivit: $(A \rightarrow (B \vee C))$.

Esimerkki. Kuvataan seuraavaa yksinkertaista järjestelmää lauselogiikan lausein.



Valitaan atomisiksi lauseiksi

L = "Lamppu palaa",
K = "Kytkin on suljettu" ja
P = "Paristossa on riittävästi varausta".

Määritellään järjestelmän sallitut tilat lauseilla

$$((\neg P) \rightarrow (\neg L)) \text{ ja}$$
$$(P \rightarrow (L \leftrightarrow K)).$$

Tulkitse nämä kaksi lausetta luonnolliselle kielelle!

1.4 Sopimukset sulkeiden käytöstä

- Uloimmat sulkeet tapana jättää pois: $A \rightarrow B$ eikä $(A \rightarrow B)$.
- Konnektiivien presedenssi:
  1. $\neg$ on vahvin konnektiiveista.
  2. $\vee$ ja $\wedge$ ovat heikompia kuin $\neg$, mutta vahvempia kuin $\rightarrow$ ja $\leftrightarrow$.
  3. $\rightarrow$ ja $\leftrightarrow$ ovat heikoimmat konnektiivit.

     Esimerkiksi: $\neg A \rightarrow B$ eikä $(\neg A) \rightarrow B$,
     $A \wedge B \rightarrow B \vee C$ eikä $(A \wedge B) \rightarrow (B \vee C)$,
     mutta $(A \rightarrow B) \vee (B \leftrightarrow C)$.
- Ketjudisjunktiot/konjunktiot: $A \vee B \vee C$ kirjoitetaan lauseiden $A \vee (B \vee C)$ ja $(A \vee B) \vee C$ sijaan.

Lisähuomioita sulkeiden käytöstä

- Edellä tehdyt sopimukset (ketjukonjuktioita ja -disjunktioita lukuunottamatta) takaavat, että lauseen ϕ jäsennyspuu säilyy yksikäsitteisenä sulkeita vähennettäessä.
- Esimerkiksi merkkijonoa $A \rightarrow B \rightarrow C$ ei pystytä jäsentämään lauseeksi (yksikäsitteisesti).

  Tarvitaan sulkeet: $A \rightarrow (B \rightarrow C)$ tai $(A \rightarrow B) \rightarrow C$.

  Näillä lauseilla on yksikäsitteiset jäsennyspuut.
- Ketjudisjunktiokin $A \vee B \vee C$ voidaan jäsentää kahdella tavalla: $(A \vee B) \vee C$ tai $A \vee (B \vee C)$.

  Jatkossa näille annetaan kuitenkin sama merkitys, joten on samantekevää kuinka jäsennys suoritetaan.

1.5 Esimerkki: rakenteinen induktio

Määritelmä. Lauseen *alilauseet* määräytyvät seuraavasti:

Atomisen lauseen A ainoa alilause on A.

Lauseen $(\neg\alpha)$ alilauseita ovat α:n alilauseet ja $(\neg\alpha)$.

Lauseen $(\alpha \wedge \beta)$ alilauseita ovat α:n ja β:n alilauseet ja $(\alpha \wedge \beta)$.

Lauseiden $(\alpha \vee \beta)$, $(\alpha \rightarrow \beta)$ ja $(\alpha \leftrightarrow \beta)$ alilauseet määritellään samaan tapaan kuin $(\alpha \wedge \beta)$:n.

Esimerkki. Lauseen $A \rightarrow B \vee C$ alilauseet ovat $A, B, C, B \vee C$ ja $A \rightarrow B \vee C$.

Väite. Lauseen ϕ alilauseita on korkeintaan niin monta kuin on ϕ:ssä esiintyvien atomisten lauseiden lukumäärän ja ϕ:n konnektiivien lukumäärän summa.

Todistetaan väite induktiolla lauserakenteen suhteen.

Otetaan lauseelle ϕ käyttöön seuraavat merkinnät:

- $\#\phi$: lauseen ϕ alilauseiden lukumäärä,
- $A\phi$: lauseessa ϕ esiintyvien atomisten lauseiden lukumäärä ja
- $K\phi$: lauseen ϕ konnektiivien lukumäärä.

Näillä merkinnöillä väite saadaan muotoon $\#\phi \leq A\phi + K\phi$.

Esimerkki. Väittämä pitää paikkansa ainakin lauseen $\phi = (B \rightarrow C) \rightarrow (B \rightarrow C)$ tapauksessa: $\#\phi = 4$, $A\phi = 2$ ja $K\phi = 3$.

Todistus.

Perustapaus: ϕ on atominen lause A.

Koska $\#\phi = 1$ määritelmän mukaan, $A\phi = 1$ ja $K\phi = 0$, väittämä pitää tässä tapauksessa paikkansa.

Induktioaskel: ϕ on muotoa $(\neg\alpha)$.

Määritelmän mukaisesti: $\#(\neg\alpha) = 1 + \#\alpha$.

Induktio-oletuksen perusteella: $\#\alpha \leq A\alpha + K\alpha$.

Täten
$$\begin{aligned} \#(\neg\alpha) &\leq 1 + A\alpha + K\alpha \\ &= A\alpha + (1 + K\alpha) \\ &= A(\neg\alpha) + (1 + K\alpha) \\ &= A(\neg\alpha) + K(\neg\alpha). \end{aligned}$$

Näin väittämä tuli todistetuksi muotoa $(\neg\alpha)$ oleville lauseille.

Induktioaskel jatkuu: ϕ on muotoa $(\alpha * \beta)$, missä $*$ on mikä tahansa binäärisistä konnektiiveista $\vee$, $\wedge$, $\rightarrow$ ja $\leftrightarrow$.

$$\begin{aligned} \#(\alpha * \beta) &= 1 + \#\alpha + \#\beta - \#(\alpha, \beta) \\ &\leq 1 + A\alpha + K\alpha + A\beta + K\beta - \#(\alpha, \beta) \quad \text{(ind.-oletus)} \\ &= A\alpha + A\beta - \#(\alpha, \beta) + 1 + K\alpha + K\beta \\ &\leq A\alpha + A\beta - A(\alpha, \beta) + K(\alpha * \beta) \\ &= A(\alpha * \beta) + K(\alpha * \beta). \end{aligned}$$

Merkintä $\#(\alpha, \beta)$ tarkoittaa lauseiden α ja β yhteisten alilauseiden lukumäärää ja $A(\alpha, \beta)$ lauseiden α ja β yhteisten atomisten lauseiden lukumäärää. Tällöin pätee $A(\alpha, \beta) \leq \#(\alpha, \beta)$.

Näin ollen väittämä tuli todistetuksi kaikille lauseille ϕ.

2 Lauselogiikan semantiikka

- Perustotuustaulukot
- Konnektiivien määriteltävyys/riittävyys
- Lauselogiikan totuusmääritelmä
- Vaihtoehtoinen totuusmääritelmä: valuaatiot

2.1 Perustotuustaulukot

Perustotuustaulukoilla määritellään eri muotoa olevien lauseiden totuusarvot alilauseidensa funktiona.

α	$(\neg\alpha)$
T	E
E	T

α	β	$(\alpha \wedge \beta)$
T	T	T
T	E	E
E	T	E
E	E	E

α	β	$(\alpha \vee \beta)$
T	T	T
T	E	T
E	T	T
E	E	E

Totuustaulukot on helppo sisäistää muistisääntöjen avulla:
esim. $(\alpha \wedge \beta)$ on tosi $\iff$ α on tosi **ja** β on tosi.

Huomio. Disjunktio $(\alpha \vee \beta)$ on tosi $\iff$ α on tosi **tai** β on tosi.
Disjunktio $(\alpha \vee \beta)$ on siis tosi myös siloin, kun sekä α että β ovat tosia.

Voidaan määritellä myös poissulkeva disjuktio $(\alpha \underline{\vee} \beta)$, joka on epätosi molempien disjunktien ollessa tosia.

α	β	$(\alpha \underline{\vee} \beta)$
T	T	E
T	E	T
E	T	T
E	E	E

$(\alpha \underline{\vee} \beta)$ on tosi $\iff$ **joko** α on tosi **tai** β on tosi.

Perustotuustaulukot (jatkoa)

α	β	$(\alpha \rightarrow \beta)$
T	T	T
T	E	E
E	T	T
E	E	T

α	β	$(\alpha \leftrightarrow \beta)$
T	T	T
T	E	E
E	T	E
E	E	T

$(\alpha \rightarrow \beta)$ on tosi $\iff$ α on epätosi **tai** β on tosi
$\iff$ **jos** α on tosi, **nin** β on tosi.

Huomio. Implikaatio $\rightarrow$ *ei* edellytä syy-seuraus-suhdetta. Esim.

$A =$ "Ruotsi sijaitsee Aasiassa", $B =$ "Joulupukki on olemassa".

Lause $(A \rightarrow B)$ on tosi, koska A on epätosi.

Perustotuustaulukoiden avulla voidaan muodostaa totuustaulukko mille tahansa lauseelle (tai jopa lausejoukolle).

Esimerkki. Tutkittava lause: $(\neg B \wedge (A \rightarrow B))$
Alilauseet: $A, B, \neg B, (A \rightarrow B), (\neg B \wedge (A \rightarrow B))$
Totuustaulukossa on $2^2 = 4$ riviä ja 5 saraketta.

A	B	$\neg B$	$(A \rightarrow B)$	$(\neg B \wedge (A \rightarrow B))$
T	T	E	T	E
T	E	T	E	E
E	T	E	T	E
E	E	T	T	T

Muistathan, että jokaisessa lauseessa ϕ on korkeintaan niin monta alilausetta kuin lauseessa ϕ on atomisia lauseita ja konnektiiveja!

2.2 Konnektiivien määriteltävyys/riittävyys

Konnektiiveille voidaan antaa määritelmiä myös toistensa avulla:

- $(\alpha \wedge \beta)$ lauseena $\neg(\neg\alpha \vee \neg\beta)$
- $(\alpha \vee \beta)$ lauseena $\neg(\neg\alpha \wedge \neg\beta)$
- $(\alpha \veebar \beta)$ lauseena $(\alpha \leftrightarrow \neg\beta)$
- $(\alpha \rightarrow \beta)$ lauseena $(\neg\alpha \vee \beta)$ tai lauseena $\neg(\alpha \wedge \neg\beta)$
- $(\alpha \leftrightarrow \beta)$ lauseena $((\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha))$ tai lauseena $((\neg\alpha \vee \beta) \wedge (\alpha \vee \neg\beta))$

Totuustaulukoissa näille muodostuu identtiset pystyrivit.

Täten konnektiivit $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$ eivät ole kaikki välttämättömiä:

- $\neg$ ja $\vee$ riittävät muiden konnektiivien määrittelemiseen,
- $\neg$ ja $\wedge$ riittävät myös,
- $\neg$ ja $\rightarrow$ riittävät myös ja
- $\perp$ (aina epätosi lause) ja $\rightarrow$ riittävät myös.

Huomio. Toistaiseksi käsittelyt konnektiivit eivät ole ainoat mahdolliset. Esimerkiksi binäärikonnektiiveja $*$, jotka kytkevät kaksi lausetta α ja β lauseeksi $\alpha * \beta$, voidaan olennaisesti määritellä $2^{(2\times 2)} = 16$ erilaista.

On myös konnektiiveja, jotka riittävät yksinään muiden määrittelemiseen:

- Peircen nuoli: $(\alpha \downarrow \beta) \equiv \neg(\alpha \vee \beta)$
- Shefferin viiva: $(\alpha \mid \beta) \equiv \neg(\alpha \wedge \beta)$

2.3 Lauselogiikan totuusmääritelmä

Määritelmä. *Totuusjakelu* $\mathcal{A}$ on atomisten lauseiden joukon $\mathcal{P}$ osajoukko.

Ajatuksena on, että

- $\mathcal{A}$:n atomiset lauseet ovat tosia,
- $\mathcal{P} - \mathcal{A}$:n atomiset lauseet ovat epätosia.

Huomioita.

- Jos $\mathcal{P}$ on äärellinen, erilaisia totuusjakeluja on $2^{|\mathcal{P}|}$ kappaletta.
- Jokainen totuusjakelu vastaa yhtä totuustaulukon riviä (ja kääntäen).

Totuusjakelu voidaan ymmärtää yhden **asiaintilan** kuvauksena.

Esimerkki. (vrt. aikaisempi lamppuesimerkki)

$\mathcal{A}_1 = \{P\}$:	Patterissa on riittävästi varausta.
	Kytkin ei ole suljettu.
	Lamppu ei pala.
$\mathcal{A}_2 = \{L, K\}$:	Patterissa ei ole riittävästi varausta.
	Lamppu palaa.
	Kytkin on suljettu.

Näistä jälkimmäinen on mitä ilmeisimmin fyysisesti mahdoton, mutta kuitenkin loogisesti mahdollinen asiaintila.

Määritelmä. Seuraavassa määritellään milloin mielivaltainen lause $\phi \in \mathcal{L}$ on **tosi** totuusjakelussa $\mathcal{A}$ (merk. $\mathcal{A} \models \phi$) ja milloin ϕ on **epätosi** totuusjakelussa $\mathcal{A}$ (merk. $\mathcal{A} \not\models \phi$).

1. $\mathcal{A} \models A \iff A \in \mathcal{A}$ (atomisille lauseille $A \in \mathcal{P}$).
2. $\mathcal{A} \models \neg\alpha \iff \mathcal{A} \not\models \alpha$.
3. $\mathcal{A} \models \alpha \land \beta \iff \mathcal{A} \models \alpha$ ja $\mathcal{A} \models \beta$.
4. $\mathcal{A} \models \alpha \lor \beta \iff \mathcal{A} \models \alpha$ tai $\mathcal{A} \models \beta$.
5. $\mathcal{A} \models \alpha \rightarrow \beta \iff \mathcal{A} \not\models \alpha$ tai $\mathcal{A} \models \beta$.
6. $\mathcal{A} \models \alpha \leftrightarrow \beta \iff$ joko $\mathcal{A} \models \alpha$ ja $\mathcal{A} \models \beta$, tai $\mathcal{A} \not\models \alpha$ ja $\mathcal{A} \not\models \beta$.

Huomio. Kohdan 3 perusteella $\mathcal{A} \not\models \alpha \land \beta \iff \mathcal{A} \not\models \alpha$ tai $\mathcal{A} \not\models \beta$.

Totuusmääritelmän seurauksia

Väite. Jos $\mathcal{A} \subseteq \mathcal{P}$ on totuusjakelu, niin kaikille lauseille $\phi \in \mathcal{L}$, joko $\mathcal{A} \models \phi$ tai $\mathcal{A} \not\models \phi$.

Merkitään At(ϕ):llä ϕ:ssä esiintyvien atomisten lauseiden joukkoa.

Väite. Olkoon $\mathcal{A}_1 \subseteq \mathcal{P}$ ja $\mathcal{A}_2 \subseteq \mathcal{P}$ kaksi totuusjakelua ja $\phi \in \mathcal{L}$ lause. Jos $\mathcal{A}_1 \cap \mathrm{At}(\phi) = \mathcal{A}_2 \cap \mathrm{At}(\phi)$, niin $\mathcal{A}_1 \models \phi \iff \mathcal{A}_2 \models \phi$.

Nämä voidaan todistaa induktiolla ϕ:n rakenteen suhteen.

