GRUPUL microINFORMATICA

Cluj-Napoca 2011

MIHAIELA LUPEA
ANDREEA MIHIŞ

LOGICI CLASICE şi CIRCUITE LOGICE

BCU Cluj-Napoca
MATEM201200001

TEORIE şi EXEMPLE

Reeditare

Descrierea CIP a Bibliotecii Naţionale a României
LUPEA, MIHAIELA-ANA
Logici clasice şi circuite logice : teorie şi exemple /
Lupea Mihaiela-Ana, Mihiş Andreea-Diana. - Ed. a 3-a, rev. –
Cluj Napoca : Editura Albastră, 2011
Bibliogr.
ISBN 978-973-650-276-7

I. Mihiş, Andreea-Diana

004.312

eferent ştiinţific:
Prof. Univ. Dr. Doina Tătar
Universitatea Babeş-Bolyai, Cluj-Napoca
Facultatea Matematică şi Informatică

itura Albastră

Director editură
Smaranda Derveşteanu

Tehnoredactare computerizată
Editura Albastră

Coperta
Editura Albastră

Tipărit
Editura Albastră

8742

itura este acreditată de CNCSIS
(Consiliul Naţional al Cercetării Ştinţifice din Învăţământul Superior)
şi este recomandată
Consiliului Naţional de Atestare a Titlurilor şi Diplomelor Universitare.

pyright © 2011
Toate drepturile asupra acestei lucrări aparţin
s.c. Albastra Casă de Editură s.r.l.
Reproducerea integrală sau parţială a textului sau a ilustraţiilor din această carte este posibilă numai cu acordul prealabil în scris al Casei de Editură Albastră.

CUPRINS

PREFAŢĂ

Scopul acestei lucrări este prezentarea *bazelor logice* ale informaticii: logica propoziţiilor şi logica predicatelor de ordinul I, algebre şi funcţii booleene. Sunt abordate aplicaţii ale logicii în matematică, informatică, electronică: formalizarea raţionamentului matematic şi cotidian, metode de demonstrare a teoremelor, proiectarea circuitelor logice secvenţiale şi combinaţionale.

Lucrarea combină prezentarea aspectelor teoretice ale logicilor clasice şi circuitelor logice, cu numeroase exemple explicate şi cu o bogată bază de probleme propuse spre rezolvare.

Capitolul 1 este dedicat *logicii propoziţionale*. Aspectele semantice discutate sunt: tabele de adevăr, validitate, inconsistenţă, echivalenţe logice, forme normale. Sunt prezentate sistemul axiomatic (deductiv), relaţia şi operatorul de derivabilitate (inferenţă) care modelează procesul de raţionament în logica propoziţiilor.

Logica predicatelor de ordinul I este subiectul **capitolului 2** al lucrării. Calculul predicatelor de ordinul I este prezentat ca un sistem axiomatic (deductiv) şi este stabilită legătura cu partea de semantică prin care formulele predicative sunt interpretate în universul modelat. Formele normale ale formulelor predicative, substituţiile şi unificatorii vor fi utilizate în rezoluţia predicativă. Metoda bazată pe teorema lui Herbrand reduce problema inconsistenţei unei mulţimi de formule predicative la calculul propoziţional.

Capitolul 3 este destinat *metodei tabelelor semantice*, o metodă de demonstrare semantică prin respingere. Formulele sunt descompuse folosind regulile $\alpha, \beta, \delta, \gamma$ cu scopul determinării modelelor acestora. Abordarea clasică, prin intermediul simbolizării grafice folosind arbori binari, este mai sugestivă, dar mai greu de implementat. Noua abordare, utilizând funcţia TP, permite o implementare mai uşoară şi eficientă a acestei metode de demonstrare.

Capitolul 4 prezintă două sisteme axiomatice complementare: *calculul secvenţelor* şi *calculul anti-secvenţelor*, utilizate pentru a demonstra validitatea/derivabilitatea, respectiv nevaliditatea/nederivabilitatea în logicile clasice.

În **capitolul 5** este studiată *metoda rezoluţiei*, o metodă de demonstrare prin respingere, foarte eficientă şi uşor de implementat. Rezoluţia este prezentată ca sistem axiomatic (formal) şi totodată din punct de vedere algoritmic. Sunt studiate strategiile rezoluţiei (saturării pe nivele, mulţimii suport, eliminării) şi rafinările (rezoluţia blocării, rezoluţia liniară) cu scopul eficientizării şi automatizării procesului rezolutiv.

Capitolul 6 conţine noţiuni de bază ale *algebrelor şi funcţiilor booleene*. ormele canonice ale funcţiilor booleene vor fi utilizate în simplificarea uncţiilor. Având ca domeniu de definiţie algebra booleană binară, funcţiile ooleene sunt foarte utile în proiectarea circuitelor logice.

Pentru a putea realiza circuite logice simple, în **capitolul 7** sunt rezentate trei *metode de simplificare* a funcţiilor booleene. *Metoda iagramelor Veitch-Karnaugh* este relativ uşor de folosit având o reprezentare rafică, dar este nepractică pentru funcţii cu mai mult de patru variabile. *letoda lui Quine-Mc'Clusky* poate fi aplicată şi pentru funcţii de mai multe ariabile datorită caracterului său analitic. *Metoda* savantului român, Grigore *loisil*, utilizează logica propoziţiilor şi poate fi implementată uşor.

Capitolul 8 se referă la *circuite logice*. Sunt prezentate diferite tipuri de orţi, de bază sau derivate şi exemple de circuite utilizate în partea hard a lculatoarelor. Totodată se exemplifică utilizarea simplificării funcţiilor oleene, care modelează funcţionarea circuitelor logice, pentru proiectarea nor circuite cât mai simple.

Ultimul capitol constituie o bază de probleme propuse pentru utilizare ât la seminarii, cât şi pentru studiu individual, oferind astfel posibilitatea rofundării cunoştinţelor teoretice prezentate pe parcursul lucrării.

Prin conţinutul său această carte se adresează tuturor celor interesaţi în noaşterea logicilor clasice şi a circuitelor logice, domenii fundamentale în formatică. Pentru specialiştii în informatică este oferită o bază teoretică în recţia aplicativă de construire a unor sisteme de demonstrare automată utile în atematică, inginerie soft, agenţi inteligenţi, robotică, limbaje naturale, vedere ificială.

Dorim să aducem calde mulţumiri doamnei Prof. Univ. Dr. Doina tar pentru discuţiile valoroase în domeniul logicilor clasice şi demonstrării oremelor, precum şi pentru comentariile constructive aduse pe parcursul borării acestei lucrări.

uj-Napoca
2008

Mihaiela-Ana Lupea
Andreea-Diana Mihiş

INTRODUCERE

Din cele mai vechi timpuri, gânditorii au încercat să studieze şi să încorseteze legile gândirii care guvernează modul de operare a minţii şi astfel a fost iniţiat câmpul de studiu al logicii. Filozoful grec Aristotel a fost printre primii care a codificat „gândirea corectă", adică procesele raţionale nerevizuibile, în faimoasa lege a silogismului.

Apărută iniţial ca domeniu de studiu al filozofilor, logica a pătruns şi ca domeniu de studiu în ştiinţele exacte: matematică şi informatică, cu scopul de a transpune cunoştinţele informale în termeni formali şi a raţiona asupra acestora, oferind astfel posibilitatea modelării gândirii raţionale.

Logicile clasice (calculul propoziţional şi calculul predicatelor de ordinul I) sunt metode formale de raţionament matematic, valid şi stau la baza metodelor de reprezentare a cunoştinţelor şi a logicilor neclasice apărute ulterior.

Alături de logicile clasice ce descriu o lume perfectă, în care funcţionează totdeauna un raţionament valid, au apărut logicile neclasice (nestandard). Aceste logici încearcă să modeleze lumea cea de toate zilele, în care oamenii, fiinţe imperfecte, limitate şi subiective, sunt supuşi cunoaşterii incomplete, uneori incorecte, iar deciziile care trebuie luate în aceste condiţii sunt deseori invalidate de noi informaţii. Fiecare dintre aceste logici neclasice încearcă să surprindă şi să modeleze aspecte din realitatea obişnuită, cotidiană.

Extinzând logicile clasice, dar păstrând caracterul de monotonie al raţionamentului modelat, au apărut: logicile multivalente şi nuanţate, logicile modale şi temporale. Aceste logici fac parte din clasa *logicilor monotone* deoarece mulţimea concluziilor derivate creşte odată cu adăugarea de noi ipoteze.

În *logicile multivalente* şi cele *nuanţate* nu mai există doar două valori de adevăr pentru variabilele propoziţionale, ci sunt introduse diferite grade de adevăr. Aceste logici sunt utilizate pentru tratarea informaţiilor incomplete sau ambigue.

Logicile modale formalizează afirmaţiile şi raţionamentul în care apar noţiunile de necesitate şi posibilitate. Scopul *logicii temporale*, derivată din logica modală, este de a modela raţionamentul cu propoziţii care îşi schimbă valoarea de adevăr în timp.

Clasa *logicilor nemonotone* modelează raţionamentul în care concluziile derivate sunt doar posibile (consistente), nu neapărat adevărate (valide), acestea putând fi ulterior invalidate prin adăugarea de noi ipoteze.

ɽaționamentul uman, cotidian, guvernat de cunoștințe incomplete, ambigue sau hiar incorecte este un raționament de tip nemonoton.

Logica, modelarea raționamentului și demonstrarea automată sunt ubdomenii de bază în inteligența artificială, acestea oferind un cadru de ormalizare și studiu teoretic a diferitelor tipuri de raționament, precum și nelte de automatizare a procesului deductiv, foarte utile în subdomeniile plicative: limbaje naturale, vedere artificială, agenți inteligenți, robotică 27, 33, 37].

Problema decizională a logicilor: *verificarea dacă o formulă conjectură) este consecință logică a unei mulțimi de formule (axiome și ipoteze)*, fost mult studiată în literatura de specialitate, rezultatele constând în propunerea nor metode de demonstrare.

Metodele de demonstrare se clasifică în *metode directe* și *metode prin spingere*. Metodele directe de demonstrare modelează raționamentul prin care construiește demonstrația concluziei (conjecturii) din axiome și ipoteze. etodele de demonstrare prin respingere modelează raționamentul reducerii la osurd, iar demonstrația constă în a arăta că negația concluziei (conjecturii) mpreună cu ipotezele și axiomele conduc la contradicție.

În 1965, Robinson a propus o metodă sintactică de demonstrare prin spingere, simplă și ușor de implementat, numită *rezoluție*. Ulterior această etodă a fost rafinată (rezoluția semantică, rezoluția liniară, rezoluția blocării), scopul eficientizării procesului rezolutiv.

Metoda tabelelor semantice introdusă de R. Smullyan în anul 1968, se zează pe considerații semantice. Demonstrația prin respingere a concluziei nstă în a căuta modelele pentru conjuncția ipotezelor și negația concluziei și arăta că acestea nu există.

Gentzen a propus *metoda deducției naturale*, ca o metodă directă și tactică, având la bază un sistem axiomatic care formalizează procesul ductiv. Ca îmbunătățire a acestei metode, au apărut cele două sisteme iomatice complementare *calculul secvențelor* și *calculul anti-secvențelor*, care nt utilizate pentru a demonstra validitatea/derivabilitatea, respectiv validitatea/nederivabilitatea în logicile clasice.

Metoda conecției se bazează pe o caracterizare matriceală a validității. ocedura de căutare a demonstrației este o reprezentare compactă și mai cientă a metodei tabelelor semantice.

Toate metodele de demonstrare propuse pentru logicile clasice au fost aptate ulterior pentru logicile neclasice: modale, temporale, multivalente, monotone.

După apariția sistemelor de calcul, o direcție de succes a constituit-o plementarea metodelor de demonstrare propuse și studiate doar teoretic, nstruindu-se *sisteme de demonstrare automată*, capabile să rezolve probleme ficile din diverse domenii.

Matematica a beneficiat de rezultatele unor astfel de sisteme de demonstrare automată astfel:

- În 10 octombrie 1996, după 8 zile de căutări, demonstratorul automat EQP a demonstrat că un grup de axiome formează baza algebrelor booleene, rezultat stabilit teoretic de H. Robbins încă din anul 1933.
- Sistemul OTTER a demonstrat diverse rezultate despre qvasi-grupuri.
- Rezultate noi în geometria euclidiană au fost demonstrate cu ajutorul sistemului Geometry Expert.

În domeniul generării și verificării produselor soft, sistemele de demonstrare automată au fost utilizate încă de la început:

- Sistemul KIDS (Kestrel Institute) – generarea unor algoritmi de planificare care depășesc eficiența unor algoritmi deja utilizați.
- Proiectul AMPHION (sponsorizat de NASA) – generarea unor programe pentru ghidarea sateliților.
- KIV (Karlsruhe Institute) – verificare soft academic și în domeniul industrial: soft pentru supervizarea cursului de neutroni într-un reactor nuclear și soft pentru transferul în siguranță a comenzilor într-un vehicul spațial.
- PVS (NASA) – verificarea softului care coordonează zborurile spațiale.

Cea mai mare aplicabilitate industrială a sistemelor de demonstrare automată o consituie domeniul verificării hard:

- Sistemul ACL2 – verificarea corectitudinii codului pentru împărțirea în virgulă mobilă la microprocesorul AMD5K86.
- ANALYTICA – verifică circuitul de împărțire care implementează standardul IEEE.
- Sistemul HOL este utilizat de Bell Laboratories pentru verificări hard.

Domenii potențiale ale utilizării demonstrării automate sunt: biologia, stiințele sociale, medicina, comerțul.

În 1854 George Boole a publicat „*An Investigation of the Laws of Thoughts, on Which are Founded the Mathematical Theories of Logic and Probability*", lucrare care introduce algebrele booleene și conține teoreme de bază ale logicii. După mai bine de o sută de ani, algebra dezvoltată de matematicianul englez a devenit scheletul teoretic al proiectului logic al calculatoarelor digitale.

În 1938 Claude Shannen a demonstrat că algebra booleană binară poate descrie operațiile ce au loc într-un circuit electric cu două valori (trece sau nu trece curentul electric) și întrerupătoare.

Marele savant român, Grigore Moisil a avut o contribuție deosebită în omeniul algebrei (algebre Lukasiewicz-Moisil), logicilor multivalente și ircuitelor logice (metoda de simplificare a funcțiilor booleene care îi poartă umele).

O deosebită importanță în înțelegerea principiilor care stau la baza ncționării calculatoarelor o au circuitele logice. Fie că se dorește înțelegerea ărții hard sau a celei soft, circuitele logice au o la fel de mare importanță. ceasta deoarece, dacă se știu utiliza algebrele booleene, simplifica funcțiile oleene și proiecta circuite logice, atunci circuitele logice hard proiectate vor veni mai simple și mai ieftin de produs și utilizat. Totodată, se vor înțelege mai ne instrucțiunile simple pe care le execută microprocesorul calculatorului, dar va avea la îndemână și un mijloc de a simplifica condițiile uneori complicate re apar în cod. Însă, cel mai mare câștig va fi din punct de vedere intelectual, ndirea logică dobândită în urma studierii active a acestei cărți permițându-i itorului să asimileze cu ușurință celelalte noțiuni informatice.

Prezența în această lucrare atât a părții de logică cât și a celei de cuite logice este una cât se poate de naturală, dat fiind faptul că, modelarea cuitelor logice se realizează prin intermediul funcțiilor booleene, funcții care niculează aceleași două valori logice („adevărat" și „fals"), dar și operatori ivalenți conectivelor din logicile clasice. Totodată algebra booleană binară la baza definirii semanticii logicii propozițiilor.

Studiul logicii, a modului de reprezentare a informației în cal ulator și ircuitelor logice face parte din bazele studiului calculatoarelor. Metodele ice prezentate pot fi folosite însă nu numai de informaticieni, logicieni, tematicieni, dar și de alte persoane pentru care raționamentul logic este esar, cum sunt persoanele implicate în procesul legislativ. Acestea nizează o metodă precisă de deducție sau de demonstrare.

Datorită omniprezenței pe care o au în ultimul timp calculatoarele în ța de zi cu zi, conținutul acestei cărți este orientat atât spre studiul teoretic al stor domenii de bază ale informaticii, cât și spre implementarea și aplicarea todelor prezentate în diverse domenii.