2.4 Vaihtoehtoinen totuusmäritelmä: valuaatiot

(Tätä määritelmää käytetään Neroden ja Shoren kirjassa).

Määritelmä. *Valuaatio* $\mathcal{V}$ on *konnektiivien totuustaulukkoja noudattava* funktio kielen $\mathcal{L}$ *lauseiden joukolta* joukolle $\{T, E\}$.

Valuaatioilla ja totuusjakeluilla on seuraava yhteys:

1. Olkoon $\mathcal{V}$ valuaatio ja $\mathcal{A} = \{A \in \mathcal{P} \mid \mathcal{V}(A) = T\}$.
   Tällöin kaikille lauseille $\phi \in \mathcal{L}$ pätee:
   $\mathcal{A} \models \phi \iff \mathcal{V}(\phi) = T$.
2. Jos $\mathcal{A}$ on totuusjakelu, niin on olemassa yksikäsitteinen valuaatio $\mathcal{V}$ siten että $\mathcal{A} = \{A \in \mathcal{P} \mid \mathcal{V}(A) = T\}$.

3 Semanttiset peruskäsitteet

- Mallin käsite
- Toteutuvuus
- Pätevyys
- Looginen seuraavuus
- Looginen ekvivalenssi
- Peruskäsitteiden väliset yhteydet
- Loogisten seurauksien ominaisuuksia

3.1 Mallin käsite

Määritelmä. Totuusjakelu $\mathcal{A} \subseteq \mathcal{P}$ on lauseen $\alpha \in \mathcal{L}$ *malli*, joss lause α on tosi $\mathcal{A}$:ssa eli $\mathcal{A} \models \alpha$.

Esimerkki. Totuusjakelut $\mathcal{A}_1 = \{A\}, \mathcal{A}_2 = \{B\}$ ja $\mathcal{A}_3 = \{A, B\}$ ovat malleja lauseelle $A \lor B$.

Määritelmä. Totuusjakelu $\mathcal{A} \subseteq \mathcal{P}$ on lausejoukon $\Sigma \subseteq \mathcal{L}$ *malli* (merk. $\mathcal{A} \models \Sigma$), joss kaikille lausejoukon Σ lauseille $\sigma \in \Sigma$ pätee $\mathcal{A} \models \sigma$.

Näin ollen lausejoukon Σ lauseet tulkitaan konjunktiivisesti.

Olkoon $\mathcal{P} = \{P, L, K\}$. Lamppuesimerkin lausejoukko on $\Sigma = \{\neg P \to \neg L, P \to (L \leftrightarrow K)\} \subseteq \mathcal{L}$. Totuustaulukko:

P	L	K	$\neg P$	$\neg L$	$\neg P \to \neg L$	$L \leftrightarrow K$	$P \to (L \leftrightarrow K)$	
T	T	T	E	E	$\boxed{T}$	T	$\boxed{T}$	$\longleftarrow$
T	T	E	E	E	T	E	E	
T	E	T	E	T	T	E	E	
T	E	E	E	T	$\boxed{T}$	T	$\boxed{T}$	$\longleftarrow$
E	T	T	T	E	E	T	T	
E	T	E	T	E	E	E	T	
E	E	T	T	T	$\boxed{T}$	E	$\boxed{T}$	$\longleftarrow$
E	E	E	T	T	$\boxed{T}$	T	$\boxed{T}$	$\longleftarrow$

Lausejoukolla Σ on siis neljä erilaista mallia (merkitty nuolin), jotka vastaavat järjestelmän sallittuja tiloja.

3.2 Toteutuvuus

Määritelmä. Lause $\alpha \in \mathcal{L}$ (tai lausejoukko $\Sigma \subseteq \mathcal{L}$) on *toteutuva*, joss ainakin yksi totuusjakelu $\mathcal{A} \subseteq \mathcal{P}$ on sen malli.

Huomio. Koska lauseiden totuusarvot määräytyvät niissä esiintyvien atomisten lauseiden totuusarvoista, voimme rajoittua totuusjakeluihin $\mathcal{A} \subseteq \mathrm{At}(\phi)$, missä $\mathrm{At}(\phi)$ on lauseessa ϕ esiintyvien atomisten lauseiden joukko, tai totuusjakeluihin $\mathcal{A} \subseteq \mathrm{At}(\Sigma) = \cup\{\mathrm{At}(\sigma) | \sigma \in \Sigma\}$ selvittäessämme ϕ:n tai Σ:n malleja.

- *Toteutumattomalla* lauseella/lausejoukolla ei ole yhtään mallia.
- Lauseen toteutuvuuden selvittäminen on yksi keskeisimmistä logiikkaan liittyvistä laskennallisista ongelmista.

Toteutuvuuden tutkiminen totuustaulukolla:

- muodostetaan α:lle (Σ:lle) totuustaulukko ja
- tarkastetaan onko α tosi (kaikki Σ:n lauseet tosia) jollakin rivillä.

Esimerkki.

Onko $A \land \neg A$ toteutuva?

A	$\neg A$	$A \land \neg A$
T	E	E
E	T	E

Ei.

Onko $A \lor \neg B$ toteutuva?

A	B	$\neg B$	$A \lor \neg B$
T	T	E	$\boxed{T}$
T	E	T	$\boxed{T}$
E	T	E	E
E	E	T	$\boxed{T}$

Kyllä.

3.3 Pätevyys

Määritelmä. Lause $\alpha \in \mathcal{L}$ on *pätevä/tautologia* (merkitään $\models \alpha$), jos $\mathcal{A} \models \alpha$ kaikille totuusjakeluille $\mathcal{A} \subseteq \mathcal{P}$.

Esimerkki. Olkoon $\mathcal{P} = \{A\}$ ja $\mathcal{L}$ vastaava kieli. Lause $A \lor \neg A$ on pätevä, koska $A \lor \neg A$ on tosi totuusjakeluissa $\mathcal{A}_1 = \{\}$ ja $\mathcal{A}_2 = \{A\}$.

Vastamallit

Tarkastellaan mielivaltaista lausetta $\phi \in \mathcal{L}$.

- Jos ϕ on pätevä, $\mathcal{A} \models \phi$ kaikille totuusjakeluille $\mathcal{A} \subseteq \mathcal{P}$.
- Jos ϕ ei ole pätevä, löytyy totuusjakelu $\mathcal{A}$ siten, että $\mathcal{A} \not\models \phi$.

Jälkimmäisessä tapauksessa kutsumme totuusjakelua $\mathcal{A}$ *vastamalliksi* (tai vastaesimerkiksi) lauseen ϕ pätevyydelle.

Pätevyyden tutkiminen totuustaulukolla:

- muodostetaan α:lle totuustaulukko ja
- tarkistetaan, että α on tosi jokaisella rivillä.

Esimerkki.

Onko $A \land B \to A$ pätevä?

A	B	$A \land B$	$A \land B \to A$
T	T	T	T
T	E	E	T
E	T	E	T
E	E	E	T

Kyllä.

Onko $A \lor B \to A$ pätevä?

A	B	$A \lor B$	$A \lor B \to A$
T	T	T	T
T	E	T	T
E	T	T	$\boxed{E}$
E	E	E	T

Ei.

3.4 Looginen seuraavuus

Määritelmä. Lause $\alpha \in \mathcal{L}$ on lausejoukon $\Sigma \subseteq \mathcal{L}$ looginen seuraus (merkitään $\Sigma \models \alpha$), jos $\mathcal{A} \models \alpha$ kaikille lausejoukon Σ malleille $\mathcal{A} \subseteq \mathcal{P}$.

Esimerkki. $\neg\neg A$ seuraa loogisesti lausejoukoista $\{A\}$ ja $\{A \land \neg A\}$.

Käytämme *vastamallin* käsitettä myös loogisen seuraavuuden yhteydessä.

- Jos $\Sigma \not\models \phi$, lausejoukolla Σ on malli $\mathcal{A}$ siten, että $\mathcal{A} \not\models \phi$.

Esimerkki. $\{A, B \to A\} \not\models \neg\neg B$, koska löytyy vastamalli $\mathcal{A} = \{A\}$ siten, että $\mathcal{A} \models \{A, B \to A\}$ ja $\mathcal{A} \not\models \neg\neg B$.

Loogisen seuraavuuden tutkiminen totuustaulukolla:

- muodostetaan lausejoukolle $\Sigma \cup \{\alpha\}$ totuustaulukko,
- todetaan rivit, joilla kaikki Σ:n lauseet ovat tosia (Σ:n mallit) ja
- tarkistetaan, että α on näillä riveillä tosi.

Esimerkki.

$\{A, (A \to D)\} \models D$?

A	D	$(A \to D)$	
T	T	T	$\longleftarrow$
T	E	E	
E	T	T	
E	E	T	

Kyllä.

$\{(A \to B)\} \models B$?

A	B	$(A \to B)$	
T	T	T	$\longleftarrow$
T	E	E	
E	T	T	$\longleftarrow$
E	$\boxed{E}$	T	$\longleftarrow$

Ei.

Huomaa, että $\Sigma \models A \vee \neg A$ kaikille lausejoukoille Σ !!!

Esimerkki. Tutkitaan, onko lause $\neg L \vee K$ looginen seuraus lamppuesimerkin lausejoukolle $\Sigma = \{P \to (L \leftrightarrow K), \neg P \to \neg L\}$. On.

P	L	K	$\neg P$	$\neg L$	$\neg P \to \neg L$	$L \leftrightarrow K$	$P \to (L \leftrightarrow K)$	$\neg L \vee K$	
T	T	T	E	E	T	T	T	$\boxed{T}$	$\longleftarrow$
T	T	E	E	E	T	E	E	E	
T	E	T	E	T	T	E	E	T	
T	E	E	E	T	T	T	T	$\boxed{T}$	$\longleftarrow$
E	T	T	T	E	E	T	T	T	
E	T	E	T	E	E	E	T	E	
E	E	T	T	T	T	E	T	$\boxed{T}$	$\longleftarrow$
E	E	E	T	T	T	T	T	$\boxed{T}$	$\longleftarrow$

3.5 Looginen ekvivalenssi

Määritelmä. Lauseet $\alpha \in \mathcal{L}$ ja $\beta \in \mathcal{L}$ ovat *loogisesti ekvivalentteja* ($\alpha \equiv \beta$), joss $\mathcal{A} \models \alpha \iff \mathcal{A} \models \beta$ kaikille totuusjakeluille $\mathcal{A} \subseteq \mathcal{P}$.

Esimerkki. A ja $\neg\neg A$ ovat loogisesti ekvivalentit, koska näillä on sama totuusarvo kaikissa totuusjakeluissa.

Väite. Lauseet $\alpha \in \mathcal{L}$ ja $\beta \in \mathcal{L}$ ovat loogisesti ekvivalentit, joss niillä on täsmälleen samat mallit $\mathcal{A} \subseteq \mathcal{P}$.

Lauseiden loogisen ekvivalenssin selvittäminen:

- muodostetaan lauseille α ja β yhteinen totuustaulukko ja
- tarkastetaan, että α:lla ja β:lla on sama totuusarvo jokaisella rivillä.

Esimerkki. Ovatko lauseet $A \to B$ ja $\neg B \to \neg A$ loogisesti ekvivalentit?

$\neg A$	$\neg B$	$A \to B$	$\neg B \to \neg A$
E	E	T	T
E	T	E	E
T	E	T	T
T	T	T	T

Kyllä.

Määritelmä. Lausejoukot $\Sigma_1 \subseteq \mathcal{L}$ ja $\Sigma_2 \subseteq \mathcal{L}$ ovat loogisesti ekvivalentit, joss kaikille totuusjakeluille $\mathcal{A} \subseteq \mathcal{P}$ pätee seuraava:

$$\mathcal{A} \models \sigma_1 \text{ kaikille } \sigma_1 \in \Sigma_1$$
$$\iff \mathcal{A} \models \sigma_2 \text{ kaikille } \sigma_2 \in \Sigma_2.$$

Väite. Lausejoukot $\Sigma_1 \subseteq \mathcal{L}$ ja $\Sigma_2 \subseteq \mathcal{L}$ ovat loogisesti ekvivalentit, mikäli niillä on täsmälleen samat mallit $\mathcal{A} \subseteq \mathcal{P}$.

Lausejoukkojen loogisen ekvivalenssin selvittäminen:

- todetaan, että $\Sigma_1 \models \sigma_2$ kaikille lauseille $\sigma_2 \in \Sigma_2$ ja
- todetaan, että $\Sigma_2 \models \sigma_1$ kaikille lauseille $\sigma_1 \in \Sigma_1$.

Yllä Σ_1 ja Σ_2 ovat siis kaksi lausejoukkoa.

3.6 Peruskäsitteiden väliset yhteydet

Olkoon $\mathcal{L}$ atomisten lauseiden joukkoon $\mathcal{P}$ perustuva kieli.

Tarkastellaan jatkossa tämän kielen lauseita.

Looginen ekvivalenssi liittyy läheisesti pätevyyteen:

- $\alpha \equiv \beta \iff \models \alpha \leftrightarrow \beta$.

  Todistus.

  $\alpha \not\equiv \beta$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A} \models \alpha$ ja $\mathcal{A} \not\models \beta$, tai $\mathcal{A} \not\models \alpha$ ja $\mathcal{A} \models \beta$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A} \not\models \alpha \leftrightarrow \beta$
  $\iff \not\models \alpha \leftrightarrow \beta$.

Pätevyydellä ja loogisella seuraavuudella on kiinteät yhteydet.

- $\models \alpha \iff \emptyset \models \alpha$.

  Seuraa helposti, koska kaikki totuusjakelut $\mathcal{A} \subseteq \mathcal{P}$ ovat tyhjän lausejoukon $\emptyset$ malleja

- $\{\phi_1, \ldots, \phi_n\} \models \phi \iff \models \phi_1 \wedge \cdots \wedge \phi_n \rightarrow \phi$.

  Todistus.

  $\{\phi_1, \ldots, \phi_n\} \not\models \phi$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A}$ on $\{\phi_1, \ldots, \phi_n\}$:n malli ja $\mathcal{A} \not\models \phi$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\forall i \in \{1, \ldots, n\} : \mathcal{A} \models \phi_i$ ja $\mathcal{A} \not\models \phi$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A} \models \phi_1 \wedge \cdots \wedge \phi_n$ ja $\mathcal{A} \not\models \phi$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A} \not\models \phi_1 \wedge \cdots \wedge \phi_n \rightarrow \phi$
  $\iff \not\models \phi_1 \wedge \cdots \wedge \phi_n \rightarrow \phi$.

Pätevyyden ja loogisen seuraavuuden yhteys toteutuvuuteen:

- $\models \alpha \iff \neg\alpha$ on toteutumaton.

  Todistus. Erikoistapaus seuraavasta.

- $\Sigma \models \alpha \iff \Sigma \cup \{\neg\alpha\}$ on toteutumaton.

  Todistus.

  $\Sigma \not\models \alpha$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A} \models \Sigma$ ja $\mathcal{A} \not\models \alpha$
  $\iff \exists \mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A} \models \Sigma \cup \{\neg\alpha\}$
  $\iff \Sigma \cup \{\neg\alpha\}$ on toteutuva
  $\iff \Sigma \cup \{\neg\alpha\}$ ei ole toteutumaton.

Edellä esitetyt yhteydet mahdollistavat lauselogiikan päättelyongelmien väliset muunnokset.

3.7 Loogisten seurauksien ominaisuuksia

Väite. (Kompaktius). Jos $\Sigma \models \phi$, niin on olemassa äärellinen osajoukko $\Sigma' \subseteq \Sigma$ siten, että $\Sigma' \models \phi$.

Todistus sivuutetaan.

Määritelmä. Lausejoukon $\Sigma \subseteq \mathcal{L}$ loogisten seurausten joukko on $\mathsf{Cn}(\Sigma) = \{\phi \in \mathcal{L} \mid \Sigma \models \phi\}$.

Loogisten seurauksien joukolla $\mathsf{Cn}(\Sigma)$ on seuraavat perusominaisuudet:

- $\Sigma \subseteq \mathsf{Cn}(\Sigma)$.
- Monotonisuus: $\Sigma_1 \subseteq \Sigma_2 \Longrightarrow \mathsf{Cn}(\Sigma_1) \subseteq \mathsf{Cn}(\Sigma_2)$.
- $\mathsf{Cn}(\mathsf{Cn}(\Sigma)) = \mathsf{Cn}(\Sigma)$.

Tietämyksen esittämisen problematiikkaa

- Jokainen lausejoukko Σ määrittää joukon *malleja*, eli totuusjakeluja $\mathcal{A}$, joissa lausejoukon kaikki lauseet ovat tosia.

  Esimerkki. Lamppuesimerkin lausejoukolla

  $$\Sigma = \{\neg P \rightarrow \neg L, P \rightarrow (L \leftrightarrow K)\} \subseteq \mathcal{L}$$

  on mallit $\mathcal{A}_1 = \{P, L, K\}$, $\mathcal{A}_2 = \{P\}$, $\mathcal{A}_3 = \{K\}$ ja $\mathcal{A}_4 = \{\}$.
- Lausejoukon Σ mallit puolestaan määräävät lausejoukon loogisten seurauksien joukon $\mathsf{Cn}(\Sigma)$.

  Esimerkki. Lamppuesimerkissä lausejoukon Σ loogisten seurausten joukko on $\mathsf{Cn}(\Sigma) = \{\neg L \vee K, L \rightarrow P \wedge K, \neg P \vee P, \ldots\}$.

- Kaikki totuusjakelut ovat tyhjän lausejoukon $\emptyset$ malleja. Täten $\mathsf{Cn}(\emptyset)$ on itse asiassa pätevien lauseiden joukko.

  Huomio. Kaikille lausejoukoille Σ pätee $\mathsf{Cn}(\emptyset) \subseteq \mathsf{Cn}(\Sigma)$.
- **Monotonisuus:**

  lisää lauseita $\Longrightarrow$ vähemmän malleja $\Longrightarrow$ lisää loogisia seurauksia.

  Esimerkki. Lamppuesimerkissä $\Sigma \not\models L \leftrightarrow K$, mutta $\Sigma \cup \{P\} \models L \leftrightarrow K$.
- *Tietämyksen esittäminen*: ongelmana rajata mallien joukko sopivat lauseet valitsemalla siten, että saadaan halutut loogiset seuraukset.
- Jos lausejoukko tulee ristiriitaiseksi, sillä ei ole mallia ja kaikki lauseet ovat tällöin lausejoukon loogisia seurauksia.

  Esimerkki. Lamppuesimerkin tapauksessa lausejoukko $\Sigma \cup \{K \leftrightarrow \neg P,\ L\}$ on ristiriitainen/toteutumaton.

Lisähuomioita loogisesta seuraavuudesta

- Jos $\Sigma \models \phi$, niin $\mathsf{Cn}(\Sigma) = \mathsf{Cn}(\Sigma \cup \{\phi\})$.

  Jos tavoitteena on lausejoukon koon minimointi, lausejoukon loogisia seurauksia ei siis kannata lisätä lausejoukkoon, koska lausejoukon loogiset seuraavuudet (eikä myöskään mallit) eivät näin muutu!
- Oletetaan, että $\Sigma \not\models \phi$ ja että Σ halutaan laajentaa lausejoukoksi Σ' siten, että $\Sigma' \models \phi$.

  Apuna voidaan käyttää vastamallia/vastamalleja $\mathcal{A}$, joille $\mathcal{A} \models \Sigma$ ja $\mathcal{A} \not\models \phi$: lausejoukkoon Σ lisättävän lauseen ψ tulisi sulkea pois vastamalli/vastamalleja $\mathcal{A}$, ts. $\mathcal{A} \not\models \Sigma \cup \{\psi\}$.

  Lause ϕ toteuttaa myös tämän vaatimuksen, mutta lausetta ϕ ei välttämättä haluta liittää eksplisiittiseen tietämykseen.

  Esimerkki. $\Sigma = \{P \rightarrow Q\}$, $\phi = Q$ ja $\psi = P$.

4 Semanttinen taulu

- Konnektiivikohtaiset taulusäännöt
- Semanttisen taulun määritelmä
- Taulumenetelmän ominaisuuksia
- Loogisten ongelmien ratkominen taulumenetelmällä
- Esimerkkejä

4.1 Konnektiivikohtaiset taulusäännöt

- Totuustaulukkoja käytetään lauseen ϕ totuusarvon laskemiseen, kun annettuna on atomisten lauseiden $A \in \mathrm{At}(\phi)$ totuusarvot.
- Semanttisissa tauluissa idea on käänteinen: määrätään ϕ:n alilauseiden (ja lopulta atomisten lauseiden $A \in \mathrm{At}(\phi)$) totuusarvot lähtien liikkeelle lauseen α totuusarvosta (tosi tai epätosi).

Esimerkki. Verrataan konjunktion totuustaulukkoa ja taulusääntöjä:

α	β	$(\alpha \wedge \beta)$
T	T	T
T	E	E
E	T	E
E	E	E

$$\begin{array}{c} T(\alpha\wedge\beta) \\ | \\ T\alpha \\ | \\ T\beta \end{array} \qquad \begin{array}{c} E(\alpha\wedge\beta) \\ / \quad \backslash \\ E\alpha \quad E\beta \end{array}$$

Konnektiivien $\neg$, $\vee$, $\wedge$ ja $\rightarrow$ taulusäännöt ovat seuraavat:

$$\begin{array}{c} T\neg\alpha \\ | \\ E\alpha \end{array} \qquad \begin{array}{c} T(\alpha\vee\beta) \\ / \quad \backslash \\ T\alpha \quad T\beta \end{array} \qquad \begin{array}{c} T(\alpha\wedge\beta) \\ | \\ T\alpha \\ | \\ T\beta \end{array} \qquad \begin{array}{c} T(\alpha\rightarrow\beta) \\ / \quad \backslash \\ E\alpha \quad T\beta \end{array}$$

$$\begin{array}{c} E\neg\alpha \\ | \\ T\alpha \end{array} \qquad \begin{array}{c} E(\alpha\vee\beta) \\ | \\ E\alpha \\ | \\ E\beta \end{array} \qquad \begin{array}{c} E(\alpha\wedge\beta) \\ / \quad \backslash \\ E\alpha \quad E\beta \end{array} \qquad \begin{array}{c} E(\alpha\rightarrow\beta) \\ | \\ T\alpha \\ | \\ E\beta \end{array}$$

Konnektiivien $\underline{\vee}$ ja $\leftrightarrow$ taulusäännöt ovat seuraavat:

$$\begin{array}{c} T(\alpha\,\underline{\vee}\,\beta) \\ / \quad \backslash \\ T\alpha \quad E\alpha \\ | \qquad | \\ E\beta \quad T\beta \end{array} \qquad \begin{array}{c} T(\alpha\leftrightarrow\beta) \\ / \quad \backslash \\ T\alpha \quad E\alpha \\ | \qquad | \\ T\beta \quad E\beta \end{array}$$

$$\begin{array}{c} E(\alpha\,\underline{\vee}\,\beta) \\ / \quad \backslash \\ T\alpha \quad E\alpha \\ | \qquad | \\ T\beta \quad E\beta \end{array} \qquad \begin{array}{c} E(\alpha\leftrightarrow\beta) \\ / \quad \backslash \\ T\alpha \quad E\alpha \\ | \qquad | \\ E\beta \quad T\beta \end{array}$$

Huomio. Näistä taulusäännöistä voi todeta konnektiivien $\underline{\vee}$ ja $\leftrightarrow$ väliset suhteet: $\alpha\,\underline{\vee}\,\beta \equiv \alpha \leftrightarrow \neg\beta \equiv \neg\alpha \leftrightarrow \beta \equiv \neg(\alpha \leftrightarrow \beta)$.

Ketjukonjunktioille ja -disjunktioille voidaan johtaa seuraavat säännöt:

$$\begin{array}{c} T(\alpha_1 \land \ldots \land \alpha_n) \\ | \\ T\alpha_1 \\ \vdots \\ T\alpha_n \end{array} \qquad \begin{array}{ccc} & T(\alpha_1 \lor \ldots \lor \alpha_n) & \\ \swarrow & | & \searrow \\ T\alpha_1 & \ldots & T\alpha_n \end{array}$$

$$\begin{array}{ccc} & E(\alpha_1 \land \ldots \land \alpha_n) & \\ \swarrow & | & \searrow \\ E\alpha_1 & \ldots & E\alpha_n \end{array} \qquad \begin{array}{c} E(\alpha_1 \lor \ldots \lor \alpha_n) \\ | \\ E\alpha_1 \\ \vdots \\ E\alpha_n \end{array}$$

Taulusäännön instantiointi

Esimerkki. Tarkastellaan lausetta $(\neg A \to C \lor D) \to A \lor B$:

Lause on muodoltaan implikaatio $\alpha \to \beta$, missä ehtona on lause $\alpha = \neg A \to C \lor D$ ja seurauksena lause $\beta = A \lor B$.

Korvataan α ja β kyseisillä lauseilla implikaation säännössä:

$$\begin{array}{cc} \multicolumn{2}{c}{T(\alpha \to \beta)} \\ / & \backslash \\ E\alpha & T\beta \end{array} \quad \leadsto \quad \begin{array}{cc} \multicolumn{2}{c}{T((\neg A \to C \lor D) \to A \lor B)} \\ / & \backslash \\ E(\neg A \to C \lor D) & T(A \lor B) \end{array}$$

4.2 Semanttisen taulun märitelmä

Semanttinen taulu on rakenteeltaan puu, jonka

- solmujen asteluku on enintään kaksi ja
- solmuina on muotoa $T\phi$ ja $E\phi$ olevia lausekkeita.

Määritelmä. Semanttinen taulu muodostetaan seuraavilla periaatteilla:

- Jokainen yksisolmuinen puu, jonka ainoana solmuna on juurisolmu $T\phi$ tai $E\phi$, on semanttinen taulu.
- Olkoon τ semanttinen taulu, P jokin τ:n polku juurisolmusta lehtisolmuun, $T\phi$ $(E\phi)$ jokin τ:n solmu polulla P.

  Jos τ' saadaan τ:sta liittämällä $T\phi$:n $(E\phi$:n) taulusääntö (ilman juurisolmua) polun P jatkoksi, niin τ' on myöskin semanttinen taulu.

Esimerkki.

$$\begin{array}{cccc} \multicolumn{4}{c}{1.\ E(A \lor B \to A \land B)} \\ \multicolumn{4}{c}{|} \\ \multicolumn{4}{c}{2.\ T(A \lor B)^{1.}} \\ \multicolumn{4}{c}{|} \\ \multicolumn{4}{c}{3.\ E(A \land B)^{1.}} \\ \multicolumn{2}{c}{4.\ TA^{2.}} & \multicolumn{2}{c}{4.\ TB^{2.}} \\ 5.\ EA^{3.} & 5.\ EB^{3.} & 5.\ EA^{3.} & 5.\ EB^{3.} \end{array}$$

Semanttisen taulun solmut on numeroitu luettavuuden parantamiseksi. *Poluille lisättävät solmut numeroidaan juoksevasti.* Yläindeksinä oleva numero kertoo, monennestako polun solmusta kyseinen solmu on saatu taulusääntöä soveltamalla.

Määritelmä. Olkoon τ semanttinen taulu, P polku juurisolmusta lehtisolmuun taulussa τ ja $T\phi$ ($E\phi$) polun P solmu.

1. Solmu $T\phi$ ($E\phi$) on *hajoitettu* polulla P, jos
   (a) ϕ on atominen lause, tai
   (b) solmun $T\phi$ ($E\phi$) taulusäännön jonkun polun jokainen solmu on polulla P.
2. Polku P on *ristiriitainen*, jos sillä esiintyy sekä $T\alpha$ että $E\alpha$ jollekin lauseelle α (tämän merkiksi kirjoitetaan usein $\times$ polun loppuun).
3. Polku P on *valmis*, jos se on ristiriitainen tai jokainen sen solmuista on hajoitettu polulla P.
4. Taulu τ on *valmis*, jos jokainen sen poluista on valmis.
5. Taulu τ on *ristiriitainen*, jos jokainen sen poluista on ristiriitainen.

Esimerkki. Palataan edelliseen esimerkkiin.

1. $E(A \lor B \to A \land B)$
2. $T(A \lor B)^{1.}$
3. $E(A \land B)^{1.}$

$TA^{2.}$ $\qquad$ $TB^{2.}$

$EA^{3.}$ $\times$ $\qquad$ $EB^{3.}$ $\qquad$ $EA^{3.}$ $\qquad$ $EB^{3.}$ $\times$

- Taulun reunimmaiset polut ovat ristiriitaiset.
- Kaikki taulun polut ovat valmiita.
- Taulu on valmis, muttei ristiriitainen.

4.3 Taulumenetelmän ominaisuuksia

Olkoon α atomisten lauseiden joukkoon $\mathcal{P}$ perustuvan kielen $\mathcal{L}$ lause.

Semanttisella taululla voidaan ratkaista seuraava looginen ongelma:

onko olemassa totuusjakelua $\mathcal{A} \subseteq \mathcal{P}$ siten, että $\mathcal{A} \models \alpha$ ($\mathcal{A} \not\models \alpha$)?

Todistamme jatkossa seuraavan ominaisuuden:

Väite. Olkoon $\alpha \in \mathcal{L}$ mikä tahansa lause. Juurisolmusta $T\alpha$ ($E\alpha$) muodostetussa valmiissa semanttisessa taulussa on ei-ristiriitainen polku, jos ja vain jos $\mathcal{A} \models \alpha$ ($\mathcal{A} \not\models \alpha$) jollekin totuusjakelulle $\mathcal{A} \subseteq \mathcal{P}$.

Taulumenetelmän virheettömyys

Olkoon $\alpha \in \mathcal{L}$ mikä tahansa lause ja τ juurisolmusta $T\alpha$ ($E\alpha$) muodostettu valmis semanttinen taulu.

Jos semanttisessa taulussa τ on ei-ristiriitainen polku,
nin $\mathcal{A} \models \alpha$ ($\mathcal{A} \not\models \alpha$) jollekin totuusjakelulle $\mathcal{A} \subseteq \mathcal{P}$.

Todistus. Olkoon P mikä tahansa taulun τ ei-ristiriitainen polku.

Polku P on valmis, koska taulu τ on valmis.

Olkoon totuusjakelu $\mathcal{A} = \{A \in \mathcal{P} \mid \text{solmu } TA \text{ esiintyy polulla } P\}$.