1. LOGICA PROPOZIȚIILOR

Capitolul este destinat prezentării aspectelor teoretice ale logicii propozițiilor. Acest formalism logic modelează doar raționamente simple din domeniul matematic și din viața reală, dar este foarte util în proiectarea circuitelor logice. Studirea logicii propoziționale este deosebit de importantă deoarece aceasta, împreună cu logica predicatelor de ordinul I stă la baza unor sisteme logice mai puternice (logicile modale, logicile temporale, logicile nemonotone) care formalizează raționamente complexe din viața reală. S-au utilizat ca și referințe bibliografice lucrările [1, 2, 7, 10, 16, 23, 32, 41, 43, 46].

1.1. Sintaxa logicii propoziționale

Pentru specificarea sintaxei logicii propoziționale este nevoie de precizarea alfabetului limbajului propozițional și a regulilor de formare a formulelor propoziționale.

$\Sigma_P = Var\_propoz \cup Conective \cup \{(,)\}$ - *alfabetul logicii propozițiilor;*

$Var\_propoz = \{p_1, p_2, ...\}$ - ***mulțime*** numărabilă de ***variabile propoziționale***;

$Conective = \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$

- $\neg$ (***negație***), notații alternative: ~, "*not*"
- $\wedge$ (***conjuncție***), notații alternative: &, "*and*"
- $\vee$ (***disjuncție***), notații alternative: |, "*or*"
- $\rightarrow$ (***implicație***), notații alternative: $\Rightarrow$
- $\leftrightarrow$ (***echivalență***), notații alternative: $\Leftrightarrow$

Conectivele logice (operatorii booleeni) sunt preluate din limbajul natural, iar prioritatea acestora în ordine descrescătoare este următoarea: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.

F_P - ***mulțimea formulelor propoziționale corect construite***, este cea mai mică mulțime de formule care satisface regulile:

- *baza*: $p_i \in F_P, i = 1,2,...$;
- *inducția*:
  dacă $U, V \in F_P$ atunci:

$\neg U \in F_P$, $U \wedge V \in F_P$, $U \vee V \in F_P$, $U \to V \in F_P$, $U \leftrightarrow V \in F_P$;

- *închiderea*: toate formulele din F_P se obțin doar prin aplicarea regulilor precedente de un număr finit de ori.

1.2. Semantica logicii propoziționale

Propozițiile logice sunt formalizări ale afirmațiilor propoziționale din limbajul natural, care sunt fie adevărate, fie false.

Scopul definirii semanticii logicii propoziționale este de a atribui un înțeles, o valoare de adevăr, formulelor propoziționale.

Domeniul semantic este mulțimea ***valorilor de adevăr***: {F(fals), T(true, adevărat)}, care satisfac relațiile: $\neg F = T$, $\neg T = F$.

Negațiile conectivelor logice uzuale sunt conective utile în proiectarea circuitelor digitale.

- $\uparrow$ ***nand*** ("not and", "*numai*"- Schaffer), unde $p \uparrow q \overset{not.}{=} \neg(p \wedge q)$
- $\downarrow$ ***nor*** ("not or", "*nici*"- Pierce), unde $p \downarrow q \overset{not.}{=} \neg(p \vee q)$
- $\oplus$ ***xor*** ("sau exclusiv"), unde $p \oplus q \overset{not.}{=} \neg(p \leftrightarrow q)$

Semantica conectivelor este furnizată de următoarele *tabele de adevăr*:

p	q	$\neg p$	$p\wedge q$	$p\vee q$	$p\to q$	$p\leftrightarrow q$	$p\uparrow q$	$p\downarrow q$	$p\oplus q$
T	T	F	T	T	T	T	F	F	F
T	F	F	F	T	F	F	T	F	T
F	T	T	F	T	T	F	T	F	T
F	F	T	F	F	T	T	T	T	F

Observații:

- O conjuncție este adevărată doar când ambii operanzi sunt adevărați.
- O disjuncție ("sau inclusiv") este falsă doar când ambii operanzi sunt falși.
- Generalizând, o conjuncție/*disjuncție* cu un număr oarecare de operanzi este adevărată/*falsă* doar când toți operanzii sunt adevărați/*falși* și este falsă/*adevărată* când cel puțin un operand este fals/*adevărat*.
- Implicația $p \to q$ este falsă doar când ipoteza p este adevărată și concluzia q este falsă ("adevăr" nu poate implica "fals").

- Echivalența $p \leftrightarrow q$ este adevărată dacă și numai dacă p și q au aceleași valori de adevăr.
- "sau exclusiv" este adevărat doar când cei doi operanzi au valori de adevăr opuse.

Definiția 1.1.

O ***interpretare*** a formulei $U(p_1, p_2, ..., p_n) \in F_P$ este o funcție $i: \{p_1, p_2, ..., p_n\} \to \{F, T\}$ care asignează valori de adevăr variabilelor propoziționale componente și poate fi extinsă la funcția $i: F_P \to \{F, T\}$ folosind relațiile:

$i(\neg p) = \neg i(p)$

$i(p \wedge q) = i(p) \wedge i(q)$

$i(p \vee q) = i(p) \vee i(q)$

$i(p \to q) = i(p) \to i(q)$

$i(p \leftrightarrow q) = i(p) \leftrightarrow i(q)$

Interpretările evaluează formulele propoziționale, conform semanticii conectivelor componente, atribuindu-le valori de *adevăr*.

Tabela de adevăr a unei formule propoziționale $U(p_1, p_2, ..., p_n) \in F_P$ corespunde evaluărilor formulei în toate cele 2^n interpretări.

Definiția 1.2. (concepte semantice)

Fie formula propozițională $U(p_1, p_2, ..., p_n)$.

1. O interpretare $i: \{p_1, p_2, ..., p_n\} \to \{F, T\}$ care evaluează formula U ca adevărată, $i(U)$=T, se numește ***model*** al formulei.
2. Formula U se numește ***consistentă (realizabilă)*** dacă și numai dacă are cel puțin un model, deci poate fi evaluată ca adevărată: $\exists i: \{p_1, p_2, ..., p_n\} \to \{F, T\}$ astfel încât $i(U)$=T.
3. Formula U se numește ***validă (tautologie)***, notație: $\models U$, dacă și numai dacă U este evaluată ca adevărată în orice interpretare, adică: $\forall i: \{p_1, p_2, ..., p_n\} \to \{F, T\}$, $i(U)$=T. Toate interpretările formulei U sunt modele ale formulei.
4. O interpretare $i: \{p_1, p_2, ..., p_n\} \to \{F, T\}$ care evaluează formula U ca falsă, $i(U)$=F, se numește ***anti-model*** al formulei.
5. Formula U se numește ***inconsistentă (nerealizabilă)*** dacă și numai dacă U nu are niciun model, adică U este interpretată totdeauna ca falsă: $\forall i: \{p_1, p_2, ..., p_n\} \to \{F, T\}$, $i(U)$=F.
6. Formula U se numește ***contingentă*** dacă și numai dacă este consistentă, dar nu este validă.

Observații:

- Dacă tabela de adevăr corespunzătoare formulei U conține doar valoarea de adevăr T, atunci U este o tautologie.
- Dacă tabela de adevăr corespunzătoare formulei U conține doar valoarea de adevăr F, atunci U este o formulă inconsistentă.
- Pentru o formulă propozițională, *modelele* corespund interpretărilor (liniilor din tabela de adevăr) în care *formula* este evaluată ca *adevărată*, iar *anti-modelele* corespund interpretărilor (liniilor din tabela de adevăr) în care *formula* este evaluată ca *falsă*.

efiniția 1.3.

Formula V este ***consecință logică*** a formulei U, notație: $U \models V$, dacă și numai dacă orice model al formulei U este model și pentru formula V, adică, $\forall i: F_P \to \{T, F\}$ astfel încât $i(U)$=T, are loc $i(V)$=T.

efiniția 1.4.

Formulele $U(p_1, p_2, ..., p_n) \in F_P$ și $V(p_1, p_2, ..., p_n) \in F_P$ sunt ***logic echivalente*** (au același înțeles), notație: $U \equiv V$, dacă și numai dacă tabelele lor de adevăr sunt identice, adică, $\forall i: F_P \to \{T, F\}$. $i(U)=i(V)$.

emplul 1.1.

Să se construiască tabelele de adevăr ale formulelor propoziționale:
$U(p,q,r) = (\neg p \vee q) \wedge (r \vee p)$, $V(p,q,r) = (\neg p \wedge r) \vee (q \wedge r) \vee (q \wedge p)$,
$W(p,q,r) = (p \uparrow (\neg p \wedge q)) \vee r$ și $Z(p,q,r) = p \wedge ((\neg q \vee r) \downarrow q)$.

p	q	r	$\neg p \vee q$	$r \vee p$	$U(p,q,r)$	$V(p,q,r)$	$W(p,q,r)$	$Z(p,q,r)$
T	T	T	T	T	T	T	T	F
T	T	F	T	T	T	T	T	F
T	F	T	F	T	F	F	T	F
T	F	F	F	T	F	F	T	F
F	T	T	T	T	T	T	T	F
F	T	F	T	F	F	F	T	F
F	F	T	T	T	T	T	T	F
F	F	F	T	F	F	F	T	F

- Se observă că doar interpretările i1, i2, i5 și i7 evaluează ca adevărată formula $U(p,q,r)$, deci acestea sunt toate modelele lui U. Formula U este contingentă.

  i1: $\{p,q,r\} \to \{T,F\}$, i1(p)=T, i1(q)=T, i1(r)=T și din tabelă i1(U)=T
  i2: $\{p,q,r\} \to \{T,F\}$, i2(p)=T, i2(q)=T, i2(r)=F și din tabelă i2(U)=T
  i5: $\{p,q,r\} \to \{T,F\}$, i5(p)=F, i5(q)=T, i5(r)=T și din tabelă i5(U)=T
  i7: $\{p,q,r\} \to \{T,F\}$, i7(p)=F, i7(q)=F, i7(r)=T și din tabelă i7(U)=T

- Interpretările i3, i4, i6 și i8 evaluează ca falsă formula $U(p,q,r)$, deci acestea sunt anti-modelele lui U.
  Are loc i3(U) = i4(U) = i6(U) = i8(U) = F.
- Formula $W(p,q,r)$ este o tautologie, toate cele 8 interpretări posibile sunt modelele sale.
- Formula $Z(p,q,r)$ este inconsistentă, aceasta fiind evaluată ca falsă în toate cele 8 interpretări.
- $U \equiv V$, deoarece formulele U și V au tabele de adevăr identice.
- $U \models \neg p \vee q$, deoarece în toate interpretările care evaluează formula U ca adevărată, adică: i1,i2,i5,i7, formula $\neg p \vee q$ este evaluată tot ca adevărată.

Observații:

1. $\models$ și $\equiv$ sunt metasimboluri utilizate pentru a exprima relații semantice între formule propoziționale.
2. Relația de echivalență logică, $\equiv$, satisface proprietățile de reflexivitate, simetrie și tranzitivitate.

1.3. Echivalențe logice în logica propozițională

Echivalențele logice următoare exprimă proprietăți ale conectivelor propoziționale. Acestea se numesc "legi" și sunt utilizate în procesul modificării sintactice a formulelor cu păstrarea semanticii acestora: înlocuirea unor formule cu formule logic echivalente.

- **Legile de simplificare**
  $\neg\neg U \equiv U$, $\quad U \to U \equiv T$
  $U \wedge \neg U \equiv F$ și $U \vee \neg U \equiv T$
  $T \wedge U \equiv U$ și $F \vee U \equiv U$
  $U \to T \equiv T$ și $U \to F \equiv \neg U$
  $T \to U \equiv U$ și $F \to U \equiv T$
  $U \leftrightarrow T \equiv U$ și $U \leftrightarrow F \equiv \neg U$
  $U \oplus T \equiv \neg U$ și $U \oplus F \equiv U$
  $U \leftrightarrow U \equiv T$ și $U \oplus U \equiv F$
- **Legile de idempotență**
  $U \wedge U \equiv U$ și $U \vee U \equiv U$
- **Legile de absorbție**
  $U \wedge (U \vee V) \equiv U$ și $U \vee (U \wedge V) \equiv U$
- **Legile de comutativitate**
  $U \wedge V \equiv V \wedge U$ și $U \vee V \equiv V \vee U$

$U \uparrow V \equiv V \uparrow U$ și $U \downarrow V \equiv V \downarrow U$

$U \leftrightarrow V \equiv V \leftrightarrow U$ și $U \oplus V \equiv V \oplus U$

- **Legile de asociativitate**
  $(U \wedge V) \wedge Z \equiv U \wedge (V \wedge Z)$ și $(U \vee V) \vee Z \equiv U \vee (V \vee Z)$
- **Legile distributivității**
  $U \wedge (V \vee Z) \equiv (U \wedge V) \vee (U \wedge Z)$
  $U \vee (V \wedge Z) \equiv (U \vee V) \wedge (U \vee Z)$
- **Legile lui DeMorgan**
  $\neg(U \wedge V) \equiv \neg U \vee \neg V$ și $\neg(U \vee V) \equiv \neg U \wedge \neg V$
- **Definirea conectivelor**
  $U \to V \equiv \neg U \vee V$ și $U \to V \equiv \neg(U \wedge \neg V)$
  $U \to V \equiv U \leftrightarrow (U \wedge V)$ și $U \to V \equiv V \leftrightarrow (U \vee V)$
  $U \leftrightarrow V \equiv (U \to V) \wedge (V \to U)$ și $U \oplus V \equiv \neg(U \to V) \vee \neg(V \to U)$
  $U \leftrightarrow V \equiv (U \vee V) \to (U \wedge V)$
  $U \vee V \equiv \neg(\neg U \wedge \neg V)$ și $U \wedge V \equiv \neg(\neg U \vee \neg V)$
  $U \vee V \equiv \neg U \to V$ și $U \wedge V \equiv \neg(U \to \neg V)$
  $\neg U \equiv U \uparrow U$ și $\neg U \equiv U \downarrow U$
  $U \vee V \equiv (U \uparrow U) \uparrow (V \uparrow V)$ și $U \wedge V \equiv (U \downarrow U) \downarrow (V \downarrow V)$
  $U \vee V \equiv (U \downarrow V) \downarrow (U \downarrow V)$ și $U \wedge V \equiv (U \uparrow V) \uparrow (U \uparrow V)$

Principiul dualității:

Pentru orice echivalență logică $U \equiv V$ care conține doar conectivele $\neg, \wedge, \vee, \uparrow, \downarrow$, există o altă echivalență logică $U' \equiv V'$, unde U' și V' sunt formule obținute din U, respectiv V, prin interschimbarea conectivelor logice duale: ($\wedge, \vee$), ($\uparrow, \downarrow$) și a valorilor de adevăr: T, F.

Se observă că legile definite mai sus și o parte din definirea conectivelor reprezintă perechi de echivalențe logice duale.

Conective duale: ($\wedge,\vee$), ($\uparrow,\downarrow$), ($\leftrightarrow,\oplus$).

Valori de adevăr duale: T și F.

Concepte duale: tautologie și formulă inconsistentă.

Definiția 1.5. (concepte semantice pentru mulțimi de formule)

1. O ***mulțime*** $\{U_1,U_2,...,U_n\}$ de formule se numește ***consistentă (realizabilă)*** dacă și numai dacă formula $U_1 \wedge U_2 \wedge ... \wedge U_n$ este consistentă, adică: $\exists i : F_P \to \{T,F\}$ astfel încât $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$=T. i se numește ***model*** al mulțimii $\{U_1,U_2,...,U_n\}$.
2. O ***mulțime*** $\{U_1,U_2,...,U_n\}$ de formule se numește ***inconsistentă (nerealizabilă, contradictorie)*** dacă și numai dacă formula $U_1 \wedge U_2 \wedge ... \wedge U_n$ este inconsistentă, adică $\forall i : F_P \to \{T,F\}$, $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$=F, i se numește ***anti-model*** al mulțimii $\{U_1,U_2,...,U_n\}$.
3. Formula V este ***consecință logică*** a mulțimii de formule $\{U_1,U_2,...,U_n\}$ și se notează $U_1,U_2,...,U_n \models V$, dacă și numai dacă $\forall i : F_P \to \{T,F\}$ astfel încât $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$=T, are loc $i(V)$=T. Formulele $U_1,U_2,...,U_n$ se numesc *premise*, *ipoteze*, *fapte*, iar V se numește *concluzie*.

Teorema 1.1.

Fie $S = \{U_1,U_2,...,U_n\}$ o mulțime de formule propoziționale.

1. *Dacă S este o mulțime consistentă, atunci $\forall j$, $1 \le j \le n$, $S - \{U_j\}$ este o mulțime consistentă.*
2. *Dacă S este o mulțime consistentă și V este o formulă validă, atunci mulțimea $S \cup \{V\}$ este consistentă.*
3. *Dacă S este o mulțime inconsistentă, atunci $\forall V \in F_P$ mulțimea $S \cup \{V\}$ este inconsistentă.*
4. *Dacă S este o mulțime inconsistentă și U_j este o formulă validă, unde $1 \le j \le n$, atunci mulțimea $S - \{U_j\}$ este inconsistentă.*

Demonstrații:

1. Dacă S este o mulțime consistentă ==>
   $\exists i$ - interpretare a.î. $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$=T ==>
   $\exists i$ - interpretare a.î. $i(U_1) = i(U_2) = ... = i(U_n)$=T ==>
   $\exists i$ - interpretare a.î. $\forall j, 1 \le j \le n$, $i(S - \{U_j\})$=T ==>
   $\forall j, 1 \le j \le n$, mulțimea $S - \{U_j\}$ este consistentă.
2. Dacă S este o mulțime consistentă ==>
   $\exists i$ -interpretare a.î. $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$=T
   și V este o formulă validă ==> $\forall j$ - interpretare, $j(V)$=T
   atunci

$\exists i, i(U_1 \wedge U_2 \wedge ... \wedge U_n \wedge V) = i(U_1 \wedge U_2 \wedge ... \wedge U_n) \wedge i(V) =$
$= T \wedge T = T$

Deci mulțimea $S \cup \{V\}$ este consistentă.

3. Dacă S mulțime inconsistentă și $V \in F_P$ ==>

$\forall i$ -interpretare, $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$ =F ==>

$\forall i$ -interpretare, are loc:

$i(U_1 \wedge U_2 \wedge ... \wedge U_n \wedge V) = i(U_1 \wedge U_2 \wedge ... \wedge U_n) \wedge i(V) =$
$= F \wedge i(V) = F$

Deci $\{U_1, U_2, ..., U_n, V\}$ este o mulțime inconsistentă.

4. Dacă S este o mulțime inconsistentă ==>

$\forall i$ -interpretare, $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$=F ==>

$\forall i$ -interpretare, $\exists k \in \{1,...n\}$, $i(U_k)$=F

și U_j este o formulă validă, unde $1 \leq j \leq n$ ==>

$\forall i$ -interpretare, $i(U_j)$=T,

atunci:

$\forall i$ -interpretare, $\exists k \in \{1,...n\}$, $k \neq j$, a.î. $i(U_k)$=F și $i(U_j)$=T

$\forall i$ -interpretare, $\exists k \in \{1,...n\}$, $k \neq j$ a.î. $U_k \in S - \{U_j\}$, $i(U_k)$=F

$\forall i$ -interpretare, $i(S - \{U_j\})$ = F

Deci mulțimea $S - \{U_j\}$ este inconsistentă.

Teorema 1.2.

Fie $U_1, U_2, ..., U_n, U, V$ formule propoziționale.

1. $\models U$ ***dacă și numai dacă*** $\neg U$ *este inconsistentă.*

O formulă este tautologie ***dacă și numai dacă*** *negația sa este o formulă inconsistentă.*

2. $U \models V$ ***dacă și numai dacă***

$\models U \to V$ ***dacă și numai dacă***

mulțimea $\{U, \neg V\}$ *este inconsistentă.*

3. $U \equiv V$ ***dacă și numai dacă*** $\models U \leftrightarrow V$.

4. $U_1, U_2, ..., U_n \models V$ ***dacă și numai dacă***

$\models U_1 \wedge U_2 \wedge ... \wedge U_n \to V$ ***dacă și numai dacă***

mulțimea $\{U_1, U_2, ..., U_n, \neg V\}$ *este inconsistentă.*

Demonstrații:

2. **"dacă $U \models V$ atunci $\models U \to V$"**

Conform definiției relației de consecință logică:

$\forall i$ - interpretare, cu $i(U)$=T, avem $i(V)$=T --- *ipoteza*

Pentru a demonstra $\models U \to V$ trebuie să evaluăm formula $U \to V$ în două cazuri:

a) dacă $i(U)$ =T atunci conform ipotezei avem:
$i(V)$=T și $i(U \to V) = i(U) \to i(V)$ =T $\to$ T=T

b) dacă $i(U)$ =F atunci
$i(U \to V) = i(U) \to i(V)$ =F $\to$ $i(V)$=T

Deci $\forall i$ -interpretare, $i(U \to V)$=T și astfel $U \to V$ este o tautologie.

"dacă $\{U, \neg V\}$ este o mulțime inconsistentă atunci $U \models V$"

Dacă mulțimea $\{U, \neg V\}$ este inconsistentă atunci $\forall i$ -interpretare $i(U \wedge \neg V)$ =F și sunt două cazuri:

a) $i(U)$ =F, caz care nu este relevant ;

b) $i(U)$ =T și $i(\neg V)$ =F, deci au loc $i(U)$ =T și $i(V)$ =T și astfel $U \models V$.

3. **"$U \equiv V$ dacă și numai dacă $\models U \leftrightarrow V$ "**

$U \equiv V$ dacă și numai dacă $\forall i$ -interpretare, $i(U)$=$i(V)$
daca și numai dacă
$\forall i$ -interpretare, $i(U \leftrightarrow V) = i(U) \leftrightarrow i(V) = i(U) \leftrightarrow i(U)$ =T
dacă și numai dacă $\models U \leftrightarrow V$.

4. **"dacă $\models U_1 \wedge U_2 \wedge ... \wedge U_n \to V$ atunci** mulțimea **$\{U_1, U_2, ..., U_n, \neg V\}$ este inconsistentă"**

$\forall i$ - interpretare, $i(U_1 \wedge U_2 \wedge ... \wedge U_n \to V)$ =T ==>

$\forall i$ - interpretare, $i(U_1 \wedge U_2 \wedge ... \wedge U_n) \to i(V)$ =T ==> două cazuri:

a) $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$ = F ==>
$i(U_1 \wedge U_2 \wedge ... \wedge U_n \wedge \neg V) = i(U_1 \wedge U_2 \wedge ... \wedge U_n) \wedge i(\neg V) =$
$=F \wedge i(\neg V) = F$

b) $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$ = T și $i(V)$ = T ==>
$i(U_1 \wedge U_2 \wedge ... \wedge U_n)$ = T și $i(\neg V)$ = F ==>
$i(U_1 \wedge U_2 \wedge ... \wedge U_n \wedge \neg V) = i(U_1 \wedge U_2 \wedge ... \wedge U_n) \wedge i(\neg V) =$
$=T \wedge F=F$

Deci $\forall i$ -interpretare, $i(U_1 \wedge U_2 \wedge ... \wedge U_n \wedge \neg V)$=F și astfel mulțimea $\{U_1, U_2, ..., U_n, \neg V\}$ este inconsistentă.

"dacă $\models U_1 \wedge U_2 \wedge ... \wedge U_n \to V$ atunci $U_1, U_2, ..., U_n \models V$"

$\forall i$ -interpretare, $i(U_1 \wedge U_2 \wedge ... \wedge U_n \to V)$ =T ==>

$\forall i$ -interpretare, $i(U_1 \wedge U_2 \wedge ... \wedge U_n) \to i(V)$ =T ==> două cazuri:

a) $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$ = F, caz care nu este relevant;

b) $i(U_1 \wedge U_2 \wedge ... \wedge U_n)$ = T şi $i(V)$ =T ==>

$U_1, U_2, ..., U_n \models V$ (conform def. relației de consecință logică).

Observație:

Afirmațiile 1, 2 şi 4 din teorema precedentă stau la baza metodelor de demonstrare prin respingere (metoda tabelelor semantice, rezoluția)

1.4. Forme normale în logica propozițiilor

Pentru a fi utilizate în diferite metode de demonstrare, formulele propoziționale trebuie transformate în formule care au o formă "normală" sau "canonică", păstrând semantica acestora.

Definiția 1.6.

1. Un ***literal*** este o variabilă propozițională sau negația sa.
2. O ***clauză*** este disjuncția unui număr finit de literali.
3. Un ***cub*** este conjuncția unui număr finit de literali.
4. ***Clauza vidă***, simbolizată prin $\Box$, este clauza fără literali, fiind singura clauză inconsistentă.
5. O formulă este în ***formă normală disjunctivă (FND)***, dacă aceasta este scrisă ca o disjuncție de cuburi: $\vee_{i=1}^{p} (\wedge_{j=1}^{q_i} l_{ij})$, unde l_{ij} sunt literali.
6. O formulă este în ***formă normală conjunctivă (FNC)***, dacă aceasta este scrisă ca o conjuncție de clauze: $\wedge_{i=1}^{n} (\vee_{j=1}^{m_i} l_{ij})$, unde l_{ij} sunt literali.

Exemplul 1.2.

- **clauze:** p, $q \vee s$, $p \vee \neg q \vee \neg s$
- **cuburi:** p, $q \wedge r$, $\neg p \wedge q \wedge s$
- **formule în formă normală disjunctivă:**
  p --- FND cu un cub format dintr-un singur literal
  $p \wedge q$ --- FND cu un cub format din doi literali
  $p \vee \neg q \vee r$ --- FND cu trei cuburi, fiecare cub format dintr-un literal
  $p \vee (q \wedge r) \vee (\neg p \wedge \neg r \wedge s)$ --- FND cu trei cuburi

- **formule în formă normală conjunctivă:**
  p --- FNC cu o clauză formată dintr-un singur literal
  $p \vee \neg q \vee r$ --- FNC cu o clauză formată din trei literali
  $p \wedge q$ --- FNC cu 2 clauze, fiecare clauză formată dintr-un literal
  $p \wedge (q \vee r) \wedge (\neg p \vee \neg r \vee s)$ --- FNC cu 3 clauze

Proprietate:

Fie mulțimea de literali $\{l_1, l_2 ..., l_n\}$. Următoarele afirmații sunt echivalente:

1. clauza $\vee_{i=1}^{n} l_i$ este o tautologie;
2. cubul $\wedge_{i=1}^{n} l_i$ este inconsistent;
3. în mulțimea $\{l_1, l_2 ..., l_n\}$ există cel puțin o pereche de literali opuşi, adică: $\exists i, j \in \{1,..,n\}$ astfel încât $l_j = \neg l_i$.

Exemplul 1.3.

- *clauza* $U = p \vee q \vee r \vee \neg p$ este o tautologie ($U \equiv$ T) deoarece $p, \neg p$ sunt literali opuşi;
- *cubul* $V = p \wedge q \wedge r \wedge \neg p$ este o formulă inconsistentă ($V \equiv$ F) deoarece conține doi literali opuşi: $p, \neg p$.

Teorema 1.3.

Orice formulă admite o formă normală conjunctivă şi o formă normală disjunctivă logic echivalente cu aceasta.

Demonstrația acestei teoreme constă în construirea unui algoritm de normalizare (furnizat în continuare) care realizează transformarea unei formule propoziționale în FNC şi/sau FND.

Algoritmul de normalizare reprezintă o succesiune de transformări (folosind echivalențele logice introduse în subcapitolul 1.3), care se aplică începând cu formula inițială şi păstrează echivalența logică, pentru a obține forme cu caracter "canonic".

Pas1:

- Înlocuirea formulelor de tip $X \to Y$ cu forma echivalentă $\neg X \vee Y$
- Înlocuirea formulelor de tip $X \leftrightarrow Y$ cu forma echivalentă: $(\neg X \vee Y) \wedge (\neg Y \vee X)$.

Pas2:

- Aplicarea legilor lui DeMorgan (se recomandă aplicarea dinspre exterior spre interior) ==> negația va preceda doar variabilele propoziționale.

- Eliminarea negațiilor multiple folosind echivalența logică: $\neg\neg X \equiv X$.

Pas3: Aplicarea legilor distributivității.

Pas4: Simplificarea formei obținute folosind alte echivalențe logice: legile de simplificare, legile absorbției, legile de idempotență.

Observație:
În urma aplicării acestui algoritm se poate obține ca primă formă normală fie FNC, fie FND. Transformarea în forma duală formei normale obținute anterior are loc prin aplicarea legilor distributivității.

Teorema 1.4.

1. *O formulă în forma normală conjunctivă (FNC) este tautologie dacă și numai dacă toate clauzele sale sunt tautologii.*
2. *O formulă în forma normală disjunctivă (FND) este inconsistentă dacă și numai dacă toate cuburile sale sunt inconsistente.*

Concepte duale:
clauză-cub, formă normală disjunctivă-formă normală conjunctivă.

Observații:

- Prima parte a teoremei precedente furnizează o *metodă directă* și *semantică* de rezolvare a problemei decizionale (verificarea dacă o formulă este tautologie) în logica propozițiilor.
- FND a unei formule propoziționale furnizează toate modelele formulei inițiale, care sunt interpretările ce evaluează, unul câte unul, cuburile componente ca adevărate.
- FNC a unei formule propoziționale furnizează toate anti-modelele formulei inițiale, care sunt interpretările ce evaluează, una câte una, clauzele componente ca false.
- Cele două formulări din teorema precedentă sunt afirmații duale.

Exemplul 1.4.

Aduceți la FND și FNC a doua axiomă a calculului propozițional:

$$A2 = ((p \to (q \to r)) \to ((p \to q) \to (p \to r)) .$$

Aplicăm algoritmul de normalizare pas cu pas astfel:

Pas1: înlocuirea conectivelor $\to$ din formulele interioare

$A2 \equiv ((p \to (\neg q \vee r)) \to ((\neg p \vee q) \to (\neg p \vee r))$

Pas1: înlocuirea conectivei principale $\to$

$A2 \equiv \neg(p \to (\neg q \vee r)) \vee ((\neg p \vee q) \to (\neg p \vee r))$

Pas1: înlocuirea celor două conective $\to$

$A2 \equiv \neg(\neg p \vee \neg q \vee r) \vee (\neg(\neg p \vee q) \vee \neg p \vee r)$

Pas2: aplicarea legilor lui DeMorgan

$A2 \equiv (p \wedge q \wedge \neg r) \vee (p \wedge \neg q) \vee \neg p \vee r$ --- FND cu 4 cuburi

Pas3: aplicarea legilor distributivității

$$A2 \equiv (p \vee \underline{p} \vee \underline{\neg p} \vee r) \wedge (q \vee \underline{p} \vee \underline{\neg p} \vee r) \wedge (\neg r \vee \underline{p} \vee \underline{\neg p} \vee r) \wedge$$
$$\wedge (\underline{p} \vee \neg q \vee \underline{\neg p} \vee r) \wedge (\underline{q} \vee \underline{\neg q} \vee \neg p \vee r) \wedge (\underline{\neg r} \vee \neg q \vee \neg p \vee \underline{r})$$

S-a obținut FNC având 6 clauze care sunt tautologii deoarece fiecare clauză conține o pereche de opuși (subliniați).
Conform primei afirmații din Teorema 1.4. s-a demonstrat că A2 este o tautologie.

Exemplul 1.5.

Folosind forma normală adecvată arătați că formula:
$U = (p \to (q \to r)) \wedge \neg(q \to (p \to r))$ este inconsistentă.
Vom utiliza FND conform Teoremei 1.4.
$U = (p \to (q \to r)) \wedge \neg(q \to (p \to r))$

(înlocuirea conectivelor $\to$ cele mai interioare)

$\equiv (p \to (\neg q \vee r)) \wedge \neg(q \to (\neg p \vee r))$

(înlocuirea celor două conective $\to$)

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

(aplicarea legii lui DeMorgan)

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

(aplicarea legii lui DeMorgan și eliminarea parantezelor redundante)

$\equiv (\neg p \vee \neg q \vee r) \wedge q \wedge p \wedge \neg r$ --- FNC cu 4 clauze

S-a obținut FNC și vom aplica legile distributivității pentru a obține FND.
$U \equiv (\underline{\neg p} \wedge q \wedge \underline{p} \wedge \neg r) \vee (\underline{\neg q} \wedge \underline{q} \wedge p \wedge \neg r) \vee (\underline{r} \wedge q \wedge p \wedge \underline{\neg r})$

Forma normală disjunctivă este formată din trei cuburi inconsistente, deoarece fiecare cub conține 2 literali opuși (subliniați). Conform Teoremei 1.4., concluzionăm că U este o formulă inconsistentă.

Exemplul 1.6.

Determinați forma normală disjunctivă și cea conjunctivă a formulei:
$U = (p \wedge q \to r) \to (p \to r) \wedge q$.