Todistetaan induktiolla polun P solmujen rakenteen suhteen, että

- $\mathcal{A} \models \beta$ kaikille $T\beta \in P$ ja
- $\mathcal{A} \not\models \beta$ kaikille $E\beta \in P$.

Perustapaus: $TA \in P$ ($EB \in P$), missä A (B) on $\mathcal{P}$:n atominen lause. Totuusjakelun $\mathcal{A}$ määritelmän perusteella $\mathcal{A} \models A$ ($\mathcal{A} \not\models B$).

Induktioaskel: Käydään läpi kaikki tapaukset
$T(\alpha \wedge \beta), E(\alpha \wedge \beta), T(\alpha \vee \beta), E(\alpha \vee \beta), T(\alpha \rightarrow \beta), E(\alpha \rightarrow \beta), \ldots$

- $E(\alpha \rightarrow \beta) \in P$, joloin $T\alpha \in P$ ja $E\beta \in P$ (koska P on valmis):
  Induktio-oletuksella $\mathcal{A} \models \alpha$ ja $\mathcal{A} \not\models \beta$.
  Totuusmääritelmän perusteella tällöin $\mathcal{A} \not\models \alpha \rightarrow \beta$.
- $T(\alpha \rightarrow \beta) \in P$, jolloin $E\alpha \in P$ tai $T\beta \in P$ (koska P on valmis):
  Induktio-oletuksella $\mathcal{A} \not\models \alpha$ jos $E\alpha \in P$, tai $\mathcal{A} \models \beta$ jos $T\beta \in P$.
  Totuusmääritelmän perusteella tällöin $\mathcal{A} \models \alpha \rightarrow \beta$.
- Muut tapaukset käsitellään samaan tapaan.

Tarvitsemme täydellisyystodistuksessa seuraavaa apukäsitettä:

Määritelmä. Semanttisen taulun τ polku P on *yhteensopiva* totuusjakelun $\mathcal{A}$ kanssa (merkitään $\mathcal{A} \| P$), joss $\mathcal{A} \models \beta$ jokaiselle polun P solmulle $T\beta$, ja $\mathcal{A} \not\models \beta$ jokaiselle polun P solmulle $E\beta$.

Taulumenetelmän täydellisyys

Olkoon $\alpha \in \mathcal{L}$ mikä tahansa lause ja τ juurisolmusta $T\alpha$ ($E\alpha$) muodostettu (ei välttämättä valmis) semanttinen taulu.

Jos $\mathcal{A} \models \alpha$ ($\mathcal{A} \not\models \alpha$) jossain totuusjakelussa $\mathcal{A} \subseteq \mathcal{P}$, **nin** semanttisessa taulussa τ on ei-ristiriitainen polku P siten, että $\mathcal{A} \| P$.

Todistus. Oletetaan $\mathcal{A} \models \alpha$ ($\mathcal{A} \not\models \alpha$) jollekin totuusjakelulle $\mathcal{A} \subseteq \mathcal{P}$. Käytetään induktiota semanttisen taulun τ rakenteen suhteen (vrt. semanttisen taulun induktiivinen määritelmä).

Perustapaus: taulu τ koostuu ainoastaan juurisolmusta $T\alpha$ ($E\alpha$). Tällöin ainoa mahdollinen polku P juurisolmusta $T\alpha$ ($E\alpha$) itseensä toteuttaa vaatimuksen $\mathcal{A} \| P$, koska $T\alpha$ ($E\alpha$) on polun ainoa solmu.

Induktioaskel: taulu τ saadaan taulusta τ' soveltamalla taulusääntöä τ':n polun P solmuun $T\beta$ tai $E\beta$ (jollekin lauseelle β). Induktio-oletuksen nojalla τ':ssa polku P' siten, että $\mathcal{A} \| P'$.

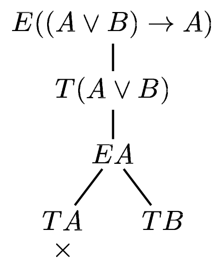
Jos $P \neq P'$, niin P' myös taulun τ polku.
Jos $P = P'$, niin $\mathcal{A} \| P$ ja päädymme tapausanalyysiin:

- $\beta = \neg\gamma$ ja $T\neg\gamma \in P$:
  $\Longrightarrow \mathcal{A} \models \neg\gamma$ (koska $\mathcal{A} \| P$) ja $E\gamma$ lisätään P:n jatkoksi
  $\Longrightarrow \mathcal{A} \not\models \gamma$ ja syntyvä τ:n polku on yhteensopiva $\mathcal{A}$:n kanssa.
- $\beta = \neg\gamma$ ja $E\neg\gamma \in P$:
  $\Longrightarrow \mathcal{A} \not\models \neg\gamma$ (koska $\mathcal{A} \| P$) ja solmu $T\gamma$ lisätään P:n jatkoksi.
  $\Longrightarrow \mathcal{A} \models \gamma$ ja syntyvä τ:n polku on yhteensopiva $\mathcal{A}$:n kanssa.

- Muut tapaukset (vaihtoehdot lauseeksi β) käsitellään vastaavasti.

Edellä osoitettu taulumenetelmän ominaisuus voidaan todeta seuraavasta esimerkistä.

Esimerkki. Totuusjakelu $\mathcal{A} = \{B\}$ on yhteensopiva allaolevan semanttisen taulun oikeanpuoleisen polun kanssa.



Täten $\mathcal{A} \not\models (A \vee B) \rightarrow A$, $\mathcal{A} \models A \vee B$, $\mathcal{A} \not\models A$ ja $\mathcal{A} \models B$.

4.4 Loogisten ongelmien ratkominen taulumenetelmällä

Lauseen toteutuvuuden tutkiminen

- Lause $\alpha \in \mathcal{L}$ on *toteutuva*

  $\iff \exists$ totuusjakelu $\mathcal{A} \subseteq \mathcal{P}$ s.e. $\mathcal{A} \models \alpha$

  $\iff$ juurisolmusta $T\alpha$ muodostetussa valmiissa semanttisessa taulussa on ainakin yksi ei-ristiriitainen polku P.

- Jokaisesta ei-ristiriitaisesta polusta P voidaan konstruoida α:lle malleja $\mathcal{A}$ seuraavasti. Jos atomiselle lauseelle $A \in \mathcal{P}$ pätee
  - $TA \in P$, niin $A \in \mathcal{A}$
  - $EA \in P$, niin $A \notin \mathcal{A}$
  - $TA \notin P$ ja $EA \notin P$, niin joko $A \in \mathcal{A}$ tai $A \notin \mathcal{A}$ (vapaa valinta).

Esimerkki. Tutkitaan lauseen $A \vee B \rightarrow A$ toteutuvuutta.

1. $T((A \vee B) \rightarrow A)$

2. $E(A \vee B)^{1.}$ 2. $TA^{1.}$

3. $EA^{2.}$

4. $EB^{2.}$

Ei-ristiriitaisista poluista voidaan muodostaa lauseelle mallit $\mathcal{A}_1 = \{\}$, $\mathcal{A}_2 = \{A\}$ ja $\mathcal{A}_3 = \{A, B\}$.

Lausejoukon toteutuvuuden tutkiminen

- Äärellinen lausejoukko $\Sigma = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathcal{L}$ on toteutuva

  $\iff$ lause $\alpha_1 \wedge \cdots \wedge \alpha_n \in \mathcal{L}$ on toteutuva

  $\iff$ juurisolmusta $T(\alpha_1 \wedge \cdots \wedge \alpha_n)$ muodostetussa valmiissa semanttisessa taulussa on ainakin yksi ei-ristiriitainen polku P.

- Lausejoukolle voidaan konstruoida malleja samaan tapaan kuin yksittäisen lauseen tapauksessa.

- Käytännössä juurisolmu $T(\alpha_1 \wedge \cdots \wedge \alpha_n)$ voidaan jättää kirjoittamatta näkyviin tilan säästämiseksi.

Lauseen pätevyyden tutkiminen

- $\models \alpha$

  $\iff \neg\alpha$ on toteutumaton

  $\iff$ juurisolmusta $T\neg\alpha$ muodostettu valmis semanttinen taulu on ristiriitainen

  $\iff$ juurisolmusta $E\alpha$ muodostettu valmis semanttinen taulu on ristiriitainen.

Täten taulumenetelmästä saadaan lauselogiikalle todistusmenetelmä.

Määritelmä. Taulu τ on lauseen α *todistus*, jos taulun τ juurisolmuna on $E\alpha$ ja τ on ristiriitainen (ja siten myös valmis).

Jos α:lla on todistus, on α *teoreema/todistuva* ($\vdash \alpha$).

Esimerkki. Osoitetaan lause $A \wedge B \to A \vee B$ teoreemaksi:

$$
\begin{array}{c}
1.\ E(A \wedge B \to A \vee B) \\
| \\
2.\ T(A \wedge B)^{1.} \\
| \\
3.\ E(A \vee B)^{1.} \\
| \\
TA^{2.} \\
| \\
TB^{2.} \\
| \\
EA^{3.} \\
| \\
EB^{3.} \\
\times
\end{array}
$$

Todistusmenetelmän virheettömyys ja täydellisyys

Todistusmenetelmä M on

- *virheetön*, jos lauseen α todistettavuudesta menetelmällä M ($\vdash_M \alpha$) seuraa lauseen α pätevyys ($\models \alpha$).
- *täydellinen*, jos lauseen α pätevyydestä ($\models \alpha$) seuraa lauseen todistettavuus menetelmällä M ($\vdash_M \alpha$)

Edellä määritelty semanttiseen tauluun perustuva todistusmenetelmä on virheetön ja täydellinen, koska $\vdash \alpha \iff \models \alpha$.

Loogisen seuraavuuden tutkiminen

- $\{\phi_1, \ldots, \phi_n\} \models \phi$
  $\iff \models \phi_1 \wedge \cdots \wedge \phi_n \to \phi$
  $\iff$ juurisolmusta $E(\phi_1 \wedge \cdots \wedge \phi_n \to \phi)$ muodostettu valmis semanttinen taulu on ristiriitainen.
- Jos juurisolmusta $E(\phi_1 \wedge \cdots \wedge \phi_n \to \phi)$ muodostettuun valmiiseen semanttiseen tauluun jää ei-ristiriitaisia polkuja, näistä voidaan konstruoida vastamalleja $\mathcal{A}$ siten, että $\mathcal{A} \models \{\phi_1, \ldots, \phi_n\}$ ja $\mathcal{A} \not\models \phi$.
- Käytännössä taulun juureen tulevat solmut $E(\phi_1 \wedge \cdots \wedge \phi_n \to \phi)$ ja $T(\phi_1 \wedge \cdots \wedge \phi_n)$ voidaan jättää kirjoittamatta näkyviin.

Johdettavuus lausejoukosta

Määritelmä. Lause β on johdettavissa äärellisestä lausejoukosta $\Sigma = \{\alpha_1, \ldots, \alpha_n\}$ (merkitään $\Sigma \vdash \beta$), joss juurisolmusta $E(\alpha_1 \wedge \cdots \wedge \alpha_n \to \beta)$ muodostettu valmis semanttinen taulu on ristiriitainen.

Huomioita.

- $\Sigma \vdash \beta \iff \Sigma \models \beta$.
- Taulumenetelmä on siis virheetön ja täydellinen todistusmenetelmä myös tässä tapauksessa.

Loogisen ekvivalenssin tutkiminen

- Lauseet α ja β ovat loogisesti ekvivalentit
  $\iff \models \alpha \leftrightarrow \beta$
  $\iff$ juurisolmusta $E(\alpha \leftrightarrow \beta)$ muodostettu valmis semanttinen taulu on ristiriitainen.
- Lausejoukkojen loogisen ekvivalenssin toteaminen edellyttää mahdollisesti usean semanttisen taulun konstruointia (kts. loogisen ekvivalenssin määritelmät).

Ohjeita semanttisten taulujen laadtaan:

- Taulun solmussa $T\phi$ (tai $E\phi$) olevan lauseen ϕ rakenne määrää, mitä taulusääntöä käytetään.
  Esimerkki. Jos ϕ on muodoltaan implikaatio $\alpha \rightarrow \beta$ käytetään sääntöä $T(\alpha \rightarrow \beta)$ (tai $E(\alpha \rightarrow \beta)$).
- Solmujen hajoittamisjärjestyksellä voi usein vaikuttaa taulun kokoon. Haarautumista kannattaa välttää seuraavaan tapaan:

$$\begin{array}{c} \vdots \\ | \\ TA \\ | \\ 6.E(B \rightarrow C) \\ | \\ 7.T(A \vee B) \\ | \\ 8.E(A \wedge C) \end{array}$$

  - Solmu 6 käsitellään ensin, koska taulu ei näin haaraudu.
  - Solmu 8 olisi seuraavaksi paras vaihtoehto, koska syntyvistä poluista toinen (jolla solmu EA) on suoraan ristiriitainen.
  - Solmun 7 käsittely haarauttaa taulun.

4.5 Esimerkkejä

Boolen funktioiden ja lauselogiikan yhteys

Mikä tahansa Boolen funktio $f : \{0,1\}^n \rightarrow \{0,1\}$ voidaan esittää lauselogiikan lauseena:

- Arvoja 0 ja 1 vastaavat totuusarvot E ja T
- Boolen muuttuja a (jolla arvona 0 tai 1) esitetään atomisena lauseena A (jolla totuusarvo E tai T)
- Komplementti esitetään negaation $\neg$ avulla
- Tulo $\cdot$ ja summa $+$ esitetään konjunktion $\wedge$ ja disjunktion $\vee$ avulla.

Esimerkki. Funktiolle $f(a,b,c) = a \cdot \overline{b} + c$ saadaan näillä periaatteilla esitys $(A \wedge \neg B) \vee C$.

Väite. Olkoon f boolen funktio ja lause ϕ sitä vastaava esitys:

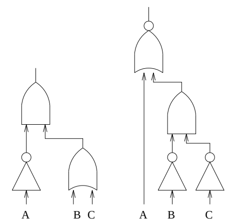
funktion f arvon laskeminen tietyillä muuttujien arvoilla vastaa lauseen ϕ totuusarvon laskentaa vastaavassa totuusjakelussa.

Esimerkki.
Jos $a = 0$, $b = 0$ ja $c = 1$, niin vastaava totuusjakelu on $\mathcal{A} = \{c\}$.

Aikaisemmalle esimerkille $f(0,0,1) = 0 \cdot \overline{0} + 1 = 0 \cdot 1 + 1 = 0 + 1 = 1$ ja vastaavasti $\mathcal{A} \models (A \wedge \neg B) \vee C$.

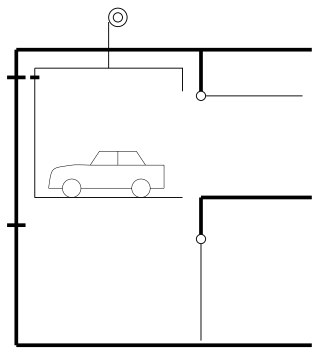
Esimerkki.

Jos haluamme osoittaa, että kaksi Boolen funktiota f ja f' ovat samat, riittää, että toteamme lause-esitysten ϕ ja ϕ' loogisen ekvivalenssin ($\phi \equiv \phi' \iff \models \phi \leftrightarrow \phi'$).