Aplicăm algoritmul de normalizare:

$U = (p \wedge q \to r) \to (p \to r) \wedge q$ (înlocuirea conectivei $\to$ principală)

$\equiv \neg(p \wedge q \to r) \vee (p \to r) \wedge q$ (înlocuirea celor două conective $\to$)

$\equiv \neg(\neg(p \wedge q) \vee r) \vee (\neg p \vee r) \wedge q$ (aplicarea legii lui DeMorgan)

$\equiv (p \wedge q \wedge \neg r) \vee (\neg p \vee r) \wedge q$ (aplicarea distributivității $\wedge$ față de $\vee$)

$\equiv (p \wedge q \wedge \neg r) \vee (\neg p \wedge q) \vee (r \wedge q)$ --- FND cu 3 cuburi

Modelele formulei U sunt interpretările care evaluează, unul câte unul, cuburile din FND ca adevărate.

Cubul: $p \wedge q \wedge \neg r$

i1: $\{p,q,r\} \to \{T,F\}$, i1(p)=T, i1(q)=T, i1(r)=F şi i1($p \wedge q \wedge \neg r$)=T

Cubul: $\neg p \wedge q$

i2: $\{p,q,r\} \to \{T,F\}$, i2(p)=F, i2(q)=T, i2(r)=T şi i2($\neg p \wedge q$)=T

i3: $\{p,q,r\} \to \{T,F\}$, i3(p)=F, i3(q)=T, i3(r)=F şi i3($\neg p \wedge q$)=T

Cubul: $r \wedge q$

i4: $\{p,q,r\} \to \{T,F\}$, i4(p)=T, i4(q)=T, i4(r)=T şi i4($r \wedge q$)=T

i5: $\{p,q,r\} \to \{T,F\}$, i5(p)=F, i5(q)=T, i5(r)=T şi i5($r \wedge q$)=T

Observăm că i2=i5. Modelele lui U sunt interpretările: i1, i2, i3 şi i4. Astfel are loc: i1(U) = i2(U) = i3(U) = i4(U) = T

Celelalte patru interpretări, din cele 8 posibile, evaluează formula U ca falsă, deci sunt anti-modelele lui U şi astfel U este o formulă contingentă.

Transformăm FND(U) în FNC, folosind legile distributivităţii:

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

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

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

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

$$\equiv \mathrm{T} \wedge \mathrm{T} \wedge (p \vee q \vee r) \wedge (p \vee q) \wedge (q \vee \neg p \vee r) \wedge (q \vee \neg p) \wedge (q \vee r) \wedge q \wedge$$

$$\wedge \mathrm{T} \wedge (\neg r \vee \neg p \vee q) \wedge \mathrm{T} \wedge (\neg r \vee q) \equiv q \text{ --- FNC}$$

S-au mai aplicat, ca ultime transformări pentru simplificare, legea absorbţiei: $a \wedge (a \vee b) \equiv a$ şi echivalenţa logică: $a \wedge \mathrm{T} \equiv a$

Deci FNC(U) = q şi observăm că modelele obţinute din FNC sunt într-adevăr cele patru interpretări care atribuie valoarea de adevăr "T" variabilei propoziţionale q, iar p şi r pot lua orice valori de adevăr.

Exemplul 1.7.

Aduceţi la FND şi FNC formula $U = (p \to (q \to r)) \to ((p \wedge s) \to r)$.

Aplicăm algoritmul de normalizare:

$$U = (p \to (q \to r)) \to ((p \wedge s) \to r) \equiv$$

$$\equiv (\neg p \vee (q \to r)) \to (\neg(p \wedge s) \vee r) \equiv$$

$$\equiv \neg(\neg p \vee (\neg q \vee r)) \vee (\neg(p \wedge s) \vee r) \equiv$$

$$\equiv (p \wedge q \wedge \neg r) \vee \neg p \vee \neg s \vee r \text{ (FND cu 4 cuburi)}$$

$$\equiv (p \vee \neg p \vee \neg s \vee r) \wedge (q \vee \neg p \vee \neg s \vee r) \wedge (\neg r \vee \neg p \vee \neg s \vee r) \equiv$$

$$\equiv T \wedge (q \vee \neg p \vee \neg s \vee r) \wedge T \equiv q \vee \neg p \vee \neg s \vee r \text{ (FNC cu o clauză)}$$

Această ultimă formă echivalentă a formulei U este FNC (având o clauză) care nu este tautologie, deci formula U nu este tautologie.

Clauza $q \vee \neg p \vee \neg s \vee r$ din forma normală conjunctivă furnizează unicul anti-model al formulei U.

i: $\{p,q,r,s\} \to \{T,F\}$, i(p)=T, i(q)=F, i(r)=F, i(s)=T şi astfel i(U)=F.

Are loc:

$$\mathrm{i}(U) = \mathrm{i}((p \to (q \to r)) \to ((p \wedge s) \to r)) =$$

$$(\mathrm{T} \to (\mathrm{F} \to \mathrm{F})) \to ((\mathrm{T} \wedge \mathrm{T}) \to \mathrm{F}) = (\mathrm{T} \to \mathrm{T}) \to (\mathrm{T} \to \mathrm{F}) = \mathrm{T} \to \mathrm{F} = \mathrm{F}$$

Din forma normală disjunctivă rezultă că formula U este o formulă contingentă, având ca modele interpretările care evaluează ca adevărate, unul câte unul, cuburile din FND.

Deoarece din cele 2^4=16 interpretări posibile, există doar un anti-model (i), toate celelalte interpretări sunt modele ale lui U. Astfel, nu mai este nevoie de obţinerea modelelor din cuburile FND.

1.5. Sistemul axiomatic (deductiv, formal) al calculului propoziţional

În acest subcapitol sunt prezentate sistemele deductive propuse de Hilbert. Cel mai cunoscut şi utilizat este sistemul axiomatic $P = (\Sigma_P, F_P, A_P, R_P)$ unde:

- $\Sigma_P = Var\_propoz \cup Conective \cup \{(,)\}$ - *alfabetul*;

  $Var\_propoz = \{p_1, p_2, ...\}$ - *mulţime* numărabilă de *variabile propoziţionale*;

  $Conective = \{\neg, \wedge, \vee, \to, \leftrightarrow\}$
- F_P - *mulţimea formulelor bine formate*, definită în subcapitolul 1.1;
- A_P ={A1, A2, A3} - *mulţimea axiomelor* calculului propoziţional;

  A1: $U \to (V \to U)$

A2: $((U \to (V \to Z)) \to ((U \to V) \to (U \to Z))$

A3: $(U \to V) \to (\neg V \to \neg U)$

A1, A2 şi A3 sunt de fapt scheme axiomatice, unde U, V şi Z pot fi formule arbitrare, generând astfel o infinitate de axiome.

- $R_P = \{mp\}$ - ***mulţimea regulilor de inferenţă (deducţie)***

  Regula ***modus ponens*** este simbolizată astfel: $U, U \to V \vdash_{mp} V$ şi are semnificaţia: "din faptele U şi $U \to V$ se deduce V ".

Definiţia 1.7.

Fie formulele propoziţionale $U_1, U_2, ..., U_n$ numite ipoteze şi V o formulă propoziţională. Spunem că V ***este deductibilă (derivabilă) din*** $U_1, U_2, ..., U_n$, notaţie: $U_1, U_2, ..., U_n \vdash V$, dacă există o secvenţă de formule propoziţionale $(f_1, f_2, ..., f_m)$ astfel încât $f_m = V$ şi $\forall i \in \{1, ..., m\}$ avem a) sau b) sau c).

a) $f_i \in A_P$ (axiomă a calculului propozitional);

b) $f_i \in \{U_1, U_2, ..., U_n\}$ (formulă ipoteză);

c) $f_{i1}, f_{i2} \vdash_{mp} f_i$, $i1 < i$ şi $i2 < i$ (formula s-a obţinut prin inferenţă, aplicând regula modus ponens, din două formule existente deja în secvenţă)

Secvenţa $(f_1, f_2, ..., f_m)$ se numeşte ***deducţia*** **lui** V **din** $U_1, U_2, ..., U_n$.

Observaţie:

"$\vdash$" este relaţia de derivabilitate clasică discutată pe larg în subcapitolul 1.6.

Definiţia 1.8.

O formulă $U \in F_P$, astfel încât $\emptyset \vdash U$ (notaţie: $\vdash U$) se numeşte ***teoremă***.

Observaţie:

Teoremele sunt formule deductibile doar din axiome, prin aplicarea regulii de inferenţă *modus ponens*.

Exemplul 1.8.

Să se demonstreze că are loc: $U, U \to V, V \to Z \vdash Z$ folosind definiţia deducţiei.

Se construieşte secvenţa de formule (f1, f2, f3, f4, f5) astfel:

f1: U --- formulă ipoteză

f2: $U \to V$ --- formulă ipoteză

f1, f2 $\vdash_{mp} V$

f3: V

f4: $V \to Z$ --- formulă ipoteză

f3, f4 $\vdash_{mp} Z$

f5: Z

Se observă că f1, f2 şi f4 sunt formulele iniţiale (ipoteze), iar f3 şi f5 se obţin prin aplicarea regulii de inferenţă modus ponens. Conform definiţiei deducţiei s-a demonstrat astfel că are loc deducţia din enunţ.

Exemplul 1.9.

Să se demonstreze că are loc: $\vdash U \to U$ construind deducţia formulei $U \to U$ doar din axiomele calculului propoziţional.

Se construieşte secvenţa de formule (f1, f2, f3, f4, f5) astfel:

f1: $U \to ((U \to U) \to U)$

- obţinută din schema axiomatică A1: $U \to (V \to U)$ prin înlocuirea lui V cu $U \to U$;

f2: $(U \to ((U \to U) \to U)) \to ((U \to (U \to U)) \to (U \to U))$

- obţinută din A2: $((U \to (V \to Z)) \to ((U \to V) \to (U \to Z))$ după înlocuirile: V cu $U \to U$ şi Z cu U;

f1, f2 $\vdash_{mp}$ $(U \to (U \to U)) \to (U \to U)$

f3: $(U \to (U \to U)) \to (U \to U)$

f4: $U \to (U \to U)$

- obţinută din A1: $U \to (V \to U)$ prin înlocuirea lui V cu U;

f3, f4 $\vdash_{mp}$ $U \to U$

f5: $U \to U$

Secvenţa de formule (f1, f2, f3, f4, f5) este deducţia formulei $U \to U$ din axiomele calculului propoziţional, deci $U \to U$ este teoremă.

Exemplul 1.10.

Să se demonstreze că are loc: $\neg U \vdash U \to V$ folosind definiţia deducţiei.

Se construieşte secvenţa de formule (f1, f2, f3, f4, f5) astfel:

f1: $\neg U$ --- formulă ipoteză

f2: $\neg U \to (\neg V \to \neg U)$

- obţinută din A1: $U \to (V \to U)$ prin înlocuirea lui U cu $\neg U$ şi a lui V cu $\neg V$;

$f1, f2 \vdash_{mp} \neg V \to \neg U$

$f3$: $\neg V \to \neg U$

$f4$: $(\neg V \to \neg U) \to (U \to V)$

- obținută din schema axiomatică A3: $(U \to V) \to (\neg V \to \neg U)$ prin înlocuirea lui U cu $\neg V$ și a lui V cu $\neg U$;

$f3, f4 \vdash_{mp} U \to V$

$f5$: $U \to V$

Secvența (f1, f2, f3, f4, f5) este deducția formulei $U \to V$ din $\neg U$ și axiomele calculului propozițional.

Observație:

Construirea deducției (demonstrației) unei teoreme din axiome sau a unei concluzii dintr-o mulțime de ipoteze și axiome, conform Definiției 1.7, este o *metodă directă și sintactică* (nu foarte eficientă și fără posibilitatea de implementare) de rezolvare a problemelor decizionale ale calculului propozițional.

Sisteme axiomatice (deductive) Hilbert

Există un grup de sisteme deductive numite sisteme Hilbert care au *modus ponens* ca unică regulă de inferență. Diferența dintre acestea constă în axiomele care le definesc, bazate pe anumite conective considerate primitive.

Varianta cea mai apropiată de sistemul axiomatic introdus anterior înlocuiește doar axioma A3 cu A3': $(\neg V \to \neg U) \to ((\neg V \to U) \to V)$, celelalte componente ale cvadruplului rămânând neschimbate.

Cele două sisteme deductive prezentate mai sus au ca și conectivă binară de bază implicația.

Un alt sistem Hilbert propus conține axiome care utilizează și disjuncția ca și conectivă de bază în procesul deductiv. Corespunzător sunt introduse axiomele:

Axioma 1: $U \vee U \to U$

Axioma 2: $U \to U \vee V$

Axioma 3: $U \vee V \to V \vee U$

Axioma 4: $(V \to Z) \to (U \vee V \to U \vee Z)$

O altă variantă de sistem axiomatic pentru calculul propozițional conține o singură axiomă, numită axioma Meredith:

$$((((U \to V) \to (\neg Z \to \neg X)) \to Z) \to Y) \to ((Y \to U) \to (X \to U))$$

Toate aceste sisteme Hilbert sunt echivalente deoarece axiomele fiecărui sistem pot fi demonstrate ca fiind teoreme în celelalte sisteme.

În continuare, pe tot parcursul lucrării, vom utiliza sistemul axiomatic P.

1.6. Relația și operatorul de derivabilitate clasică

Raționamentul modelat de logicile clasice (calculul propozițional, calculul predicatelor de ordinul I), precum și de alte logici nestandard, dar monotone: logicile modale, logicile temporale, logicile multivalente, este *pur deductiv*, adică adăugarea de noi premise nu va duce la inconsistențe în mulțimea concluziilor.

Modul de raționare în logica propozițiilor, precum și rezultatele procesului de raționament sunt modelate de relația și operatorul de derivabilitate (inferență).

Logica propozițională este caracterizată de o ***relație de derivabilitate (deducție, inferență)*** $\vdash \subseteq \mathcal{P}(F_P) \times F_P$, care satisface următoarele proprietăți:

Fie $S, R \subseteq F_P$ și $U, V_i \in F_P$, $\forall i \in I \subset N$.

(M1) ***reflexivitate:*** dacă $U \in S$ atunci $S \vdash U$.

(M2) ***eliminare:*** dacă $S \vdash V_i, \forall i \in I$ și $S \cup \{V_i \mid i \in I\} \vdash U$ atunci $S \vdash U$.

(M3) ***monotonie:*** dacă $R \vdash U$ și $R \subseteq S$ atunci $S \vdash U$.

Acestei relații de inferență i se poate asocia ***operatorul de derivabilitate monotonă:*** **Th**: $\mathcal{P}(F_P) \to \mathcal{P}(F_P)$ definit prin proprietățile:

(Th1) ***incluziune:*** $S \subseteq Th(S)$.

(Th2) ***idempotență:*** $Th(S)=Th(Th(S))$.

(Th3) ***monotonie:*** dacă $R \subseteq S$ atunci $Th(R) \subseteq Th(S)$.

Operatorul **Th** definește închiderea deductivă a unei mulțimi de premise, iar Th(S) reprezintă mulțimea ***consecințelor sintactice monotone***, derivate din premisele S utilizând sistemul axiomatic P.

Legătura dintre notația relațională și cea operațională a mecanismului inferențial este următoarea:

$S \vdash U$ *dacă și numai dacă* $U \in Th(S)$.

Grupul de proprietăți (M1)+(M2)+(M3) este echivalent cu grupul de proprietăți (Th1)+(Th2)+(Th3).

Reflexivitatea este reprezentată în abordarea operațională prin ***incluziune*** și are următoarea semnificație: premisele sunt derivate în mod trivial.

Proprietățile (M2) și (Th2) nu sunt echivalente, relația dintre ele fiind următoarea: dacă proprietatea de incluziune este satisfăcută, atunci idempotența este un caz particular al proprietății de eliminare.

Proprietatea de eliminare se exprimă în notaţie operaţională prin: „*dacă* S⊆R⊆Th(S) *atunci* Th(R) ⊆Th(S)" şi aceasta asigură faptul că utilizarea unor concluzii derivate deja, păstrează rezultatele procesului inferenţial în limita închiderii deductive a mulţimii iniţiale de premise.

Proprietatea de monotonie afirmă faptul că mulţimea concluziilor derivate (consecinţelor monotone) creşte odată cu creşterea mulţimii ipotezelor (premiselor), adică adăugarea de noi premise nu va duce la inconsistenţe în mulţimea concluziilor. Această proprietate situează un sistem logic în categoria logicilor monotone sau nu.