- $E(\neg A \wedge (B \vee C) \leftrightarrow \neg(A \vee (\neg B \wedge \neg C)))$
  - $T(\neg A \wedge (B \vee C))$
    - $E\neg(A \vee (\neg B \wedge \neg C))$
    - $T\neg A$
    - $T(B \vee C)$
    - EA
    - $T(A \vee (\neg B \wedge \neg C))$
      - TA $\times$
      - $T(\neg B \wedge \neg C)$
        - $T\neg B$
        - $T\neg C$
        - EB
        - EC
          - TB $\times$
          - TC $\times$
  - $E(\neg A \wedge (B \vee C))$
    - $T\neg(A \vee (\neg B \wedge \neg C))$
    - $E(A \vee (\neg B \wedge \neg C))$
    - EA
    - $E(\neg B \wedge \neg C)$
      - $E\neg A$
        - TA $\times$
      - $E(B \vee C)$
        - EB
        - EC
          - $E\neg B$
            - TB $\times$
          - $E\neg C$
            - TC $\times$

Esimerkki. Mallinnetaan yksinkertaistettua hissijärjestelmää lauselogiikalla:



Atomiset lauseet:

A_i: kerroksen i ovi on auki ja
K_i: hissi on kerroksessa i,

missä $i = 1$ tai $i = 2$.

Kuvataan lauselogiikalla järjestelmän sallitut tilat (eli haetaan lausejoukko Σ, jonka *mallit* vastaavat sallittuja tiloja).

- Tyhjällä lausejoukolla $\emptyset$ on $2^4 = 16$ mallia, kuten esimerkiksi $\mathcal{A} = \{K_1, K_2, A_1, A_2\}$.
  Tämä vastaa järjestelmän tilaa, jossa hissi on yhtäaikaa molemmissa kerroksissa ja molemmat ovet ovat auki. Tämä on mahdotonta, joten tarvitaan lisää lauseita (rajoitteita).
- Fyysisen maailman asettama rajoitus: $\neg K_1 \vee \neg K_2$
  (huomaa, että edellä annettu $\mathcal{A}$ ei ole tämän lauseen malli).
- Emme kuitenkaan halua asettaa vaatimusta $K_1 \vee K_2$
  (hissi voi olla matkalla kerrosten välillä).
- Oville asetettavat turvallisuusvaatimukset: $A_1 \rightarrow K_1$ ja $A_2 \rightarrow K_2$.

Järjestelmälle saadaan spesifikaatioksi lausejoukko

$$\Sigma = \{\neg K_1 \vee \neg K_2, A_1 \rightarrow K_1, A_2 \rightarrow K_2\}.$$

Etsitään spesifikaation mallit semanttisen taulun avulla.

- 1. $T(\neg K_1 \vee \neg K_2)$
- 2. $T(A_1 \rightarrow K_1)$
- 3. $T(A_2 \rightarrow K_2)$
  - 4. $T\neg K_1$ $^{1.}$
    - 5. EK_1 $^{4.}$
      - 6. EA_1 $^{2.}$
        - 7. EA_2 $^{3.}$
        - 7. TK_2 $^{3.}$
      - 6. TK_1 $^{2.}$ $\times$
  - 4. $T\neg K_2$ $^{1.}$
    - 5. EK_2 $^{4.}$
      - 6. EA_2 $^{2.}$
        - 7. EA_1 $^{3.}$
        - 7. TK_1 $^{3.}$
      - 6. TK_2 $^{2.}$ $\times$

Taulun avoimista haaroista lausejoukolle löydetään siis seuraavat mallit: $\mathcal{A}_1 = \emptyset$, $\mathcal{A}_2 = \{K_1\}$, $\mathcal{A}_3 = \{K_1, A_1\}$, $\mathcal{A}_4 = \{K_2\}$ ja $\mathcal{A}_5 = \{K_2, A_2\}$.

Voimme myös osoittaa, että hissin molemmat ovet eivät voi olla yhtäaikaisesti auki (turvallisuusominaisuus):

$\Sigma \models \neg(A_1 \wedge A_2)$.

1. $E\neg(A_1 \wedge A_2)$
2. $T(\neg K_1 \vee \neg K_2)$
3. $T(A_1 \rightarrow K_1)$
4. $T(A_2 \rightarrow K_2)$
5. $T(A_1 \wedge A_2)^{1.}$
6. $TA_1{}^{5.}$
7. $TA_2{}^{5.}$

$8. EA_1{}^{3.}$ $\times$ | $8. TK_1{}^{3.}$

$9. FA_2{}^{4.}$ $\times$ | $9. TK_2{}^{4.}$

$10. T\neg K_1{}^{2.}$ | $10. T\neg K_2{}^{2.}$

$11. FK_1$ $\times$ | $11. FK_2$ $\times$

5 Vaihtoehtoisia todistusjärjestelmiä

- Hilbertin järjestelmä
- Suppesin järjestelmä
- Järjestelmien välistä vertailua

5.1 Hilbertin järjestelmä

Aksiomat:

1. $\alpha \rightarrow (\beta \rightarrow \alpha)$
2. $(\alpha \rightarrow (\beta \rightarrow \gamma)) \rightarrow ((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma))$
3. $(\neg\beta \rightarrow \neg\alpha) \rightarrow ((\neg\beta \rightarrow \alpha) \rightarrow \beta)$

Päättelysääntö:

$$\frac{\alpha \rightarrow \beta \quad \alpha}{\beta} \quad \text{(modus ponens)}$$

Huomio. Hilbertin järjestelmässä negaatio ja implikaatio ovat peruskonnektiiveina.

Määritelmä. Olkoon Σ joukko lauseita.

Todistus Σ:sta on jono lauseita $\alpha_1, \ldots, \alpha_n$ siten, että kaikille $i \in \{1, \ldots, n\}$ pätee jokin seuraavista

- $\alpha_i \in \Sigma$,
- α_i on aksioma tai
- α_i on saatu modus ponensilla lauseista α_j, missä $j < i$ ja $\alpha_k = (\alpha_j \rightarrow \alpha_i)$, missä $k < i$.

Lause α on *teoreema/todistuva* Hilbert-järjestelmässä ($\vdash_{\mathrm{H}} \alpha$), joss $\emptyset \vdash_{\mathrm{H}} \alpha$

Lause α on *johdettavissa* lausejoukosta Σ Hilbert-järjestelmällä ($\Sigma \vdash_{\mathrm{H}} \alpha$), joss on olemassa todistus $\alpha_1, \ldots, \alpha_n$ Σ:sta siten, että $\alpha = \alpha_n$.

Esimerkki. Osoitetaan $\{A, B \to (A \to C)\} \vdash_{\mathrm{H}} B \to C$.

1.	A	olettamus
2.	$B \to (A \to C)$	olettamus
3.	$A \to (B \to A)$	aksioma 1
4.	$B \to A$	MP, 1, 3
5.	$(B \to (A \to C)) \to ((B \to A) \to (B \to C))$	aksioma 2
6.	$(B \to A) \to (B \to C)$	MP, 2, 5
7.	$B \to C$	MP, 4, 6

Väite. Olkoon lausejono $\alpha_1, \ldots, \alpha_n$ todistus lausejoukosta Σ Hilbertin järjestelmässä. Tällöin $\Sigma \models \alpha_i$ kaikille $i \in \{1, \ldots, n\}$.

Todistus induktiolla i:n suhteen.

- **Perustapaus:**
  Jos α_1 on aksioma $\implies \models \alpha_1 \implies \Sigma \models \alpha_1$.
  Jos $\alpha_1 \in \Sigma \implies \Sigma \models \alpha_1$.
- **Induktioaskel:**
  Tapaukset missä α_i on aksioma tai $\alpha_i \in \Sigma$ käsitellään kuten edellä.

- **Induktioaskel jatkuu:**
  Jos α_i saatiin modus ponensilla lauseista α_j ja $\alpha_k = (\alpha_j \to \alpha_i)$, missä $j < i$ ja $k < i$, niin induktiohypoteesin nojalla saadaan $\Sigma \models \alpha_j$ ja $\Sigma \models \alpha_j \to \alpha_i$.
  Oletetaan $\Sigma \not\models \alpha_i$
  $\implies \exists$ totuusjakelu $\mathcal{A}$ s.e. $\mathcal{A} \models \Sigma$ ja $\mathcal{A} \not\models \alpha_i$
  $\implies \mathcal{A} \not\models \alpha_j$ (koska $\Sigma \models \alpha_j \to \alpha_i \implies \mathcal{A} \models \alpha_j \to \alpha_i$)
  $\implies \Sigma \not\models \alpha_j$, ristiriita.
  Siis $\Sigma \models \alpha_i$.

Hilbertin järjestelmä on virheetön ja täydellinen.

Väite. Jos $\Sigma \vdash_{\mathrm{H}} \alpha$, niin $\Sigma \models \alpha$ (todistettiin edellä).
Jos $\Sigma \models \alpha$, niin $\Sigma \vdash_{\mathrm{H}} \alpha$ (todistus sivuutetaan).

5.2 Suppesin järjestelmä

Suppesin luonnollisen päättelyn järjestelmässä ei ole aksiomia ja päättelysäännöt ovat seuraavat:

MP (modus ponens):

$$\frac{\alpha \to \beta \quad \alpha}{\beta}$$

TP (modus tollendo ponens):

$$\frac{\alpha \lor \beta \quad \neg\alpha}{\beta}$$

TT (modus tollendo tollens):

$$\frac{\alpha \to \beta \quad \neg\beta}{\neg\alpha}$$

Vaihtosäännöt:

KV:

$$\frac{\alpha \land \beta}{\beta \land \alpha}$$

DV:

$$\frac{\alpha \lor \beta}{\beta \lor \alpha}$$

Tuonti- ja eliminointisäännöt:

KNT: $\dfrac{\alpha}{\neg\neg\alpha}$ KT: $\dfrac{\alpha \quad \beta}{\alpha \wedge \beta}$ DT: $\dfrac{\alpha}{\alpha \vee \beta \quad \beta \vee \alpha}$ ET: $\dfrac{\alpha \rightarrow \beta \quad \beta \rightarrow \alpha}{\alpha \leftrightarrow \beta}$

KNE: $\dfrac{\neg\neg\alpha}{\alpha}$ KE: $\dfrac{\alpha \wedge \beta}{\alpha \quad \beta}$ DE: $\dfrac{\alpha \vee \alpha}{\alpha}$ EE: $\dfrac{\alpha \leftrightarrow \beta}{\alpha \rightarrow \beta \quad \beta \rightarrow \alpha}$

DM 1: $\dfrac{\alpha \wedge \beta}{\neg(\neg\alpha \vee \neg\beta)}$ DM 2: $\dfrac{\neg(\alpha \wedge \beta)}{\neg\alpha \vee \neg\beta}$ DM 3: $\dfrac{\alpha \vee \beta}{\neg(\neg\alpha \wedge \neg\beta)}$ DM 4: $\dfrac{\neg(\alpha \vee \beta)}{\neg\alpha \wedge \neg\beta}$

HS: $\dfrac{\alpha \rightarrow \beta \quad \beta \rightarrow \gamma}{\alpha \rightarrow \gamma}$ DS: $\dfrac{\alpha \vee \beta \quad \alpha \rightarrow \gamma \quad \beta \rightarrow \delta}{\gamma \vee \delta}$

ET (ehdollinen todistaminen):

$$\dfrac{\begin{array}{c}[\alpha]^{(1)} \\ \vdots \\ \beta\end{array}}{\alpha \rightarrow \beta}$$

ES (epäsuora todistaminen):

$$\dfrac{\neg\beta \rightarrow \alpha \wedge \neg\alpha}{\beta}$$

(1) Hakasulut tarkoittavat, ettei johtopäätös $\alpha \rightarrow \beta$ riipu apuolettamuksesta α.

Määritelmä. Todistus lausejoukosta Σ on jono lauseita $\alpha_1, \ldots, \alpha_n$, joihin liittyy lausejoukot $H_0 = \emptyset$ ja $H_1, \ldots, H_n$ (apuolettamukset) siten, että kaikille $i \in \{1, \ldots, n\}$ pätee jokin seuraavista:

- $H_i = H_{i-1}$ ja $\alpha_i \in \Sigma$ tai
- $H_i = H_{i-1} \cup \{\alpha_i\}$ ja α_i voi olla mikä tahansa lause tai
- $H_i = H_{i-1}$ ja α_i on saatu jollain Suppes-järjestelmän päättelysäännöllä (paitsi ehdollisen todistamisen säännöllä) jonon aikaisemmista lauseista α_j $(j < i)$, joille $H_j \subseteq H_i$ tai
- $H_i = H_{i-1}$ ja $\alpha_i = \alpha_j$ jollekin jonon aikaisemmalle lauseelle α_j $(j < i)$, jolle $H_j \subseteq H_i$ tai
- $H_i = H_{i-1} - \{\alpha_j\}$ ja $\alpha_i = \alpha_j \rightarrow \alpha_{i-1}$ on saatu ehdollisen todistamisen säännöllä viimeisimmästä apuolettamuksesta α_j $(j < i)$ ja α_{i-1}.

Määritelmä. Lause α on *johdettavissa* lausejoukosta Σ Suppes-järjestelmällä ($\Sigma \vdash_S \alpha$), joss on olemassa todistus $\alpha_1, \ldots, \alpha_n$ Σ:sta siten, että $\alpha_n = \alpha$ ja $H_n = \emptyset$.

Lause α on *teoreema/todistuva* Suppes-järjestelmässä ($\vdash_S \alpha$), joss $\emptyset \vdash_S \alpha$

Väite. Suppesin järjestelmä on lauselogiikan virheetön ja täydellinen todistusmenetelmä ($\Sigma \vdash_S \alpha \iff \Sigma \models \alpha$).

Esimerkki.

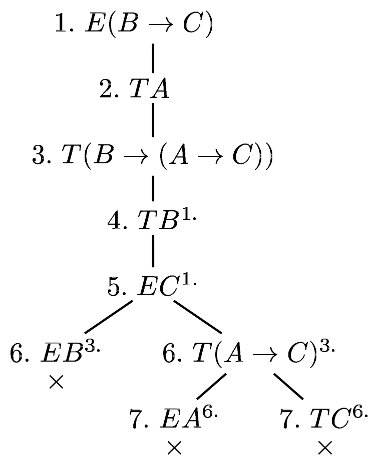
Osoitetaan $\{A, B \to (A \to C)\} \vdash_S B \to C$.

1.	A	olettamus	$H_1 = \emptyset$
2.	$B \to (A \to C)$	olettamus	$H_2 = \emptyset$
3.	B	apuolettamus	$H_3 = \{B\}$
4.	$A \to C$	MP, 3, 2	$H_4 = \{B\}$
5.	C	MP, 1, 4	$H_5 = \{B\}$
6.	$B \to C$	ET, 3, 5	$H_6 = \emptyset$

Käytännössä apuolettamusten joukoista pidetään kirjaa tekemällä todistukseen sisennyksiä, joloin H_i:t voidaan jättää pois todistuksesta.

5.3 ärjestelmien vertailua

Esimerkki.
Osoitetaan vielä semanttisella taululla, että $B \to C$ on johdettavissa lausejoukosta $\{A, B \to (A \to C)\}$.

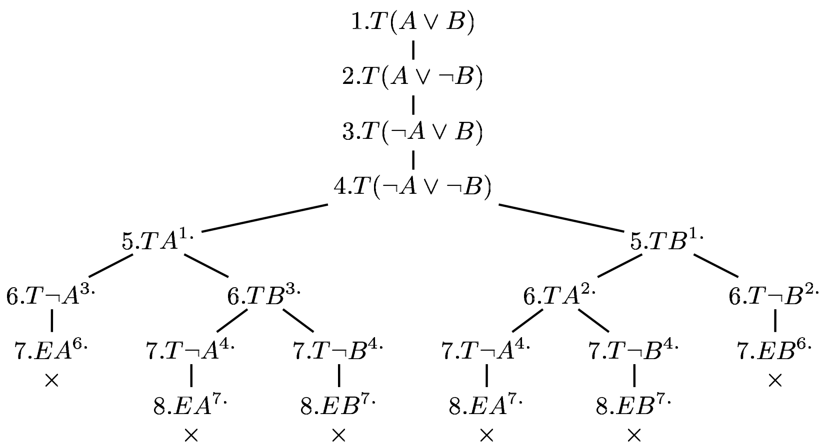


1. **Hilbertin järjestelmä**
   - + Minimalistinen koneisto teoreemien tuottamiseksi.
   - – Yksittäisten todistusten löytäminen voi olla vaikeaa.
   - – Hankala todeta milloin lause ei ole todistuva/johdettavissa.

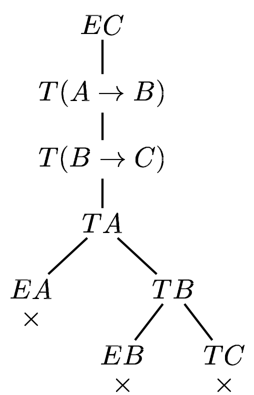
2. **Suppesin järjestelmä**
   - + Päättelysääntöjen laaja valikoima tukee erilaisten logisten päätelmien tekemistä ja todistusten löytämistä.
   - – Hankala todeta milloin lause ei ole todistuva/johdettavissa.

3. **Semanttinen taulu**
   - + Lauseiden syntaksi ohjaa päättelysäännön (taulusäännöt) valintaa.
   - + Mikäli lause ei ole todistuva/johdettavissa saadaan konkreettinen vastaesimerkki.
   - + Helppo toteuttaa tietokoneella.
   - – Semanttinen taulu ei ole tehokkain mahdollinen menetelmä lauselogiikan päättelyongelmien ratkomiseen.

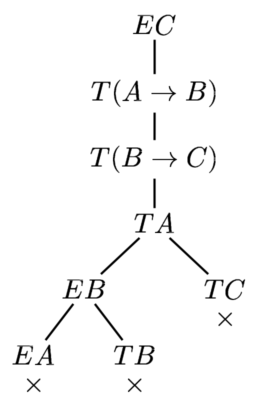
Esimerkki. Pahimmassa tapauksessa taulun koko voi kasvaa eksponentiaalisesti atomisten lauseiden lukumäärään nähden.



Esimerkki. Semanttisella taululla pystytään *simuloimaan* monia Suppes-järjestelmän päättelysääntöjä.



"Modus ponens"



"Modus tollendo tollens"

6 Normaalimuodot

- Konjunktiivinen ja disjunktiivinen normaalimuoto
- Muunnokset normaalimuotoihin
- Normaalimuotojen hakeminen taulumenetelmällä
- Normaalimuotojen sieventäminen
- Lauseiden klausuulimuoto

Motivaatio

Normaalimuotojen tarkoituksena on saattaa tutkittava asia johonkin säännölliseen muotoon, jotta sen käsittely saadaan yksinkertaisemmaksi. Esimerkkeinä mainittakoon seuraavat.

- Polynomien esittäminen muodossa

$$c_n \cdot x^n + c_{n-1} \cdot x^{n-1} + \cdots + c_0 \cdot x^0.$$

- Lineaaristen yhtälöryhmien esittäminen matriiseina:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

6.1 Konjunktiivinen ja disjunktiivinen normaalimuoto

Määritelmä. Jos A on atominen lause, niin A ja $\neg A$ ovat *literaaleja*.

Määritelmä. *Positivisen* literaalin A *komplementti* $\overline{A} = \neg A$ ja *negatiivisen* literaalin $\neg A$ *komplementti* $\overline{\neg A} = A$.

Määritelmä. Lause α on *konjunktiivisessa normaalimuodossa*, jos α on muodoltaan konjunktio $\beta_1 \land \beta_2 \land \cdots \land \beta_n$, missä jokainen β_i on literaaleista $l_1, l_2, \ldots, l_{m_i}$ koostuva disjunktio $l_1 \lor l_2 \lor \cdots \lor l_{m_i}$.

Disjunktiivinen normaalimuoto määritellään samaan tapaan, mutta konjunktion ja disjunktion roolit vaihdetaan keskenään:

Määritelmä. Lause α on *disjunktiivisessa normaalimuodossa*, jos α on muodoltaan disjunktio $\beta_1 \lor \beta_2 \lor \cdots \lor \beta_n$, missä jokainen β_i on literaaleista $l_1, l_2, \ldots, l_{m_i}$ koostuva konjunktio $l_1 \land l_2 \land \cdots \land l_{m_i}$.

Esimerkki.

KNM:	$(A_1 \lor \neg A_2) \land (\neg A_1 \lor \neg A_3) \land (A_5 \lor A_6 \lor A_7)$
	$A_5 \land (\neg A_3 \lor A_6)$
KNM & DNM:	$A_1 \lor \neg A_2 \lor A_3$
	$\neg A_7$
	$A_1 \land A_2 \land A_3$
DNM:	$(\neg A_1 \land \neg A_2) \lor (A_3 \land A_1) \lor (A_1 \land \neg A_2 \land A_3)$
	$(A_1 \land \neg A_1) \lor A_2$

Väite. Jokaiselle lauselogiikan lauseelle α on olemassa konjunktiivisessa (disjunktiivisessa) normaalimuodossa oleva lause β, joka on loogisesti ekvivalentti α:n kanssa.

Määritelmä. Sanotaan, että em. β on α:n konjunktiivinen (disjunktiivinen) normaalimuoto.

Esimerkki. $A \lor (\neg B \land C)$ on loogisesti ekvivalentti konjunktiivisessa normaalimuodossa olevan lauseen $(A \lor \neg B) \land (A \lor C)$ kanssa.

A	B	C	$\neg B$	$\neg B \land C$	$A \lor \neg B$	$A \lor C$	$A \lor (\neg B \land C)$	$(A \lor \neg B) \land (A \lor C)$
F	F	F	T	F	T	F	F	F
F	F	T	T	T	T	T	T	T
F	T	F	F	F	F	F	F	F
F	T	T	F	F	F	T	F	F
T	F	F	T	F	T	T	T	T
T	F	T	T	T	T	T	T	T
T	T	F	F	F	T	T	T	T
T	T	T	F	F	T	T	T	T

6.2 Muunnokset normaalimuotoihin

Mikä hyvänsä lauselogiikan lause voidaan muuttaa konjunktiiviseen tai disjunktiiviseen normaalimuotoon seuraavalla menettelyllä.

1. Poista konnektiivit $\leftrightarrow$ seuraavanlaisilla korvauksilla:

   $\alpha \leftrightarrow \beta \quad \rightsquigarrow \quad (\alpha \rightarrow \beta) \land (\beta \rightarrow \alpha) \qquad (1)$

   $\alpha \leftrightarrow \beta \quad \rightsquigarrow \quad (\neg\alpha \lor \beta) \land (\neg\beta \lor \alpha) \qquad (1')$

2. Poista konnektiivit $\rightarrow$ seuraavanlaisilla korvauksilla:

   $\alpha \rightarrow \beta \quad \rightsquigarrow \quad \neg\alpha \lor \beta \qquad (2)$

3. Siirrä negaatiot välittömästi atomisten lauseiden eteen:

   $\neg\neg\alpha \quad \rightsquigarrow \quad \alpha \qquad (3)$

   $\neg(\alpha \lor \beta) \quad \rightsquigarrow \quad \neg\alpha \land \neg\beta \qquad (4)$

   $\neg(\alpha \land \beta) \quad \rightsquigarrow \quad \neg\alpha \lor \neg\beta \qquad (5)$

Viimeinen vaihe riippuu tavoiteltavasta normaalimuodosta:

4. (KNM) Siirrä $\wedge$-konnektiivit $\vee$-konnektiivien ulkopuolelle:

$$\alpha \vee (\beta \wedge \gamma) \rightsquigarrow (\alpha \vee \beta) \wedge (\alpha \vee \gamma) \quad (6)$$
$$(\alpha \wedge \beta) \vee \gamma \rightsquigarrow (\alpha \vee \gamma) \wedge (\beta \vee \gamma) \quad (7)$$

(DNM) Siirrä $\vee$-konnektiivit $\wedge$-konnektiivien ulkopuolelle:

$$\alpha \wedge (\beta \vee \gamma) \rightsquigarrow (\alpha \wedge \beta) \vee (\alpha \wedge \gamma) \quad (8)$$
$$(\alpha \vee \beta) \wedge \gamma \rightsquigarrow (\alpha \wedge \gamma) \vee (\beta \wedge \gamma) \quad (9)$$

Huomio. Jokainen edellä annetuista muunnoksista säilyttää loogisen ekvivalenssin, joten lopputuloksena saadaan lauseelle konjunktiivinen tai disjunktiivinen normaalimuoto.

Esimerkki. Muutetaan lause $A \vee B \rightarrow (B \leftrightarrow C)$ konjunktiiviseen ja disjunktiiviseen normaalimuotoon.

$$A \vee B \rightarrow (B \leftrightarrow C)$$
$$A \vee B \rightarrow (B \rightarrow C) \wedge (C \rightarrow B) \quad (1)$$
$$\neg(A \vee B) \vee ((\neg B \vee C) \wedge (\neg C \vee B)) \quad (2)$$
$$(\neg A \wedge \neg B) \vee ((\neg B \vee C) \wedge (\neg C \vee B)) \quad (4)$$

Konjunktiivinen normaalimuoto:

$$(\neg A \vee ((\neg B \vee C) \wedge (\neg C \vee B))) \wedge (\neg B \vee ((\neg B \vee C) \wedge (\neg C \vee B))) \quad \textbf{(7)}$$
$$((\neg A \vee (\neg B \vee C)) \wedge (\neg A \vee (\neg C \vee B))) \wedge (\neg B \vee ((\neg B \vee C) \wedge (\neg C \vee B))) \quad \textbf{(6)}$$
$$((\neg A \vee (\neg B \vee C)) \wedge (\neg A \vee (\neg C \vee B))) \wedge ((\neg B \vee (\neg B \vee C)) \wedge (\neg B \vee (\neg C \vee B))) \quad \textbf{(6)}$$
$$(\neg A \vee \neg B \vee C) \wedge (\neg A \vee \neg C \vee B) \wedge (\neg B \vee \neg B \vee C) \wedge (\neg B \vee \neg C \vee B)$$

Disjunktiivinen normaalimuoto:

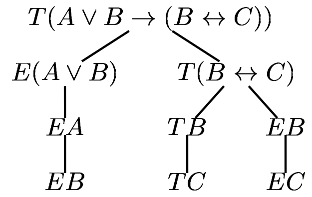
$$(\neg A \wedge \neg B) \vee ((\neg B \vee C) \wedge \neg C) \vee ((\neg B \vee C) \wedge B) \quad (8)$$
$$(\neg A \wedge \neg B) \vee (\neg B \wedge \neg C) \vee (C \wedge \neg C) \vee (\neg B \wedge B) \vee (C \wedge B) \quad (9)$$

6.3 Normaalimuotojen hakeminen taulumenetelmällä

Normaalimuodon voi hakea myös taulumenetelmällä.

Lauseen α disjunktiivinen normaalimuoto saadaan juurisolmusta $T\alpha$ muodostetun valmiin semanttisen taulun ristiriidattomista poluista.

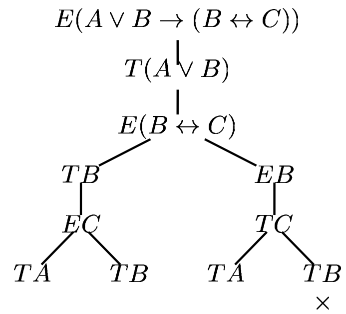
Esimerkki.



Vasemmanpuolisella polulla on vaatimukset EA ja EB, keskimmäisellä TB ja TC ja oikeanpuolisella EB ja EC. Näistä saadaan lauseen disjunktiiviseksi normaalimuodoksi $(\neg A \wedge \neg B) \vee (B \wedge C) \vee (\neg B \wedge \neg C)$.

Lauseelle α saadaan konjunktiivinen normaalimuoto lauseen $\neg\alpha$ disjunktiivisesta normaalimuodosta sääntöjen (3)-(5) avulla.

Esimerkki.



Ristiriidattomista poluista saadaan DNM lauseelle $\neg\alpha$:
$\neg\alpha \equiv (A \wedge B \wedge \neg C) \vee (B \wedge \neg C) \vee (A \wedge \neg B \wedge C)$.
Lauseen α konjunktiivinen normaalimuoto on tällöin:
$\alpha \equiv \neg\neg\alpha \equiv (\neg A \vee \neg B \vee C) \wedge (\neg B \vee C) \wedge (\neg A \vee B \vee \neg C)$.

6.4 Normaalimuotojen sieventäminen

Konjunktiivista/disjunktiivista normaalimuotoa voidaan sieventää mm. seuraavilla periaatteilla:

- Poistetaan literaalien disjunktioista/konjunktioista literaalien moninkertaiset esiintymät.

  Esimerkki.
  $(\neg A \vee \neg B \vee C) \wedge (\neg A \vee \neg C \vee B) \wedge (\neg B \vee \neg B \vee C) \wedge (\neg B \vee \neg C \vee B)$
  $\rightsquigarrow\ (\neg A \vee \neg B \vee C) \wedge (\neg A \vee \neg C \vee B) \wedge (\neg B \vee C) \wedge (\neg B \vee \neg C \vee B)$

- Poistetaan literaalien disjunktiot/konjunktiot, joissa esiintyy jokin literaali l ja sen komplementti $\bar{l}$.

  Esimerkki. Jatketaan edellisestä:
  $\rightsquigarrow\ (\neg A \vee \neg B \vee C) \wedge (\neg A \vee \neg C \vee B) \wedge (\neg B \vee C)$

- Jos **konjunktiivisessa** normaalimuodossa esiintyy disjunktiot $(l_1 \vee \cdots \vee l_n)$ ja $(k_1 \vee \cdots \vee k_m)$ s.e. $\{l_1, \ldots, l_n\} \subseteq \{k_1, \ldots, k_m\}$, poistetaan jälkimmäinen (joka on edellisen looginen seuraus).

  Esimerkki. $(A \vee B \vee C) \wedge (A \vee B) \wedge (C \vee \neg B \vee A) \wedge (A \vee C)$
  $\rightsquigarrow\ (A \vee B) \wedge (C \vee \neg B \vee A) \wedge (A \vee C)$

- Jos **konjunktiivisessa** normaalimuodossa esiintyy yksiliteraalinen disjunktio l, poistetaan muut disjunktiot, joissa l esiintyy, sekä mahdolliset komplementin $\bar{l}$ esiintymät muista disjunktioista.