Proprietăţi utile în procesul deductiv (inferenţial) modelat de logica propoziţională:

1. ***Tranzitivitatea (caz particular al proprietăţii de eliminare)***
   - *notaţie relaţională*:
     dacă $S \vdash U$ şi $\{U\} \vdash V$ atunci $S \vdash V$.
   - *notaţie operaţională*:
     dacă $U \in Th(S)$ şi $V \in Th(\{U\})$ atunci $V \in Th(S)$.
2. ***Conjuncţia în concluzii („şi" la dreapta)***
   - *notaţie relaţională*:
     dacă $S \vdash U$ şi $S \vdash V$ atunci $S \vdash U \wedge V$.
   - *notaţie operaţională*:
     dacă $U \in Th(S)$ şi $V \in Th(S)$ atunci $U \wedge V \in Th(S)$.
3. ***Disjuncţia în premise („sau" la stânga)***
   - *notaţie relaţională*:
     dacă $S \cup \{U\} \vdash Z$ şi $S \cup \{V\} \vdash Z$ atunci $S \cup \{U \vee V\} \vdash Z$.
   - *notaţie operaţională*:
     $Th(S \cup \{U\}) \cap Th(S \cup \{V\}) \subseteq Th(S \cup \{U \vee V\})$.
4. ***Demonstraţia pe cazuri***
   - *notaţie relaţională*:
     dacă $S \cup \{U\} \vdash V$ şi $S \cup \{\neg U\} \vdash V$ atunci $S \vdash V$.
   - *notaţie operaţională*: $Th(S \cup \{U\}) \cap Th(S \cup \{\neg U\}) \subseteq Th(S)$.
5. ***Reciprocitatea***
   - *notaţie operaţională*:
     dacă $R \subseteq Th(S)$ şi $S \subseteq Th(R)$ atunci $Th(S)=Th(R)$
6. ***Păstrarea consistenţei:***
   - dacă S este o mulţime consistentă de formule propoziţionale, atunci Th(S) este consistentă.

1.7. Teorema de deducţie şi inversa sa

Teorema 1.5. (teorema de deducţie)
Dacă $U_1,...,U_{n-1},U_n \vdash V$ *atunci* $U_1,...,U_{n-1} \vdash U_n \to V$.

Teorema 1.6. (inversa teoremei de deducţie)
Dacă $U_1,...,U_{n-1} \vdash U_n \to V$ *atunci* $U_1,...,U_{n-1},U_n \vdash V$.

Aplicând de *n* ori teorema de deducţie şi reciproca sa obţinem:
$U_1,...,U_{n-1},U_n \vdash V$ dacă şi numai dacă
$U_1,...,U_{n-1} \vdash U_n \to V$ dacă şi numai dacă
$U_1,...,U_{n-2} \vdash U_{n-1} \to (U_n \to V)$ dacă şi numai dacă
...
$U_1 \vdash U_2 \to (... \to (U_{n-1} \to (U_n \to V))...)$ dacă şi numai dacă
$\vdash U_1 \to (U_2 \to (... \to (U_{n-1} \to (U_n \to V))...))$

Se poate obţine o *metodă directă şi sintactică de demonstrare a teoremelor* care conţin preponderent implicaţia ca şi conectivă, astfel:

Pas1: Găsirea deducţiei de plecare, obţinută prin aplicarea inversei teoremei de deducţie asupra formulei ce trebuie demonstrată că este teoremă.

Observaţie:
În general deducţia de plecare se poate "ghici" uşor, fără a fi nevoie de aplicarea inversei teoremei de deducţie, astfel: premisele implicaţiilor din dreapta, în ordinea inversă efectuării acestora, se vor muta la stânga metasimbolului "$\vdash$" ca şi formule ipoteză.

Pas2: Demonstrarea deducţiei obţinute la pasul anterior (folosind Definiţia 1.7.).

Pas3: Aplicarea teoremei de deducţie asupra deducţiei demonstrate, pentru a muta spre dreapta formulele ipoteză şi a obţine teorema dorită.

Consecinţe ale teoremei de deducţie:

1. $\vdash U \to ((U \to V) \to V)$
2. $\vdash (U \to V) \to ((V \to Z) \to (U \to Z))$ legea silogismului
3. $\vdash (U \to (V \to Z)) \to (V \to (U \to Z))$ legea permutării premiselor
4. $\vdash (U \to (V \to Z)) \to (U \wedge V \to Z)$ legea reunirii premiselor
5. $\vdash (U \wedge V \to Z) \to (U \to (V \to Z))$ legea separării premiselor

Demonstrație consecințele 1 și 2.

1. Pornim de la regula de inferență *modus ponens*: $U, U \to V \vdash_{mp} V$.

   Aplicând teorema de deducție obținem: $U \vdash (U \to V) \to V$ și continuând cu încă o aplicare a teoremei de deducție obținem: $\vdash U \to ((U \to V) \to V)$.

2. Utilizând Exemplul 1.8. pornim de la deducția: $U, U \to V, V \to Z \vdash Z$

   Aplicăm teorema de deducție ==> $U \to V, V \to Z \vdash U \to Z$

   Aplicăm teorema de deducție ==> $U \to V \vdash (V \to Z) \to (U \to Z)$

   Aplicăm teorema de deducție ==>

$$\vdash (U \to V) \to ((V \to Z) \to (U \to Z))$$

Legile enunțate anterior sub forma unor teoreme au exprimări echivalente sub forma unor deducții, care pot reprezenta *reguli de „inferență"* utilizate în procesul deductiv:

- $U, U \to V, V \to Z \vdash Z$, regula (legea) silogismului
- $U \to (V \to Z) \vdash V \to (U \to Z)$, regula (legea) permutării premiselor
- $U \to (V \to Z) \vdash U \wedge V \to Z$, regula (legea) reunirii premiselor
- $U \wedge V \to Z \vdash U \to (V \to Z)$, regula (legea) separării premiselor.

Exemplul 1.11.

Folosind teorema de deducție și inversa sa demonstrați că are loc:

$\vdash (p \to r) \to ((p \wedge r \to q) \to (p \to q))$

Aplicăm inversa teoremei de deducție de 3 ori pentru a obține deducția de la care ar trebui să pornim:

$\vdash (p \to r) \to ((p \wedge r \to q) \to (p \to q))$ dacă și numai dacă

$p \to r \vdash (p \wedge r \to q) \to (p \to q)$ dacă și numai dacă

$p \to r, p \wedge r \to q \vdash p \to q$ dacă și numai dacă

$p \to r, p \wedge r \to q, p \vdash q$

Pentru a demonstra ultima deducție utilizăm Definiția 1.7. construind secvența de formule (f1, f2, f3, f4, f5, f6) astfel:

f1: p --- formulă ipoteză

f2: $p \to r$ --- formulă ipoteză

f1, f2 $\vdash_{mp} r$

f3: r

Se aplică proprietatea de conjuncție a concluziilor: p și r (subcapitolul 1.6), deci considerăm formula derivată f4.

f4: f1 $\wedge$ f3= $p \wedge r$

f5: $p \wedge r \to q$ --- formulă ipoteză

f4, f5 $\vdash_{mp} q$

f6: q

Secvența (f1, f2, f3, f4, f5, f6) este deducția lui q din ipotezele $p \to r, p \wedge r \to q, p$.

Se pornește de la deducția: $p \to r, p \wedge r \to q, p \vdash q$ și se va aplica teorema de deducție de 3 ori.

Există 3!=6 astfel de posibilități (de a muta cele 3 ipoteze ale deducției la dreapta metasimbolului $\vdash$) demonstrând astfel 6 teoreme: T1, T2, T3, T4, T5, T6, dintre care T1 este cea cerută în enunțul problemei:

1. ordinea de mutare a ipotezelor deducției la dreapta este:

$p,\ p \wedge r \to q,\ p \to r$

$p \to r, p \wedge r \to q, p \vdash q$ dacă și numai dacă

$p \to r, p \wedge r \to q \vdash p \to q$ dacă și numai dacă

$p \to r \vdash (p \wedge r \to q) \to (p \to q)$ dacă și numai dacă

$\vdash T1 = (p \to r) \to ((p \wedge r \to q) \to (p \to q))$ --- teorema din enunț.

2. ordinea de mutare a ipotezelor deducției la dreapta este:

$p,\ p \to r,\ p \wedge r \to q$

$p \to r, p \wedge r \to q, p \vdash q$ dacă și numai dacă

$p \to r, p \wedge r \to q \vdash p \to q$ dacă și numai dacă

$p \wedge r \to q \vdash (p \to r) \to (p \to q)$ dacă și numai dacă

$\vdash T2 = (p \wedge r \to q) \to ((p \to r) \to (p \to q))$

3. ordinea de mutare a ipotezelor deducției la dreapta este:

$p \to r,\ p,\ p \wedge r \to q$

$p \to r, p \wedge r \to q, p \vdash q$ dacă și numai dacă

$p \wedge r \to q, p \vdash (p \to r) \to q$ dacă și numai dacă

$p \wedge r \to q \vdash p \to ((p \to r) \to q)$ dacă și numai dacă

$\vdash T3 = (p \wedge r \to q) \to (p \to ((p \to r) \to q))$

4. *ordinea de mutare a ipotezelor deducției la dreapta este:*
$p \to r,\ p \wedge r \to q,\ p$

$p \to r, p \wedge r \to q, p \vdash q$ dacă şi numai dacă

(se aplică de 3 ori teorema de deducție)

$\vdash T4 = p \to ((p \wedge r \to q) \to ((p \to r) \to q))$

5. *ordinea de mutare a ipotezelor deducției la dreapta este:*
$p \wedge r \to q,\ p,\ p \to r$

$p \to r, p \wedge r \to q, p \vdash q$ dacă şi numai dacă

(se aplică de 3 ori teorema de deducție)

$\vdash T5 = (p \to r) \to (p \to ((p \wedge r \to q) \to q))$

6. *ordinea de mutare a ipotezelor deducției la dreapta este:*
$p \wedge r \to q,\ p \to r,\ p$

$p \to r, p \wedge r \to q, p \vdash q$ dacă şi numai dacă

(se aplică de 3 ori teorema de deducție)

$\vdash T6 = p \to ((p \to r) \to ((p \wedge r \to q) \to q))$

1.8. Proprietățile logicii propozițiilor

Teoremele din acest subcapitol introduc rezultate teoretice referitoare la proprietățile logicii propozițiilor.

Teorema 1.7. (teorema 1 de compactitate)
O mulțime infinită de formule propoziționale are un model dacă şi numai dacă orice submulțime finită a sa are un model.

Teorema 1.8.
Fie $S=\{U_1,U_2,\ldots,U_m,\ldots\}$ *o mulțime infinită de formule propoziționale.*

1. *S este inconsistentă dacă şi numai dacă* $\exists k \in N^*$, *astfel încât mulțimea* $\{U_1,U_2,\ldots,U_k\}$ *este inconsistentă.*
2. *S este consistentă dacă şi numai dacă*
$\{U_1\}$ *este consistentă şi*
$\{U_1,U_2\}$ *este consistentă şi*
...
$\{U_1,U_2,\ldots,U_m\}$ *este consistentă şi*
.

Observații:

- O mulțime infinită de formule propoziționale este inconsistentă dacă şi numai dacă are o submulțime finită inconsistentă, fapt verificabil într-un număr finit de paşi.
- O mulțime infinită de formule propoziționale este consistentă dacă şi numai dacă toate submulțimile sale (un număr infinit) sunt consistente, dar acest fapt nu este verificabil într-un număr finit de paşi.

Teorema 1.9. (teorema 2 de compactitate)
O formulă propozițională V este consecință logică a mulțimii infinite S de formule ($S \models V$) dacă şi numai dacă există o submulțime finită a lui S: $\{U_1,U_2,\ldots,U_n\} \subset S$ *astfel încât* $U_1,U_2,\ldots,U_n \models V$.

Teoremele de corectitudine şi completitudine stabilesc legătura dintre semantica logicii propozițiilor şi sintaxa introdusă prin sistemul axiomatic P.

Teorema 1.10. (teorema de corectitudine a calculului propozițional)
Validitatea sintactică implică validitatea semantică:

dacă $\vdash U$ *atunci* $\models U$ *(o teoremă este o tautologie).*

Teorema 1.11. (teorema de completitudine a calculului propozițional)
Validitatea semantică implică validitatea sintactică:

dacă $\models U$ *atunci* $\vdash U$ *(o tautologie este o teoremă).*

Teorema 1.12. (teorema de corectitudine şi completitudine)

1. $\vdash U$ *dacă şi numai dacă* $\models U$.
2. generalizare:
$U_1,U_2,\ldots,U_n \vdash V$ *dacă şi numai dacă* $U_1,U_2,\ldots,U_n \models V$.

Consecințe ale acestei teoreme sunt următoarele proprietăți ale logicii propozițiilor:

1) ***Logica propozițiilor este necontradictorie:***

„ Nu pot avea loc simultan $\vdash U$ şi $\vdash \neg U$ ".

Demonstrație:
Proprietatea este echivalentă, conform teoremei de corectitudine, cu afirmația: „Nu pot avea loc simultan $\models U$ şi $\models \neg U$.", adică o formulă şi negația sa nu pot fi adevărate simultan, ceea ce este evident adevărat.

2) ***Logica propozițiilor este coerentă***:

„Nu orice formulă propozițională este teoremă."

Demonstrație:
Conform Teoremei 1.10. proprietatea este echivalentă cu afirmația: „Nu orice formulă propozițională este tautologie.", care este adevărată deoarece s-a demonstrat în acest capitol că există formule propoziționale consistente și formule inconsistente.

3) ***Logica propozițiilor este decidabilă***: se poate decide dacă o formulă propozițională este sau nu teoremă (tautologie).

Demonstrație:
Construirea tabelei de adevăr asociate unei formule propoziționale este o *metodă directă*, bazată pe *considerente semantice*, utilizată pentru a decide dacă formula este sau nu tautologie.

Problemele decizionale din logica propozițiilor:

- O formulă propozițională U este tautologie (teoremă)?
- O formulă propozițională (*conjectură*) este consecință logică (sintactică) a unei mulțimi de formule propoziționale (*axiome* și *ipoteze*)?

Scopul metodelor de demonstrare (metoda tabelelor semantice, calculul secvențelor, rezoluția) prezentate în capitolele următoare, este să ofere proceduri eficiente de rezolvare a acestor probleme decizionale.

2. LOGICA PREDICATELOR DE ORDINUL I

Logica predicatelor de ordinul I este un sistem formal, deductiv, utilizat în formalizarea raționamentului valid din domenii diverse: matematică, filozofie, lingvistică, informatică. Acest sistem logic extinde logica propozițiilor prin utilizarea unor noi categorii sintactice: *constante, variabile, simboluri de funcții* și *predicate, cuantificatori* aplicați asupra variabilelor aparținând unui domeniu de interpretare.

Referințele bibliografice utilizate pentru noțiunile și rezultatele teoretice prezentate în acest capitol sunt lucrările [1, 2, 10, 11, 13, 14, 15, 16, 17, 20, 28, 29, 32, 40, 42, 43, 44, 45].

2.1. Sistemul axiomatic (deductiv, formal) al logicii predicatelor de ordinul I

În reprezentarea cunoștințelor, se obișnuiește să se reprezinte categoriile sintactice de bază în termeni care se referă explicit la universul ce va fi reprezentat (modelat). Conceptualizarea universului presupune stabilirea: *obiectelor* care fac parte din domeniul de interpretare D precum și a relațiilor dintre obiecte și variabile. Aceste relații sunt reprezentate prin *funcții* și *predicate*. Formalizarea cunoștințelor din universul studiat necesită definirea sintaxei limbajului de reprezentare, în acest caz logica predicatelor de ordinul I.

Sistemul deductiv atașat logicii predicatelor de ordinul I conține alfabetul sistemului, mulțimea formulelor bine formate, mulțimea axiomelor logice și mulțimea regulilor de inferență utilizate în procesul deductiv.