  Esimerkki. $\neg A \wedge (A \vee B \vee C) \wedge (\neg A \vee \neg B) \rightsquigarrow \neg A \wedge (B \vee C)$

  Huomio. Jos disjunktiosta poistetaan kaikki jäsenet, jäljelle jää tyhjä disjunktio $\perp$, joka on aina epätosi.

  Esimerkki. $A \wedge B \wedge (\neg A \vee \neg B) \rightsquigarrow A \wedge B \wedge \neg B \rightsquigarrow A \wedge B \wedge \perp \rightsquigarrow \perp$
  (alkuperäinen lause on siis toteutumaton).

6.5 Lauseiden klausuulimuoto

- Atomiset lauseet A ja niiden negaatiot $\neg A$ ovat *literaaleja*.
- Literaalin komplementti: $\overline{A} = \neg A$, $\overline{\neg A} = A$.
- Literaalien $l_1, \ldots, l_n$ disjunktio $l_1 \vee \cdots \vee l_n$ on *klausuuli*.
- Klausuulit esitetään usein literaalien *joukkoina* $\{l_1, \ldots, l_n\}$.
- Joukko klausuuleita S edustaa klausuuliensa konjunktiota.

Huomio. Esitystavoilla on hienoiset erot:

$$\neg A \vee B \vee \neg A \ \rightsquigarrow\ \{\neg A, B\}$$
$$\neg A \vee B \ \rightsquigarrow\ \{\neg A, B\}$$
$$B \vee \neg A \ \rightsquigarrow\ \{\neg A, B\}$$

Konjunktiivisesta normaalimuodosta saadaan muodostettua sitä vastaava klausuulijoukko.

Esimerkki. Konjunktiivista normaalimuotoa $(A \vee \neg B) \wedge (\neg C \vee \neg A \vee D)$ vastaava klausuulijoukko on $\{\{A, \neg B\}, \{\neg C, \neg A, D\}\}$

Huomaa seuraavat erikoistapaukset:

- *Tyhjä klausuuli* $\square$ edustaa tyhjää disjunktiota (ja on siten aina epätosi).
- Tyhjä klausuulijoukko $\emptyset$ edustaa tyhjää konjunktiota (ja on siten aina tosi).

Huomio. Muistanet seuraavan analogian matematiikasta (tyhjät summat ja tulot): $\sum_{i=1}^{n} x_i = 0$ ja $\prod_{i=1}^{n} x_i = 1$, kun $n = 0$.

7 Resoluutio

- Totuusmääritelmä ja toteutuvuus klausuuleille
- Resoluutiosääntö
- Resoluutiotodistukset
- Resoluution virheettömyys ja täydellisyys
- Loogisten ongelmien ratkominen resoluutiolla

7.1 Totuusmäritelmä ja toteutuvuus klausuuleille

Määritelmä. Totuusjakelun $\mathcal{A} \subseteq \mathcal{P}$ literaaliesitys
$\text{Lit}(\mathcal{A}) = \mathcal{A} \cup \{\neg A \mid A \in \mathcal{P} - \mathcal{A}\}$.

Määritelmä. Klausuuli C on tosi totuusjakelussa $\mathcal{A} \subseteq \mathcal{P}$ (merkitään $\mathcal{A} \models C$), joss $C \cap \text{Lit}(\mathcal{A}) \neq \emptyset$.

Määritelmä. Klausuulijoukko S on tosi totuusjakelussa $\mathcal{A}$ (merkitään $\mathcal{A} \models S$), joss kaikille klausuuleille $C \in S$ pätee $\mathcal{A} \models C$.

Määritelmä. Totuusjakelu $\mathcal{A} \subseteq \mathcal{P}$ on klausuulijoukon S malli, joss $\mathcal{A} \models S$.

Esimerkki. Tarkastellaan atomisten lauseiden joukkoon $\mathcal{P} = \{A, B, C, D, E\}$ perustuvia klausuuleita.

Olkoon $\mathcal{A} = \{A\}$, joloin $\text{Lit}(\mathcal{A}) = \{A, \neg B, \neg C, \neg D, \neg E\}$.

Nyt esimerkiksi

$$\begin{array}{lcl}
\mathcal{A} & \models & \{\{A, C, E\}, \{\neg A, \neg B\}\} \\
\mathcal{A} & \models & \emptyset \\
\mathcal{A} & \not\models & \{\{A, B\}, \{C\}\} \\
\mathcal{A} & \models & \{\neg A, B, \neg D\} \\
\mathcal{A} & \not\models & \{\neg A, D\} \\
\mathcal{A} & \not\models & \Box
\end{array}$$

Määritelmä. Klausuulijoukko S on *toteutuva* joss S:llä on ainakin yksi malli $\mathcal{A} \subseteq \mathcal{P}$. Muuten S on *toteutumaton*.

Esimerkki. Klausuulijoukot $\{\{A, C, E\}, \{\neg A, \neg B\}\}$ ja $\emptyset$ ovat toteutuvia, koska esimerkiksi $\mathcal{A} = \{A\}$ on näiden molempien malli.

Esimerkki. Klausuulijoukot $\{\{A\}, \{\neg A\}\}$ ja $\{\Box\}$ ovat toteutumattomia.

7.2 Resoluutiosääntö

Määritelmä. Olkoot $C_1 = \{l, l_1, \ldots, l_n\}$ ja $C_2 = \{\bar{l}, l'_1, \ldots, l'_m\}$ klausuuleja. Klausuuli $C = (C_1 \cup C_2) - \{l, \bar{l}\} = \{l_1, \ldots, l_n, l'_1, \ldots, l'_m\}$ on klausuulien C_1 ja C_2 *yhdistelmä*.

Esimerkki. Sovelletaan resoluutiosääntöä seuraaviin klausuuleihin:

$$\{A, B, \neg C\} \qquad \{\neg A, D, \neg C\}$$

$$\{B, \neg C, D\}$$

Sääntöä on sovellettu literaalien A ja $\neg A$ suhteen. Klausuulit ovat joukkoja, joten $\neg C$ esiintyy yhdistelmässä $\{B, \neg C, D\}$ vain kertaalleen.

Huomio. Resoluutiosääntöä saa soveltaa korkeintaan yhden literaaliparin (l ja $\bar{l}$) suhteen kerrallaan.

Esimerkki. Tarkastellaan klausuulijoukkoa $S = \{\{A, \neg B\}, \{\neg A, B\}\}$.

- Klausuuljoukosta voidaan johtaa resoluutiosäännöllä klausuulit $\{A, \neg A\}$ (literaali $l = B$) ja $\{B, \neg B\}$ (literaali $l = A$).
- Edelleen näistä ja joukon S klausuuleista voidaan johtaa resoluutiosäännöllä ainoastaan S:ään kuuluvia klausuuleita.
- Missään tapauksessa S:stä ei saada resoluutiosäännöllä tyhjää klausuulia $\Box$ (tämä tarkoittaisi, että S on toteutumaton).
- Huomaa, että S on toteutuva ($\mathcal{A} \models S$) totuusjakelussa $\mathcal{A} = \{A, B\}$.

7.3 Resoluutiotodistukset

Lähtökohtana klausuulijoukko S, jonka klausuuleihin sovelletaan resoluutiosääntöä.

Määritelmä. Klausuulin C *johto* klausuulijoukosta S on äärellinen jono klausuuleja $C_1, \ldots, C_n$, missä $C_n = C$ ja jokaiselle C_i joko $C_i \in S$ tai C_i on joidenkin aikaisempien klausuulien yhdistelmä.

Määritelmä. Jos klausuulijoukosta S voidaan johtaa tyhjä klausuuli $\Box$, kyseistä johtoa kutsutaan S:n *hylkäykseksi* (refutaatioksi).

Ajatuksena on, että tällöin S joudutaan hylkäämään toteutuvana klausuulijoukkona.

Esimerkki. Tarkastellaan klausuulijoukkoa

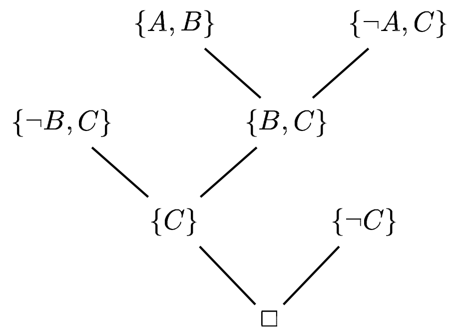
$$S = \{\{A, B\}, \{\neg A, C\}, \{\neg B, C\}, \{\neg C\}\},$$

jolle saadaan seuraava hylkäys:

1.	$\{A, B\}$	S
2.	$\{\neg A, C\}$	S
3.	$\{\neg B, C\}$	S
4.	$\{\neg C\}$	S
5.	$\{B, C\}$	1, 2
6.	$\{C\}$	3, 5
7.	$\Box$	4, 6

Resoluutiotodistuksen puuesitys

Edellä esitetty hylkäys voidaan kirjoittaa myös puumuotoon.



Huomio. Lehtisolmuina on ainoastaan klausuulijoukon S klausuleja ja vastaava lineaarinen hylkäys voidaan tuottaa käymällä puumuotoinen todistus syvyysjärjestyksessä lävitse.

Esimerkki. Klausuulijoukolle

$$S = \{\ \{A, B, \neg C, \neg D\}, \{\neg A, E\}, \{\neg B, \neg F\}, \{C, E\}, \{D, \neg F\}, \{\neg E\}, \{F\}\ \}$$

saadaan seuraava hylkäys:

1.	$\{A, B, \neg C, \neg D\}$	S
2.	$\{\neg A, E\}$	S
3.	$\{E, B, \neg C, \neg D\}$	1,2
4.	$\{\neg B, \neg F\}$	S
5.	$\{E, \neg F, \neg C, \neg D\}$	3,4
6.	$\{C, E\}$	S
7.	$\{E, \neg F, \neg D\}$	5,6
8.	$\{D, \neg F\}$	S
9.	$\{E, \neg F\}$	7,8
10.	$\{\neg E\}$	S
1.	$\{\neg F\}$	9,10
12.	$\{F\}$	S
13.	$\Box$	11, 12

7.4 Resoluution virheettömyys ja täydellisyys

Väite. Jos klausuulijoukolla S on malli $\mathcal{A} \subseteq \mathcal{P}$ ja C on kahden klausuulin $C_1 \in S$ ja $C_2 \in S$ yhdistelmä, niin $\mathcal{A}$ on myös klausuulijoukon $S' = S \cup \{C\}$ malli.

Todistus. Oletetaan $\mathcal{A} \not\models S \cup \{C\}$.

$\Longrightarrow$ $\mathcal{A} \not\models C$

$\Longrightarrow$ $C \cap \mathrm{Lit}(\mathcal{A}) = \emptyset$

$\Longrightarrow$ $C_1 \cap \mathrm{Lit}(\mathcal{A}) = \emptyset$ **tai** $C_2 \cap \mathrm{Lit}(\mathcal{A}) = \emptyset$

$\Longrightarrow$ $\mathcal{A} \not\models C_1$ tai $\mathcal{A} \not\models C_2$

$\Longrightarrow$ $\mathcal{A} \not\models S$, ristiriita.

Siis $\mathcal{A} \models S \cup \{C\}$.

Väite. Jos klausuulijoukolle S on hylkäys, niin S on toteutumaton.

Todistus. Oletetaan, että klausuulijoukolle S on hylkäys $C_1, \cdots, C_n$, missä $C_n = \Box$. Tehdään vastaoletus, että S on toteutuva.

Osoitetaan induktiolla i:n suhteen, että $S \cup \{C_1, \ldots, C_i\}$ on toteutuva.

Perustapaus $i = 0$: S on toteutuva (vastaoletus).

Induktioaskel: Klausuulijoukko $S \cup \{C_1, \ldots, C_{i-1}\}$ on toteutuva induktiohypoteesin nojalla. Hylkäyksen klausuuli C_i on saatu resoluutiosäännöllä tämän joukon kahdesta klausuulista. Siten myös $S \cup \{C_1, \ldots, C_i\}$ on toteutuva edellä todistamamme väitteen nojalla.

Induktiotodistuksesta seuraa, että myös $S \cup \{C_1, \ldots, C_n\}$ on toteutuva.

Ristiriita, koska $C_n = \Box$ on epätosi kaikissa totuusjakeluissa.

Puukonstruktio täydellisyystodistusta varten

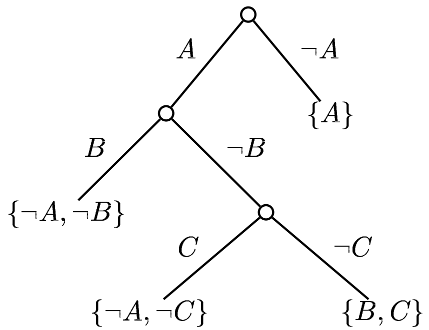
Olkoon S atomisten lauseiden joukkoon $\mathcal{P} = \{A_1, \ldots, A_n\}$ perustuva klausuulijoukko, jolle muodostetaan binääripuu seuraavilla periaatteilla.

Olkoon s syvyydellä i $(0 \leq i \leq n)$ oleva puun solmu (juurisolmu on syvyydellä 0) ja L_s niiden literaalien joukko, jotka ovat juurisolmusta solmuun s johtavilla kaarilla.

Jos $\overline{C} = \{\bar{l} \mid l \in C\} \subseteq L_s$ jollekin klausuulille $C \in S$ (eli $\mathcal{A} \not\models C$ kaikille totuusjakeluille $\mathcal{A}$ s.e. $L_s \subseteq \mathrm{Lit}(\mathcal{A})$), merkitään C solmuun s ja lopetetaan puun laajentaminen tästä solmusta eteenpäin. Muutoin:

1. Jos $i < n$, merkitään solmulle s vasen lapsi s_v ja oikea lapsi s_o sekä merkitään näihin solmusta s johtaville kaarille literaalit A_i ja $\neg A_i$. Jatketaan puun laajentamista solmuista s_v ja s_o vastaavasti.
2. Jos $i = n$, lopetetaan puun laajentaminen solmusta s eteenpäin.

Esimerkki. Konstruoidaan edellä kuvattu binääripuu klausuulijoukolle $S = \{\{A\}, \{B, C\}, \{\neg A, \neg B\}, \{\neg A, \neg C\}\}$:



Huomio. Kuhunkin lehtisolmuun s merkitty klausuuli C on epätosi totuusjakeluissa $\mathcal{A}$, joille $L_s \subseteq \mathrm{Lit}(\mathcal{A})$.

Väite. Jos S on toteutumaton, niin binääripuun jokainen polku päättyy solmuun s, johon on merkitty klausuuli $C \in S$ s.e. $\overline{C} \subseteq L_s$.

Todistus. Vastaoletus: edellä kuvatulla tavalla muodostetussa binääripuussa on solmu s tasolla n siten, että $\overline{C} \not\subseteq L_s$ kaikille $C \in S$.
$\Longrightarrow L_s = \mathrm{Lit}(\mathcal{A})$ jollekin $\mathcal{A} \subseteq \mathcal{P}$ ja $C \cap L_s \neq \emptyset$ kaikille $C \in S$.
$\Longrightarrow \mathcal{A} \models C$ kaikille $C \in S$, eli S on toteutuva, ristiriita.