$\mathrm{Pr} = (\Sigma_{\mathrm{Pr}}, F_{\mathrm{Pr}}, A_{\mathrm{Pr}}, R_{\mathrm{Pr}})$ unde:

- $\Sigma_{\mathrm{Pr}} = Var \cup Const \cup (\cup_{j=1}^{n} F_j) \cup (\cup_{j=1}^{m} P_j) \cup P_0 \cup Conective \cup Cuantif$ - ***alfabetul***

  $Var = \{x, y, z, \ldots\}$ - **mulțimea *simbolurilor de variabile***;
  $Const = \{a, b, c, \ldots\}$ - **mulțimea *constantelor***;

  $F_i = \{f \mid f : D^i \to D\}$ - **mulțimea *simbolurilor de funcții* de aritate "*i*"**;

  $P_i = \{p \mid p : D^i \to \{\mathrm{T}, \mathrm{F}\}\}$ - **mulțimea *simbolurilor de predicate* de aritate "*i*"**. $i \geq 1$, care sunt de obicei reguli de conectare între variabile şi constante.

$P_0 = \{p, q, r, ...\} \cup \{T, F\}$ - mulţimea variabilelor propoziţionale şi a valorilor de adevăr: T şi F.

Conective = $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$;

Cuantif = { $\forall$ (cuantificatorul universal), $\exists$ (cuantificatorul existenţial)}

- ***TERM*** - **mulţimea *termenilor*** este definită astfel:
  - $Var \subset TERM$; $Const \subset TERM$;
  - dacă $f \in F_k$ şi $t_1, ..., t_k \in TERM$ atunci $f(t_1, ..., t_k) \in TERM$
- ***ATOM* - *mulţimea formulelor atomice (atomilor)*** este definită astfel:
  - $T, F \in ATOM$;
  - dacă $p \in P_k$ şi $t_1, ..., t_k \in TERM$ atunci $p(t_1, ..., t_k) \in ATOM$.
- ***Literal*** = un atom sau negaţia sa.

- F_{Pr} - **mulţimea *formulelor bine formate (corect construite)*** este definită astfel:
  - $ATOM \subset F_{Pr}$;
  - dacă $U \in F_{Pr}$ şi $x \in Var$ astfel încât x nu se află deja sub incidenţa unui cuantificator, atunci $(\forall x)U(x) \in F_{Pr}$ şi $(\exists x)U(x) \in F_{Pr}$;
  - dacă $U, V \in F_{Pr}$ astfel încât U şi V nu conţin aceeaşi variabilă atât liberă (nu este sub incidenţa unui cuantificator), cât şi legată (se află sub incidenţa unui cuantificator), atunci:

    $\neg U \in F_{Pr}, U \wedge V \in F_{Pr}, U \vee V \in F_{Pr}, U \rightarrow V \in F_{Pr}, U \leftrightarrow V \in F_{Pr}$.

Definiţia 2.1.

Variabilele care se află sub incidenţa unui cuantificator se numesc ***variabile legate***, în caz contrar acestea se numesc ***variabile libere***.

- A_{Pr} = {A1, A2, A3, A4, A5} - **mulţimea *axiomelor***

  A1: $U \rightarrow (V \rightarrow U)$

  A2: $((U \rightarrow (V \rightarrow Z)) \rightarrow ((U \rightarrow V) \rightarrow (U \rightarrow Z))$

  A3: $(U \rightarrow V) \rightarrow (\neg V \rightarrow \neg U)$

  A4: $(\forall x)U(x) \rightarrow U(t)$, unde t este termen arbitrar.

  A5: $(U \rightarrow V(y)) \rightarrow (U \rightarrow (\forall x)V(x))$, unde x nu este variabilă liberă nici în U, nici în V, iar y este o variabilă liberă în V şi nu apare în U.

- $R_{Pr} = \{mp, gen\}$ conţine ***regulile de inferenţă***:
  - ***modus ponens*** simbolizată astfel: $U, U \rightarrow V \vdash_{mp} V$
  - ***regula generalizării*** simbolizată astfel:

    $U(x) \vdash_{gen} (\forall x)U(x)$, unde x este o variabilă liberă în U.

Observaţii:

- Variabilele iau valori diferite într-un domeniu D şi sunt termeni generici, definiţii de tip cum ar fi: *carte, copil, eveniment.*
- Constantele iau valori fixe în domeniul D şi de obicei specifică nume de obiecte, persoane, evenimente (Ex: Paul_2, Carte_3).
- Cuantificatorii au prioritate mai mare decât conectivele.
- Au loc: $\Sigma_P \subset \Sigma_{Pr}$, $A_P \subset A_{Pr}$, $F_P \subset F_{Pr}$, $R_P \subset R_{Pr}$, deci $Teoreme_P \subset Teoreme_{Pr}$ (teoremele din calculul propoziţional sunt teoreme şi în calculul predicatelor).

Definiţia 2.2.

O ***formulă*** predicativă se numeşte ***închisă***, dacă toate variabilele sale sunt legate, iar în caz contrar se numeşte ***deschisă***.

Exemple de termeni:

- x, a, $f(x)$, $g(x,a)$, $g(f(x),y)$ unde:

  $x, y \in Var$, $a \in Const$, $f \in F_1$, $g \in F_2$.

Exemple de atomi:

- T, F, $p(x,y,a)$, $q(f(x),a)$, $r(a,g(f(x)),y)$ unde:

  $x, y \in Var$, $a \in Const$, $f, g \in F_1$, $p, r \in P_3$, $q \in P_2$.

Exemple de literali:

- $p(f(x), a, y, g(x,b))$, $\neg q(x, a, h(x))$ unde:

  $x, y \in Var$, $a, b \in Const$, $f, h \in F_1$, $g \in F_2$, $p \in P_4$, $q \in P_3$.

Exemple de formule predicative:

- Formula predicativă $(\forall x)(\exists z)(p(x,z,a) \vee (\exists y)q(x, f(y)))$ este închisă (toate variabilele sunt legate), unde $x, y, z \in Var$, $a \in Const$, $f \in F_1$, $p \in P_3$, $q \in P_2$.
- În formula predicativă deschisă $(\forall x)p(x,y) \wedge q(z,a)$ variabilele y şi z sunt libere, iar variabila x este legată (se află sub incidenţa cuantificatorului $\forall$), $a \in Const$, $x, y, z \in Var$, $p, q \in P_2$.

Transformarea afirmațiilor din limbaj natural în logica predicatelor

Exemplele următoare ilustrează transformarea unor afirmații din limbaj natural în formule predicative, punând în evidență utilizarea constantelor, simbolurilor de variabile, de funcții și de predicate, precum și a cuantificatorilor.

1. Obiectul ***a*** se află *deasupra* obiectului ***b*** dacă obiectul ***a*** se află *peste* obiectul ***b*** sau dacă obiectul ***a*** se află *peste* un alt obiect aflat *deasupra* obiectului ***b***.

   $(\forall x)(\forall y)\,(peste(x,y) \vee (\exists z)\,(peste(x,z) \wedge deasupra(z,y)) \rightarrow deasupra(x,y))$

   - *peste* și *deasupra* sunt simboluri predicative de aritate 2;
   - *peste*(a,b) = T dacă obiectul *a* se află peste obiectul *b*, în caz contrar predicatul este fals;
   - *deasupra*(a,b) = T dacă obiectul *a* se află deasupra obiectului *b*, în caz contrar predicatul este fals.

2. Dacă x și y sunt întregi nenegativi și x este mai mare decât y, atunci x^2 este mai mare decât y^2.

$(\forall x)(\forall y)(nneg(x) \wedge nneg(y) \wedge maimare(x, y) \rightarrow maimare(\,patrat(x), patrat(y))$

   - simbol de funcție: $patrat \in F_1$, $patrat(x) = x^2$
   - simboluri de predicate: *nneg* și *maimare*:
     $nneg \in P_1$, $nneg(x)$:"$x \geq 0$" și
     $maimare \in P_2$, $maimare(x,y)$:"$x > y$" .

3. **Axiomele care definesc numerele naturale:**

   a1. *Orice* număr natural are un unic successor imediat.

   existența: $(\forall x)(\exists y)egal(y, succesor(x))$

   unicitatea:

$(\forall x)(\forall y)(\forall z)(egal\,(y, succesor\,(x)) \wedge egal\,(z, successor\,(x)) \rightarrow egal\,(y,z))$

   a2. Numărul natural 0 *nu* este succesorul imediat al niciunui alt număr natural.

   $\neg\,(\exists x)egal(0, succesor(x))$

   a3. *Orice* număr natural cu excepția lui 0 are un unic predecesor imediat.

   existența: $(\forall x)(\exists y)(\neg\, egal(0,x) \wedge egal(y,\, predecesor(x)))$

unicitatea:

$(\forall x)(\forall y)(\forall z)(egal(y,\, predecesor(x)) \wedge egal(z,\, predecesor(x)) \rightarrow egal(y,z))$

Funcții unare: *succesor*, *predecesor* ; **predicat binar**: *egal*

Predicatul "*egal*" este definit de următoarele axiome:

$(\forall x)\, egal(x,x)$ --- reflexivitatea relației de egalitate

$(\forall x)(\forall y)(egal(x,y) \rightarrow egal(y,x))$ --- simetria relației de egalitate

$(\forall x)(\forall y)(\forall z)(egal(x,y) \wedge egal(y,z) \rightarrow egal(x,z))$ --- tranzitivitatea egalității

Următoarele formule predicative exprimă egalitatea succesorilor, respectiv predecesorilor, a două numere egale.

$(\forall x)(\forall y)(egal(x,y) \rightarrow egal(succesor(x), succesor(y)))$

$(\forall x)(\forall y)(egal(x,y) \rightarrow egal(predecesor(x), predecesor(y)))$

Definiția 2.3.

Fie formulele predicative $U_1, U_2, ..., U_n$ numite ipoteze (premise) și V o formulă predicativă. Spunem că V *este deductibilă (derivabilă) din* $U_1, U_2, ..., U_n$ (notație: $U_1, U_2, ..., U_n \vdash V$), dacă există o secvență de formule $(f_1, f_2, ..., f_m)$ astfel încât $f_m = V$ și $\forall i \in \{1,...,m\}$ avem a) sau b) sau c) sau d).

a) $f_i \in A_{\text{Pr}}$ (axiomă a calculului predicatelor);

b) $f_i \in \{U_1, U_2, ..., U_n\}$ (formulă ipoteză);

c) $f_{i_1}, f_{i_2} \vdash_{mp} f_i$, $i_1 < i$ și $i_2 < i$ (formula f_i s-a obținut prin inferență, folosind regula modus ponens, din două formule existente în secvență);

d) $f_j \vdash_{gen} f_i$, $j < i$ (formula f_i s-a obținut dintr-o formulă existentă în secvență, folosind regula generalizării).

Secvența $(f_1, f_2, ..., f_m)$ se numește ***deducția* lui V din** $U_1, U_2, ..., U_n$.

Definiția 2.4.

O formulă $U \in F_{\text{Pr}}$, astfel încât $\varnothing \vdash U$ (notație: $\vdash U$) se numește *teoremă*.

Observație:

Teoremele sunt formule deductibile doar din axiome, prin aplicarea regulilor de inferență: modus ponens și regula generalizării.

Sistemul axiomatic (deductiv) definit anterior permite extinderea unei mulțimi de formule predicative cu noi formule care sunt *consecințe sintactice* ale formulelor inițiale.

Notăm cu **Th**(X) mulțimea tuturor formulelor obținute prin aplicarea regulilor de inferență asupra formulelor din X și axiomelor logice, adică mulțimea tuturor formulelor derivabile (deduse, inferate) din X.

Th(X) se mai numește *închiderea deductivă* a mulțimii X

Th este *operatorul de derivabilitate (inferență) clasică*:

Th(X)={A | X $\vdash$ A}.

Operatorul **Th** și relația de derivabilitate $\vdash$ sunt extensii ale operatorului și relației corespunzătoare din logica propozițiilor, păstrând proprietățile prezentate în subcapitolul 1.6.

Exemplul 2.1.

Folosind definiția deducției să se demonstreze că are loc:

$(\forall y)(\forall z)(p(y) \vee q(z)) \vdash (\forall x)(p(x) \vee q(x))$

Construim secvența (f1, f2, f3, f4, f5, f6) de formule predicative astfel:

f1: $(\forall y)(\forall z)(p(y) \vee q(z))$ --- formulă ipoteză

f2: $(\forall y)(\forall z)(p(y) \vee q(z)) \rightarrow (\forall z)(p(x) \vee q(z))$ --- axioma A4, x-termen

f1, f2 $\vdash_{mp}$ f3 = $(\forall z)(p(x) \vee q(z))$

f4: $(\forall z)(p(x) \vee q(z)) \rightarrow p(x) \vee q(x)$ --- axioma A4, x-termen

f3, f4 $\vdash_{mp}$ f5 = $p(x) \vee q(x)$

f5 $\vdash_{gen}$ f6 = $(\forall x)(p(x) \vee q(x))$

(f1,f2,f3,f4,f5,f6) este deducția (demonstrația) formulei $(\forall x)(p(x) \vee q(x))$ din formula $(\forall y)(\forall z)(p(y) \vee q(z))$.

Exemplul 2.2.

Folosind definiția deducției să se demonstreze că are loc:

$\vdash (\forall x)(p(x) \wedge q(x)) \rightarrow (\forall x)p(x)$

Construim secvența (f1, f2, f3, f4, f5) de formule predicative astfel:

f1: $(\forall x)(p(x) \wedge q(x)) \rightarrow p(y) \wedge q(y)$ --- axioma A4, y-termen

f2: $p(y) \wedge q(y) \rightarrow p(y)$ --- teoremă în calculul propozițional

f1,f2 $\vdash_{reg.silogism}$ f3 = $(\forall x)(p(x) \wedge q(x)) \rightarrow p(y)$

S-a utilizat regula silogismului din calculul propozițional ca regulă de deducție (inferență): $U \rightarrow V, V \rightarrow Z \vdash U \rightarrow Z$.

f4: $((\forall x)(p(x) \wedge q(x)) \rightarrow p(y)) \rightarrow ((\forall x)(p(x) \wedge q(x)) \rightarrow (\forall x)p(x))$ --- --- axioma A5

f3, f4 $\vdash_{mp}$ f5 = $(\forall x)(p(x) \wedge q(x)) \rightarrow (\forall x)p(x)$

Secvența (f1, f2, f3, f4, f5) este deducția (demonstrația) teoremei din enunț.

2.2. Semantica logicii predicatelor de ordinul I

Semantica logicii predicatelor de ordinul I realizează legătura dintre constantele, simbolurile de funcții, simbolurile de predicate respectiv constantele, funcțiile și predicatele din conceptualizarea universului modelat. Totodată este furnizat un înțeles în termenii universului modelat pentru orice formulă din limbaj.

Definiția 2.5.

O ***interpretare*** pentru un limbaj L al calculului predicatelor este o pereche I =<D,m>, unde :

- D este o mulțime nevidă numită ***domeniul interpretării***;
- m este o funcție care asociază:
  - o valoare fixă $m(c)$ din domeniul D unei constante c;
  - o funcție $m(f)$: $D^n \rightarrow D$ fiecărui simbol de funcție f de aritate n;
  - un predicat $m(P)$: $D^n \rightarrow \{T,F\}$ fiecărui simbol de predicat P de aritate n.

Notații:

Fie I=<D,m> o interpretare, iar Var mulțimea variabilelor limbajului.

1. $|I| = D$ este domeniul interpretării I.
2. $I|X| = m(X)$ unde X este constantă, simbol de funcție sau simbol de predicat.
3. As(I) este mulțimea funcțiilor de asignare de variabile peste domeniul interpretării I.
   O funcție $a \in$ As(I) este definită astfel $a: Var \rightarrow |I|$.
4. $[a]_x = \{a' | a' \in \text{As}(I)$ și $a'(y) = a(y)$, pentru orice $y \neq x\}$.

Definiția 2.6.

Fie o interpretare I și $a \in$ As(I). Se definește inductiv ***funcția de evaluare*** v_a^I astfel:

- $v_a^I(x) = a(x),\ x \in Var$;
- $v_a^I(c) = I\,|c|,\ c \in Const$;
- $v_a^I(f(t_1,\dots,t_n)) = I\,|f|\,(v_a^I(t_1),\dots, v_a^I(t_n)),\ f \in F_n,\ n>0$;
- $v_a^I(P(t_1,\dots,t_n)) = I\,|P|\,(v_a^I(t_1),\dots, v_a^I(t_n)),\ P \in P_n,\ n>0$;
- $v_a^I(\neg A) = \neg\, v_a^I(A)$;
- $v_a^I(A \wedge B) = v_a^I(A) \wedge v_a^I(B)$;
- $v_a^I(A \vee B) = v_a^I(A) \vee v_a^I(B)$;
- $v_a^I(A \to B) = v_a^I(A) \to v_a^I(B)$;
- $v_a^I((\exists x)A(x))$=T dacă şi numai dacă $v_{a'}^I(A(x))$=T pentru o funcţie $a' \in [a]_x$;
- $v_a^I((\forall x)A(x))$=T dacă şi numai dacă $v_{a'}^I(A(x))$=T pentru orice funcţie $a' \in [a]_x$.