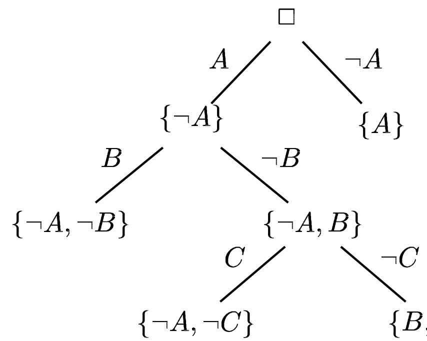
Väite. Jos klausuulijoukko S on toteutumaton, sille on hylkäys.

Todistus. Olkoon S toteutumaton, joloin binääripuun lehtisolmuina on S:n klausuulit. Käydään läpi sisäsolmut s (käänteisessä järjestyksessä).

Merkitään solmuun s klausuuliksi lapsisolmuihin merkittyjen klausuulien C_v ja C_o yhdistelmä C, jolle pätee $\overline{C} \subseteq L_s$ ($\overline{C_v} \subseteq L_{s_v}$ ja $\overline{C_o} \subseteq L_{s_o}$). Tämä ominaisuus siirtyy kaikille sisäsolmujen s klausuuleille C.

Juurisolmun s tapauksessa $L_s = \emptyset$, joten $\overline{C} = \emptyset$ ja $C = \Box$.

Esimerkki. Palataan edelliseen esimerkkiin ja muodostetaan ko. klausuulijoukolle S hylkäys:



Huomio. Tästä on helppo todeta edellä esitetty ominaisuus, että jokaiseen solmuun s merkitylle klausuulille C pätee $\overline{C} \subseteq L_s$.

7.5 Loogisten ongelmien ratkominen resoluutiolla

Pätevyyden tutkiminen resoluutiolla

Väite. $\models \alpha \iff \neg\alpha$ on toteutumaton.

- Menettely: muutetaan lause $\neg\alpha$ konjunktiiviseen normaalimuotoon ja edelleen klausuulijoukoksi S.
- Tällöin S on toteutumaton
  $\iff$ $\neg\alpha$:n KNM on toteutumaton
  $\iff$ $\neg\alpha$ on toteutumaton
  $\iff$ α on pätevä.

Esimerkki. Onko $(A \to B) \land (\neg A \to C) \to B \lor C$ pätevä?

Lauseen negaation KNM on $(\neg A \lor B) \land (A \lor C) \land \neg B \land \neg C$.

Klausuulijoukko $S = \{\{\neg A, B\}, \{A, C\}, \{\neg B\}, \{\neg C\}\}$.

Resoluutiotodistus (hylkäys):

1.	$\{\neg A, B\}$	S
2.	$\{A, C\}$	S
3.	$\{\neg B\}$	S
4.	$\{\neg C\}$	S
5.	$\{B, C\}$	1, 2
6.	$\{C\}$	3, 5
7.	$\square$	4, 6

Lause on siis pätevä.

Loogisen seuraavuuden tutkiminen resoluutiolla

Väite. $\Sigma \models \alpha \iff \Sigma \cup \{\neg\alpha\}$ on toteutumaton.

- Menettely: muutetaan lausejoukon $\Sigma \cup \{\neg\alpha\}$ lauseet konjunktiiviseen normaalimuotoon ja edelleen klausulijoukoksi S.
- Tällöin S on toteutumaton
  $\iff$ lausejoukon $\Sigma \cup \{\neg\alpha\}$ KMN:n joukko on toteutumaton
  $\iff$ lausejoukko $\Sigma \cup \{\neg\alpha\}$ on toteutumaton
  $\iff$ α on Σ:n looginen seuraus.

Esimerkki. Onko $\{\neg A \to B, B \lor C \to \neg B\} \models A$?

lause	KNM	klausuuleina
$\neg A$	$\neg A$	$\{\neg A\}$
$\neg A \to B$	$A \lor B$	$\{A, B\}$
$B \lor C \to \neg B$	$(\neg B \lor \neg B) \land (\neg C \lor \neg B)$	$\{\neg B\}, \{\neg B, \neg C\}$

Saadaan klausuulijoukko $S = \{\{\neg A\}, \{A, B\}, \{\neg B\}, \{\neg B, \neg C\}\}$.

1.	$\{\neg A\}$	S
2.	$\{A, B\}$	S
3.	$\{\neg B, \neg C\}$	S
4.	$\{\neg B\}$	S
5.	$\{B\}$	1, 2
6.	$\square$	4, 5

Vastaus: lause A on lausejoukon looginen seuraus.

Esimerkki. Palataan hissiesimerkin spesifikaatioon ja osoitetaan ko. turvallisuusominaisuus resoluutiolla:

lause	KNM	klausuuleina	
$\neg K_1 \vee \neg K_2$	$\neg K_1 \vee \neg K_2$	1. $\{\neg K_1, \neg K_2\}$	
$A_1 \rightarrow K_1$	$\neg A_1 \vee K_1$	2. $\{\neg A_1, K_1\}$	
$A_2 \rightarrow K_2$	$\neg A_2 \vee K_2$	3. $\{\neg A_2, K_2\}$	
$\neg\neg(A_1 \wedge A_2)$	$A_1 \wedge A_2$	4. $\{A_1\}$, 5. $\{A_2\}$	
	Hylkäys:	6. $\{K_1\}$	2,4
		7. $\{K_2\}$	3,5
		8. $\{\neg K_2\}$	1,6
		9. $\square$	7,8

$\Longrightarrow \neg(A_1 \wedge A_2)$ on muiden lauseiden looginen seuraus.

8 Laskennallisesta vaativuudesta

- Laskennan malli
- Keskeiset vaativuusluokat
- Redusoituvuus ja vaikeat ongelmat

8.1 Laskennan malli

Oletamme jatkossa, että laskennan mallina ovat *Turing-koneet*.

Määritelmä.
Deterministinen Turing-kone T on nelikkö $\langle A, S, s_0, t\rangle$, missä

- A on *aakkosto*, johon kuuluu aina erikoissymboli $\sqcup$ (tyhjä symboli).
- S on joukko tiloja, johon kuuluu aina annettu *alkutila* $s_0 \in S$ sekä erikoistilat k (kyllä), e (ei) ja p (pysähdy).
- $t : S \times A \rightarrow S \times A \times \{\leftarrow, \rightarrow, \downarrow\}$ on *tilansiirtofunktio*.

Huomio. Tyhjää merkkijonoa merkitään symbolilla ϵ.

Määritelmä. Turing-koneen T kokonaistilan määrää *konfiguraatio* $\langle s, v, w\rangle$, missä $s \in S$ on T:n tila ja $v \in A^*$ ja $w \in A^+$ ovat merkkijonoja. T käsittelee aina w:n ensimmäistä merkkiä.

- Laskenta alkaa konfiguraatiosta $\langle s_0, \epsilon, w\rangle$, missä merkkijono $w \in (A - \{\sqcup\})^*$ tai $w = \sqcup$ (T:n syöte).
- Yhdessä laskennan askeleessa siirrytään konfiguraatiosta $\langle s, v, aw\rangle$ tilansiirtofunktion t arvon $t(s, a) = \langle s', a', m\rangle$ perusteella uuteen konfiguraatioon seuraavasti:
  1. Jos $m = \downarrow$, uusi konfiguraatio on $\langle s', v, a'w\rangle$.
  2. Jos $m = \rightarrow$, uusi konfiguraatio on $\langle s', va', w'\rangle$, missä $w' = w$, jos $w \neq \epsilon$, ja $w' = \sqcup$, jos $w = \epsilon$.
  3. Jos $m = \leftarrow$ ja $v = v'b$ joillekin $v' \in A^*$ ja $b \in A$, uusi konfiguraatio on $\langle s', v', ba'w\rangle$.
- Laskenta päättyy, jos $s' \in \{k, e, p\}$.

Määritelmä. Turing-koneen T laskenta on sekvenssi konfiguraatioita $\langle s_0, v_0, w_0\rangle \xrightarrow{T} \ldots \xrightarrow{T} \langle s_{n-1}, v_{n-1}, w_{n-1}\rangle$ missä $s_{n-1} \in \{k, e, p\}$. Laskenta on hyväksyvä, jos $s_{n-1} = k$.

Esimerkki. Olkoon $A = \{0, 1, \sqcup\}$ ja $S = \{s_0, s_1, k, e, p\}$. Binääriluvun pariteetti voidaan tarkastaa seuraavalla Turing-koneella T:

S	A	$S \times A \times \{\rightarrow, \leftarrow, \downarrow\}$
s_0	0	$\langle s_0, 0, \rightarrow\rangle$
s_0	1	$\langle s_1, 1, \rightarrow\rangle$
s_0	$\sqcup$	$\langle k, \sqcup, \downarrow\rangle$
s_1	0	$\langle s_1, 0, \rightarrow\rangle$
s_1	1	$\langle s_0, 1, \rightarrow\rangle$
s_1	$\sqcup$	$\langle e, \sqcup, \downarrow\rangle$

Syötteellä 101 T suorittaa seuraavan laskennan:
$\langle s_0, \epsilon, 101\rangle$
$\xrightarrow{T} \langle s_1, 1, 01\rangle$
$\xrightarrow{T} \langle s_1, 10, 1\rangle$
$\xrightarrow{T} \langle s_0, 101, \sqcup\rangle$
$\xrightarrow{T} \langle k, 101, \sqcup\rangle$.

Epädeterministiset Turing-koneet

- Tilansiirtofunktio t korvataan tilansiirtorelaatiolla $t : S \times A \rightarrow 2^{S\times A\times\{\leftarrow,\rightarrow,\downarrow\}}$.
- Konfiguraatiossa $\langle s, v, aw\rangle$ valitaan epädeterministisesti $\langle s', a', m\rangle \in t(s, a)$ ja siirrytään tämän perusteella uuteen konfiguraatioon. Mahdollisia laskentoja voi olla useita.

Esimerkki. Olkoon $A = \{0, 1, \sqcup\}$ ja $S = \{s_0, k, e, p\}$. Määritellään epädeterministinen Turing-kone seuraavasti:

S	A	$2^{S\times A\times\{\leftarrow,\rightarrow,\downarrow\}}$
s_0	$\sqcup$	$\{\langle s_0, 0, \rightarrow\rangle, \langle s_0, 1, \rightarrow\rangle, \langle p, \sqcup, \downarrow\rangle\}$

Yksi mahdollinen laskenta: $\langle s_0, \epsilon, \sqcup\rangle \xrightarrow{T} \langle s_0, 1, \sqcup\rangle \xrightarrow{T} \langle s_0, 10, \sqcup\rangle \xrightarrow{T} \langle s_0, 101, \sqcup\rangle \xrightarrow{T} \langle p, 101, \sqcup\rangle$.

Määritelmä. Turing-kone T *hyväksyy* kielen $L \subseteq (A - \{\sqcup\})^*$, jos kaikille merkkijonoille $x \in (A - \{\sqcup\})^*$ pätee: T:llä on ainakin yksi hyväksyvä laskenta syötteellä $x \iff x \in L$.

Turing-koneiden käyttö päätösongelmien ratkaisemiseen

Päätösongelman O instanssin ratkaisuna on joko vastaus "kyllä" tai "ei".

Päätösongelman O ratkaiseminen Turing-koneella edellyttää ongelmainstanssien esittämistä merkkijonoina ja Turing-koneen T konstruointia siten, että T hyväksyy "kyllä"-instansseja vastaavan kielen.

Esimerkki. Lauselogiikan toteutuvuusongelmassa SAT:ssa on tarkoituksena selvittää, onko annettu lause $\phi \in \mathcal{L}$ toteutuva vai ei.

SAT-ongelmaa vastaava kieli ("kyllä"-instanssien joukko) on siis toteutuvien lauseiden ϕ joukko (lauseet merkkijonoesityksinä).

8.2 Keskeiset vaativuusluokat

- Erilaisten ongelmien laskennallista vaativuutta voidaan analysoida asettamalla Turing-koneen laskentaresurssille rajoituksia.

  Keskeinen rajoitus: Turing-kone T pysähtyy polynomisessa ajassa syötteen pituuden suhteen $\iff$ on olemassa polynomi p siten, että kaikilla syötteillä $w \in (A - \{\sqcup\})^*$ koneen T laskenta käsittää korkeintaan $p(|w|)$ erilaista konfiguraatiota.
- Kaksi keskeistä ongelmien luokkaa ovat
  1. $\mathbf{P}$: ongelmat, jotka voidaan ratkaista polynomisessa ajassa *deterministisellä* Turing-koneella.
  2. $\mathbf{NP}$: ongelmat, jotka voidaan ratkaista polynomisessa ajassa *epädeterministisellä* Turing-konella.
- Luokka $\mathbf{P}$ on luokan $\mathbf{NP}$ aliluokka (ja mitä ilmeisimmin aito).

Väite. SAT kuuluu luokkaan $\mathbf{NP}$.

Todistuksen idea:

Voidaan konstruoida epädeterministinen Turing-kone T, joka

- valitsee epädeterministisesti totuusjakelun $\mathcal{A}$,
- laskee syötteenä ϕ annetun lauseen totuusarvon $\mathcal{A}$:ssa ja
- pysähtyy tilaan k, jos $\mathcal{A} \models \phi$, ja muutoin tilaan e.

Tarvittava laskenta pystytään suorittamaan polynomisessa ajassa lauseen ϕ merkkijonoesityksen pituuden suhteen.

Voidaan osoittaa, että $\phi \in$ SAT $\iff$ koneella T on ainakin yksi hyväksyvä laskenta syötteellä ϕ.

8.3 Redusoituvuus ja vaikeat ongelmat

Määritelmä. Ongelma O on $\mathbf{C}$-vaikea, jos kaikki luokan $\mathbf{C}$ ongelmat voidaan redusoida O:ksi polynomisessa ajassa.

Ongelman O_1 *redusoituvuus* ongelmaksi O_2 edellyttää, että löytyy i:n pituuden suhteen polynomisessa ajassa (deterministisellä Turing-koneella) laskettava funktio $f : O_1 \to O_2$ siten, että $i \in O_1 \iff f(i) \in O_2$.

Määritelmä. Ongelma O on $\mathbf{C}$-täydellinen, jos O kuuluu luokkaan $\mathbf{C}$ ja O on $\mathbf{C}$-vaikea.

$\Longrightarrow$ $\mathbf{C}$-täydelliset ongelmat ovat vaativimpia luokan $\mathbf{C}$ ongelmia.

Väite. SAT on $\mathbf{NP}$-vaikea (Cook, 1971).

$\Longrightarrow$ SAT on $\mathbf{NP}$-täydellinen ongelma.

Todistuksen idea:

Jokaiselle epädeterministiselle Turing-koneelle T, merkkijonolle $w \in (A - \{\sqcup\})^*$ ja polynomille p on olemassa lausejoukko Σ s.e.

- koneella T on syötteellä w ainakin yksi hyväksyvä laskenta, jonka pituus on pienempi kuin $p(|w|)$ $\iff$ lausejoukko Σ on toteutuva.

Huomioita.

- Näin ollen epädeterministisen Turing-koneen suorittama polynominen laskenta voidaan palauttaa polynomisessa ajassa SAT-ongelman ratkaisemiseen.
- Pahimmassa tapauksessa SAT-ongelman ratkaisu vaatii eksponentiaalisen ajan lauseen pituuteen nähden.
- Tehokas algoritmi: Davis-Putnam [1960]