Definiţia 2.7. (concepte semantice)

- ***Formula*** predicativă A este ***realizabilă (consistentă)*** dacă şi numai dacă există o interpretare I şi o funcţie $a \in As(I)$ astfel încât $v_a^I(A)$=T. În caz contrar ***formula*** se numeşte ***nerealizabilă (inconsistentă)***.
- ***Formula*** predicativă A este ***adevărată în interpretarea I*** (notaţie $\models_I$ A), dacă şi numai dacă pentru orice funcţie $a \in As(I)$ de asignare de variabile are loc $v_a^I(A)$=T, iar I se numeşte ***model*** al lui A.
- Interpretarea I se numeşte ***anti-model*** al formulei predicative A dacă A este evaluată ca falsă în I, adică: $\forall\, a \in As(I)$ are loc $v_a^I(A)$=F.
- ***Formula*** predicativă A este ***validă (tautologie)***, notaţie: $\models$ A, dacă şi numai dacă A este adevărată în orice interpretare, adică: $\forall I$ -interpretare, $\models_I$ A.
- Două ***formule*** predicative, A şi B, sunt ***logic echivalente*** (notaţie: $A \equiv B$) dacă şi numai dacă $v_a^I(A) = v_a^I(B)$. $\forall I$ -interpretare şi $\forall\, a \in As(I)$.

- O mulţime S de formule predicative ***implică logic*** o formulă A dacă toate modelele mulţimii S (adică modelele conjuncţiei formulelor din S) sunt modele ale formulei A. Spunem că A este o ***consecinţă logică*** a mulţimii de formule S, notaţie: S $\models$ A.
- O ***mulţime de formule*** predicative este ***consistentă*** dacă formula obţinută prin conjuncţia elementelor sale este consistentă (are cel puţin un model când toate formulele iniţiale sunt închise).
- O ***mulţime de formule*** predicative este ***inconsistentă (nerealizabilă, contradictorie)*** dacă formula obţinută prin conjuncţia elementelor sale este inconsistentă (nu are niciun model când toate formulele iniţiale sunt închise).

Observaţii:

1. Evaluarea unei formule predicative închise **A** depinde doar de interpretarea, I, în care se evaluează formula, notându-se v^I (A).
2. La evaluarea unei formule predicative într-o interpretare cu domeniul finit se aplică următoarele:
   - o formulă cuantificată universal este înlocuită cu conjuncţia instanţelor acesteia folosind toate elementele domeniului de interpretare;
   - o formulă cuantificată existenţial este înlocuită cu disjuncţia instanţelor acesteia obţinute prin utilizarea tuturor elementelor domeniului de interpretare.

Datorită faptului că pentru o formulă predicativă există o infinitate de interpretări, este mai dificil de demonstrat validitatea sau inconsistenţa acesteia, spre deosebire de calculul propoziţional.

Exemplul 2.3.

Să se furnizeze un model şi un anti-model pentru formula predicativă:

$$U = (\forall x)(p(x) \vee q(x)) \to (\forall x)p(x) \vee (\forall x)q(x)$$

Formula U este închisă deci evaluarea sa nu depinde de asignările de variabile peste domeniul de interpretare, ci doar de interpretare.

1) Fie interpretarea: $I_1 = < D_1, m >$, unde:
   D_1 = **N** (mulţimea numerelor naturale);
   $m(p)$: **N** $\to$ {T,F}, $m(p)(x)$: "$x \vdots 2$";
   $m(q)$: **N** $\to$ {T,F}, $m(q)(x)$: "$x \vdots 3$".
   $v^{I_1}(U) = v^{I_1}((\forall x)(p(x) \vee q(x))) \to v^{I_1}((\forall x)p(x) \vee (\forall x)q(x)) =$
   $= v^{I_1}((\forall x)(p(x) \vee q(x))) \to v^{I_1}((\forall x)p(x)) \vee v^{I_1}((\forall x)p(x)) =$

$= (\forall x)_{x\in N}(x\vdots 2 \vee x\vdots 3) \to (\forall x)_{x\in N}(x\vdots 2) \vee (\forall x)_{x\in N}(x\vdots 3) =$
$= F \to F \vee F = F \to F = T.$

Are loc $v^{I_1}(U) = T$, deci formula U este evaluată ca adevărată în interpretarea I_1 care este astfel un *model* pentru U.

2) Fie interpretarea: $I_2 = < D_2, m >$, unde:

$D_2 = \{4,9\}$ – domeniul de interpretare;
$m(p) : \{4,9\} \to \{T,F\}$, $m(p)(x)$: "$x \vdots 2$";
$m(q) : \{4,9\} \to \{T,F\}$, $m(q)(x)$: "$x \vdots 3$".

Se observă că se utilizează aceeaşi funcţie de asignare m (restrânsă la domeniul D_2) ca şi la interpretarea I_1.

Pentru evaluarea formulei U în interpretarea I_2, cu domeniul finit $D_2 = \{4,9\}$, se va utiliza observaţia precedentă. Subformulele cuantificate universal se vor înlocui cu conjuncţia instanţelor acestora pentru $x=4$ şi $x=9$.

$$v^{I_2}(U) = v^{I_2}((\forall x)(p(x) \vee q(x))) \to v^{I_2}((\forall x)p(x) \vee (\forall x)q(x)) =$$
$$= v^{I_2}((\forall x)(p(x) \vee q(x))) \to v^{I_2}((\forall x)p(x)) \vee v^{I_2}((\forall x)q(x)) =$$
$$= (4\vdots 2 \vee 4\vdots 3) \wedge (9\vdots 2 \vee 9\vdots 3) \to (4\vdots 2 \wedge 9\vdots 2) \vee (4\vdots 3 \wedge 9\vdots 3) =$$
$$= (T \vee F) \wedge (F \vee T) \to (T \wedge F) \vee (F \wedge T) = T \wedge T \to F \vee F = T \to F = F$$

I_2 evaluează formula U ca falsă, deci este un *anti-model* pentru U şi astfel U nu este o formulă validă.

Exemplul 2.4.

Să se evalueze formula predicativă deschisă:

$U(z) = (\exists x)(\exists y)p(f(x,y),z)$ în interpretările I_1 şi I_2:

1) $I_1 = < D, m_1 >$, $D = \mathbf{Z}$ (mulţimea numerelor întregi),

$m_1(f)(x,y) = (x+y)^2$ şi $m_1(p)(x,y)$:"$x = y$".

Deoarece formula este deschisă, evaluarea sa depinde de asignarea de valori în domeniul interpretării pentru variabila liberă z, adică de funcţiile $a \in As(I_1)$, unde $a(z) \in \mathbf{Z}$.

$$v_a^{I_1}(U(z)) = (\exists x)_{x\in Z}(\exists y)_{y\in Z} \text{ "}(x+y)^2 = a(z)\text{"} =$$

$$= \begin{cases} T, \text{ daca } a(z) = \textit{patrat perfect} \\ F, \text{ in caz contrar} \end{cases}$$

Se observă că formula $U(z)$ este evaluată ca adevărată dacă variabilei libere z i se asignează în interpretarea I_1 un întreg care este pătrat perfect şi este evaluată ca falsă dacă lui z i se asignează un întreg care nu este pătrat perfect.

Concluzionăm că $U(z)$ este o formulă consistentă (realizabilă), dar nu este adevărată în interpretarea I_1, deci I_1 nu este model al formulei.

2) $I_2 = < D, m_2 >$, $D = \mathbf{Z}$ (mulţimea numerelor întregi),

$m_2(f)(x,y) = x + y$ şi $m_2(p)(x,y)$:"$x = y$".

Această interpretare diferă de interpretarea I_1 doar prin funcţia care se asignează simbolului de funcţie binară f.

$$v_a^{I_2}(U(z)) = (\exists x)_{x\in Z}(\exists y)_{y\in Z} \text{ "}x + y = a(z)\text{"} = T, \ \forall a(z) \in \mathbf{Z}.$$

Formula $U(z)$ este adevărată în interpretarea I_2, deci $\models_{I_2} U(z)$ şi astfel I_2 este model al formulei.

2.3. Echivalenţe logice în calculul predicatelor

Echivalenţele logice prezentate în acest subcapitol sunt perechi de echivalenţe duale ($\forall$ şi $\exists$ sunt cuantificatori duali) exprimând proprietăţi ale cuantificatorilor şi legătura acestora cu conectivele.

1. **Legile de expansiune**

   $(\forall x)A(x) \equiv (\forall x)A(x) \wedge A(t)$, unde t este un termen oarecare, $t \neq x$.
   semnificaţie: cuantificatorul universal este o conjuncţie infinită

   $(\exists x)A(x) \equiv (\exists x)A(x) \vee A(t)$, unde t este un termen oarecare, $t \neq x$.
   semnificaţie: cuantificatorul existenţial este o disjuncţie infinită

2. **Legile infinite ale lui DeMorgan**

   $\neg(\exists x)A(x) \equiv (\forall x)\neg A(x)$
   $\neg(\forall x)A(x) \equiv (\exists x)\neg A(x)$

3. **Legile de interschimbare a cuantificatorilor**

$(\exists x)(\exists y)A(x,y) \equiv (\exists y)(\exists x)A(x,y)$

$(\forall x)(\forall y)A(x,y) \equiv (\forall y)(\forall x)A(x,y)$

4. **Legile de extragere a cuantificatorilor în fața formulei**

$A \vee (\exists x)B(x) \equiv (\exists x)(A \vee B(x))$

$A \vee (\forall x)B(x) \equiv (\forall x)(A \vee B(x))$

$A \wedge (\exists x)B(x) \equiv (\exists x)(A \wedge B(x))$

$A \wedge (\forall x)B(x) \equiv (\forall x)(A \wedge B(x))$

- unde formula A nu conține pe x ca variabilă liberă sau legată.

$(\exists x)A(x) \vee B \equiv (\exists x)(A(x) \vee B)$

$(\forall x)A(x) \vee B \equiv (\forall x)(A(x) \vee B)$

$(\exists x)A(x) \wedge B \equiv (\exists x)(A(x) \wedge B)$

$(\forall x)A(x) \wedge B \equiv (\forall x)(A(x) \wedge B)$

- unde formula B nu conține pe x ca variabilă liberă sau legată.

5. **Legile distributivității**

5.1. ***distributivitate*** $\exists$ ***față de*** $\vee$

$(\exists x)(A(x) \vee B(x)) \equiv (\exists x)A(x) \vee (\exists x)B(x)$

5.2. ***distributivitate*** $\forall$ ***față de*** $\wedge$

$(\forall x)(A(x) \wedge B(x)) \equiv (\forall x)A(x) \wedge (\forall x)B(x)$

Observație:

Nu au loc proprietățile de distributivitate $\exists$ față de $\wedge$, $\exists$ față de $\to$, $\forall$ față de $\vee$ și $\forall$ față de $\to$, dar au loc proprietăți de semidistributivitate astfel:

5.3. ***semidistributivitate*** $\exists$ ***față de*** $\wedge$

Are loc: $\models (\exists x)(A(x) \wedge B(x)) \to (\exists x)A(x) \wedge (\exists x)B(x)$.

Formula $(\exists x)A(x) \wedge (\exists x)B(x) \to (\exists x)(A(x) \wedge B(x))$ *nu este validă.*

5.4. ***semidistributivitate*** $\forall$ ***față de*** $\vee$

Are loc: $\models (\forall x)A(x) \vee (\forall x)B(x) \to (\forall x)(A(x) \vee B(x))$.

Formula $(\forall x)(A(x) \vee B(x)) \to (\forall x)A(x) \vee (\forall x)B(x)$ *nu este validă.*

5.5. ***semidistributivitatea*** $\exists$ ***față de*** $\to$

Are loc: $\models ((\exists x)A(x) \to (\exists x)B(x)) \to (\exists x)(A(x) \to B(x))$.

Formula $(\exists x)(A(x) \to B(x)) \to ((\exists x)A(x) \to (\exists x)B(x))$ *nu este validă.*

5.6. ***semidistributivitatea*** $\forall$ ***față de*** $\to$

Are loc: $\models (\forall x)(A(x) \to B(x)) \to ((\forall x)A(x) \to (\forall x)B(x))$.

Formula $((\forall x)A(x) \to (\forall x)B(x)) \to (\forall x)(A(x) \to B(x))$ *nu este validă.*

Exemplul 2.5.

Să se arate că nu se poate interschimba cuantificatorul existențial cu cel universal, deci nu are loc echivalența logică:

$$(\exists x)(\forall y)B(x,y) \equiv (\forall y)(\exists x)B(x,y).$$

Interpretăm cele două formule considerând ca domeniu al interpretării, mulțimea tuturor persoanelor de pe glob, iar $B(x,y)$ reprezentând relația: "persoana x are încredere în persoana y".

În limbaj natural, conform interpretării definite. formula $U_1 = (\exists x)(\forall y)B(x,y)$ se traduce în afirmația:

a1: "Există o persoană care are încredere în toate persoanele de pe glob",

iar formula $U_2 = (\forall y)(\exists x)B(x,y)$ se traduce în afirmația:

a2: "Orice persoană de pe glob este de încredere pentru cel puțin o persoană".

Cele două afirmații nu au aceeași semnificație, deci în interpretarea furnizată, formulele U_1 și U_2 au semantică (înțeles) diferită. Concluzionăm astfel că nu are loc $U_1 \equiv U_2$. Se observă că **a1** implică **a2**, dar **a2** nu implică **a1**.

Demonstrațiile consecinței logice $U_1 \models U_2$ și a $U_2 \not\models U_1$ sunt propuse spre rezolvare cititorului.

Exemplul 2.6. (proprietatea 5.3.)

Să se arate că formula $U = (\exists x)A(x) \wedge (\exists x)B(x) \to (\exists x)(A(x) \wedge B(x))$ nu este validă.

Fie interpretarea I=<D,m>, unde:

D = mulțimea tuturor dreptelor unui plan P.

Fie $d \in P$, un obiect (dreaptă) constant din domeniul interpretării.

$m(A)$: $D \to \{T,F\}$, $m(A)(x)$: " $x \perp d$ ";

$m(B)$: $D \to \{T,F\}$, $m(B)(x)$: " $x \parallel d$ ";

$v^I(U) = v^I((\exists x)A(x) \wedge (\exists x)B(x)) \to v^I((\exists x)(A(x) \wedge B(x))) =$

$$= v^I((\exists x)A(x)) \wedge v^I((\exists x)B(x)) \rightarrow v^I((\exists x)(A(x) \wedge B(x))) =$$
$$= (\exists x)_{x\in D}(x \perp d) \wedge (\exists x)_{x\in D}(x \parallel d) \rightarrow (\exists x)_{x\in D}(x \perp d \ \wedge x \parallel d) =$$
$$= \mathrm{T} \wedge \mathrm{T} \rightarrow \mathrm{F} = \mathrm{T} \rightarrow \mathrm{F} = \mathrm{F}.$$

U este evaluată ca falsă în interpretarea I, deci I este un anti-model al formulei U. Concluzionăm că U nu este o formulă validă.

Exemplul 2.7. (proprietatea 5.4.)

Să se arate că formula $U = (\forall x)(A(x) \vee B(x)) \rightarrow (\forall x)A(x) \vee (\forall x)B(x))$ nu este validă.

Fie interpretarea: $I = <D,m>$, unde:

$D = \mathbf{R}\text{-}\{0\}$, mulţimea numerelor reale, exceptând 0.
$m(A): \mathrm{D} \rightarrow \{\mathrm{T},\mathrm{F}\}$, $m(A)(x)$: "$x>0$";
$m(B): \mathrm{D} \rightarrow \{\mathrm{T},\mathrm{F}\}$, $m(B)(x)$: "$x<0$";

$$v^I(U) = v^I((\forall x)(A(x) \vee B(x))) \rightarrow v^I((\forall x)A(x) \vee (\forall x)B(x)) =$$
$$= v^I((\forall x)(A(x) \vee B(x))) \rightarrow v^I((\forall x)A(x)) \vee v^I((\forall x)B(x)) =$$
$$= (\forall x)_{x\in R-\{0\}}(x>0 \ \vee \ x<0) \rightarrow (\forall x)_{x\in R-\{0\}}(x>0) \vee (\forall x)_{x\in R-\{0\}}(x<0)$$
$$= \mathrm{T} \rightarrow \mathrm{F} \vee \mathrm{F} = \mathrm{T} \rightarrow \mathrm{F} = \mathrm{F}.$$

Are loc $v^I(U) = \mathrm{F}$ deci I este un anti-model al formulei U.

În concluzie U nu este evaluată ca adevărată în toate interpretările posibile, deci nu este o formulă validă.

2.4. Forme normale ale formulelor predicative

Formele normale ale formulelor din calculului predicatelor sunt utilizate ca date de intrare în metode de demonstrare cum ar fi rezoluţia şi procedura bazată pe teorema lui Herbrand.

Definiţia 2.8.

- O formulă predicativă U este în ***formă normală prenexă*** dacă este de forma $(Q_1x_1)...(Q_nx_n)\ M$, unde Q_i, $i=1,...,n$ sunt cuantificatori logici, iar M nu conţine cuantificatori. Secvenţa $(Q_1x_1)...(Q_nx_n)$ se numeşte ***prefixul formulei*** U, iar M este ***matricea formulei*** U.
- O formulă predicativă U este în ***formă normală prenexă conjunctivă*** dacă U este în formă normală prenexă, iar matricea este în FNC.

Teorema 2.1.

Orice formulă din calculul predicatelor poate fi transformată într-o formă normală prenexă logic echivalentă cu aceasta.

Algoritmul de aducere la forma normală prenexă constă din următoarele transformări care păstrează echivalenţa logică:

Pas 1: Se înlocuiesc conectivele $\rightarrow$ şi $\leftrightarrow$ folosind $\neg, \wedge, \vee$.
Pas 2: Se redenumesc variabilele legate astfel încât acestea să fie distincte.
Pas 3: Se aplică legile finite şi infinite ale lui DeMorgan astfel încât cuantificatorii să nu fie precedaţi de negaţie.
Pas 4: Se utilizează echivalenţele logice care reprezintă legile de extragere a cuantificatorilor în faţa formulei.

!!! Ordinea de extragere a cuantificatorilor în faţa formulei este arbitrară.

Observaţii:

- După pasul 4 se obţine o formă normală prenexă care nu este unică.
- Formele prenexe obţinute în urma combinaţiilor privind ordinea de extragere a cuantificatorilor sunt logic echivalente conform legilor de interschimbare a cuantificatorilor.

Definiţia 2.9.

Fie U o formulă predicativă, iar $U^P = (Q_1x_1)...(Q_nx_n)\ M$ una dintre formele sale normale prenexe.

- Formulei U îi corespunde o formulă în ***formă normală Skolem*** notată U^S care se obţine astfel: pentru fiecare cuantificator existenţial Q_r din prefix se aplică următoarea transformare:
  (1) dacă înaintea simbolului Q_r nu apare niciun cuantificator universal, atunci se alege o constantă notată a, diferită de toate constantele care apar în M şi se înlocuiesc toate apariţiile variabilei x_r în M cu a. Se şterge (Q_rx_r) din prefixul formulei.
  (2) dacă înaintea simbolului Q_r apar cuantificatorii universali $Q_{s1}, ..., Q_{sm}$, unde $1 \le s1 < ... < sm < r$, atunci alegem un simbol f de funcţie de m variabile, diferit de celelalte simboluri de funcţii şi se înlocuieşte fiecare apariţie a variabilei x_r în M cu $f(x_{s1}, ..., x_{sm})$. Se şterge (Q_rx_r) din prefixul formulei.
- Constantele şi funcţiile utilizate pentru a înlocui variabilele cuantificate existenţial se numesc ***constante Skolem*** şi ***funcţii Skolem***.

- Formulei U îi corespunde o formulă în ***formă normală Skolem fără cuantificatori*** notată U^{Sq} care se obține prin eliminarea cuantificatorilor universali din U^S.
- Formulei U îi corespunde o formulă în ***formă normală clauzală*** notată U^c care se obține din U^{Sq} prin aducerea la forma normală conjunctivă.

Observații:

- Prefixul formei U^S este format doar din cuantificatori universali.
- O formulă predicativă are în general mai multe forme Skolem corespunzătoare tuturor formelor normale prenexe.
- Transformările utilizate în procesul de Skolemizare nu păstrează echivalența logică, dar păstrează inconsistența conform teoremei următoare.

Teorema 2.2.

Fie $U_1,U_2,...,U_n,V$ formule predicative.

1. *V este inconsistentă **dacă și numai dacă** V^p este inconsistentă **dacă și numai dacă** V^S este inconsistentă **dacă și numai dacă** V^{Sq} este inconsistentă **dacă și numai dacă** V^c este inconsistentă.*
2. *Mulțimea $\{U_1,U_2,...,U_n\}$ este inconsistentă **dacă și numai dacă** $\{U_1^c,U_2^c,...,U_n^c\}$ este o mulțime inconsistentă.*

Exemplul 2.8.

Aduceți la formele normale prenexă, Skolem și clauzală formula:

$$A = \neg((\forall x)(p(x)\to q(x)) \to ((\forall x)p(x) \to (\forall x)q(x))).$$

Se aplică algoritmul de aducere la forma prenexă, apoi skolemizarea.

Pas 1: înlocuirea conectivelor $\to$ interioare

$$A \equiv \neg((\forall x)(\neg p(x)\vee q(x)) \to (\neg(\forall x)p(x) \vee (\forall x)q(x)))$$

Pas 1: înlocuirea conectivei $\to$

$$A \equiv \neg(\neg((\forall x)(\neg p(x)\vee q(x))) \vee (\neg(\forall x)p(x) \vee (\forall x)q(x)))$$

Pas 2: redenumirea variabilelor legate astfel încât să fie distincte

$$A \equiv \neg(\neg((\forall x)(\neg p(x)\vee q(x))) \vee (\neg(\forall y)p(y) \vee (\forall z)q(z)))$$

Pas 3: aplicarea legilor finite și infinite ale lui DeMorgan

$$A \equiv (\forall x)(\neg p(x)\vee q(x)) \wedge \neg(\neg(\forall y)p(y) \vee (\forall z)q(z))$$

Pas 3: aplicarea legilor finite și infinite ale lui DeMorgan

$$A \equiv (\forall x)(\neg p(x)\vee q(x)) \wedge (\forall y)p(y) \wedge (\exists z)\neg q(z)$$

Pas 4: extragerea cuantificatorilor în fața formulei

$$A \equiv (\exists z)(\forall x)(\forall y)((\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(z)) = A_1 = A_1^p$$

$$A \equiv (\forall x)(\exists z)(\forall y)((\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(z)) = A_2 = A_2^p$$

$$A \equiv (\forall x)(\forall y)(\exists z)((\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(z)) = A_3 = A_3^p$$

A_1^p, A_2^p, A_3^p sunt trei forme normale prenexe asociate formulei A. Aceste forme prenexe diferă prin ordinea cuantificatorilor în prefixul formei, matricea fiind aceeași. Se observă că matricea este în FNC și deci aceste forme sunt totodată forme normale prenexe conjunctive.

Deoarece după redenumirea variabilelor legate din A, sunt 3 variabile legate distincte, independente, există 3!=6 posibile forme normale prenexe logic echivalente cu A, dintre care s-au ales doar cele trei forme de mai sus.

Se aplică skolemizarea formelor prenexe: A_1^p, A_2^p, A_3^p.

Din A_1^p cu substituția $[z \leftarrow a]$, a-constantă Skolem, se obține:

$$A_1^S = (\forall x)(\forall y)((\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(a))$$

Din A_2^p cu substituția $[z \leftarrow f(x)]$, f-funcție Skolem unară, se obține:

$$A_2^S = (\forall x)(\forall y)((\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(f(x)))$$

Din A_3^p cu substituția $[z \leftarrow g(x,y)]$, g-funcție Skolem binară, se obține:

$$A_3^S = (\forall x)(\forall y)((\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(g(x,y)))$$

Prin eliminarea cuantificatorilor universali se obțin formele clauzale:

$$A_1^{Sq} = (\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(a) = A_1^c$$

$$A_2^{Sq} = (\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(f(x)) = A_2^c$$

$$A_3^{Sq} = (\neg p(x)\vee q(x)) \wedge p(y) \wedge \neg q(g(x,y)) = A_3^c$$

Exemplul 2.9.

Construiți o formă normală prenexă și o formă normală Skolem pentru formula $U = (\exists x)(\forall y)p(x,y) \vee (\exists z)(\neg q(z) \vee (\forall u)(\exists t)r(z,u,t))$.

- formă normală prenexă:

$$U^p = (\exists x)(\forall y)(\exists z)(\forall u)(\exists t)(p(x,y) \vee \neg q(z) \vee r(z,u,t))$$

- formă normală Skolem:

  $U^S = (\forall y)(\forall u)(p(a,y) \vee \neg q(f(y)) \vee r(f(y),u,g(y,u)))$

  S-au utilizat substituțiile $[x \leftarrow a], [z \leftarrow f(y)], [t \leftarrow g(y,u)]$, unde: a-constantă Skolem, f și g - funcții Skolem.
- forma Skolem fără cuantificatori este și formă normală clauzală:

  $U^c = U^{Sq} = p(a,y) \vee \neg q(f(y)) \vee r(f(y),u,g(y,u))$

2.5. Proprietățile logicii predicatelor de ordinul I

Teorema 2.3. (teorema de completitudine și corectitudine)

Fie S o mulțime de formule predicative, iar A o formulă predicativă.

- ***completitudinea:*** *dacă $S \models A$ atunci $S \vdash A$.*
- ***corectitudinea:*** *dacă $S \vdash A$ atunci $S \models A$.*

O particularizare a acestei teoreme conduce la următorul rezultat: *„O formulă predicativă este tautologie dacă și numai dacă este teoremă în calculul predicatelor."*

Teorema 2.4. (teorema respingerii)

Fie S o mulțime de formule predicative, iar A o formulă predicativă. Dacă $S \cup \{\neg A\}$ este inconsistentă atunci $S \vdash A$.

Această teoremă formalizează raționamentul reducerii la absurd și stă la baza metodelor de demonstrare prin respingere cum ar fi: rezoluția, metoda tabelelor semantice, metoda bazată pe teorema lui Herbrand.

Teorema 2.5. (teorema de deducție)

Fie S o mulțime de formule predicative, iar A o formulă predicativă. Dacă $S \cup \{A\} \vdash B$ atunci $S \vdash A \rightarrow B$.

Exemplul 2.10.

Folosind teorema de deducție să se demonstreze că formula $(\forall x)(A(x) \rightarrow B(x)) \rightarrow ((\forall x)A(x) \rightarrow (\forall x)B(x))$ este o teoremă.

Pentru început demonstrăm că are loc: $(\forall x)(A(x) \rightarrow B(x)), (\forall x)A(x) \vdash (\forall x)B(x)$ folosind definiția deducției.

Se construiește secvența de formule predicative (f1,f2,..., f8) astfel:

f1: $(\forall x)(A(x) \rightarrow B(x))$ --- formulă ipoteză

f2: $(\forall x)(A(x) \rightarrow B(x)) \rightarrow (A(y) \rightarrow B(y))$ --- axioma A4, y-termen

f1, f2 $\vdash_{mp}$ f3 $= A(y) \rightarrow B(y)$

f4: $(\forall x)A(x)$ --- formulă ipoteză

f5: $(\forall x)A(x) \rightarrow A(y)$ --- axioma A4, y-termen

f4, f5 $\vdash_{mp}$ f6 $= A(y)$

f3, f6 $\vdash_{mp}$ f7 $= B(y)$

f7 $\vdash_{gen}$ f8 $= (\forall x)\, B(x)$

(f1,...,f8) este deducția formulei $(\forall x)\, B(x)$ din ipotezele $(\forall x)(A(x) \rightarrow B(x))$ și $(\forall x)A(x)$.

Pornind de la această deducție și aplicând de două ori teorema de deducție obținem:

$(\forall x)(A(x) \rightarrow B(x)), (\forall x)A(x) \vdash (\forall x)B(x)$ dacă și numai dacă

$(\forall x)(A(x) \rightarrow B(x)) \vdash (\forall x)A(x) \rightarrow (\forall x)B(x)$ dacă și numai dacă

$\vdash (\forall x)(A(x) \rightarrow B(x)) \rightarrow ((\forall x)A(x) \rightarrow (\forall x)B(x))$

Astfel s-a demonstrat teorema din enunț.

Teorema 2.6. (Church 1936)

Problema validității unei formule în calculul predicatelor de ordinul I este nedecidabilă. Mulțimea formulelor valide din acest sistem logic este recursiv numărabilă, adică există o procedură P care, având ca intrare o formulă U din limbaj, are următorul comportament:

- *dacă formula U este validă, P se termină și furnizează răspunsul corespunzător;*
- *dacă formula U nu este validă, P se termină cu răspunsul corespunzător sau execuția procedurii nu se încheie niciodată.*

Datorită acestor considerente ***calculul predicatelor de ordinul I nu este decidabil, este doar semi-decidabil.***

Rezultatul de nedecidabilitate stabilit de teorema precedentă poate fi îmbunătățit pentru fragmente ale logicii predicatelor de ordinul I.

Teorema 2.7. [11]

1. Există o procedură decizională pentru validitatea formulelor aparținând claselor de formule cu proprietatea că prefixul formelor prenexe este:

- $\forall^*\exists^*$, adică $(\forall x_1)...(\forall x_n)(\exists y_1)...(\exists y_m)$, $m,n \geq 0$;
- $\forall^*\exists\forall^*$, adică $(\forall x_1)...(\forall x_n)(\exists y)(\forall z_1)...(\forall z_m)$, $m,n \geq 0$;
- $\forall^*\exists\exists\forall^*$, adică $(\forall x_1)...(\forall x_n)(\exists y_1)(\exists y_2)(\forall z_1)...(\forall z_m)$, $m,n \geq 0$;

2. *Există o procedură decizională pentru consistența (realizabilitatea) formulelor aparținând claselor de formule cu proprietatea că prefixul formelor prenexe este:*

- $\exists^*\forall^*$, adică $(\exists x_1)...(\exists x_n)(\forall y_1)...(\forall y_m)$, $m,n \geq 0$;
- $\exists^*\forall\exists^*$, adică $(\exists x_1)...(\exists x_n)(\forall y)(\exists z_1)...(\exists z_m)$, $m,n \geq 0$;
- $\exists^*\forall\forall\exists^*$, adică $(\exists x_1)...(\exists x_n)(\forall y_1)(\forall y_2)(\exists z_1)...(\exists z_m)$, $m,n \geq 0$;

Cele două enunțuri ale teoremei sunt duale și impun restricții asupra sintaxei formulelor pentru decidabilitatea validității/consistenței formulelor predicative.

Se observă că tabelele semantice (vezi capitolul 3) asociate formulelor (negațiilor formulelor) din clasele de formule respective sunt finite, deci se poate decide consistența (validitatea) acestor formule.

2.6. Substituții și unificatori

Definiția 2.10.

O ***substituție*** este o funcție definită pe mulțimea variabilelor, *Var*, cu valori în mulțimea termenilor *TERM*.

Se notează cu $\theta = [x_1 \leftarrow t_1,...,x_k \leftarrow t_k]$ și reprezintă o mulțime finită de înlocuiri de variabile cu termeni. $x_1,...,x_k$ sunt variabile distincte, iar $t_1,...,t_k$ sunt termeni, astfel încât: $\forall i = 1,...,k$, $t_i \neq x_i$ și x_i nu este subtermen al lui t_i.

$dom(\theta)=\{x_1,...,x_k\}$ se numește ***domeniul substituției*** θ.

Se utilizează literele grecești: $\varphi, \delta, \phi, \eta, \theta, \lambda$ pentru a nota o substituție, iar ε simbolizează substituția vidă.

Definiția 2.11.

Aplicarea substituției $\theta = [x_1 \leftarrow t_1,...,x_k \leftarrow t_k]$ asupra formulei predicative deschise *U* este definită recursiv astfel:

- $\theta(x_i) = t_i, x_i \in dom(\theta)$; $\theta(x) = x, x \notin dom(\theta)$;
- $\theta(c) = c$, c – constantă;
- $\theta(f(t_1,...,t_n)) = f(\theta(t_1),...,\theta(t_n)), f \in F_n$;
- $\theta(p(t_1,...,t_n)) = p(\theta(t_1),...,\theta(t_n)), p \in P_n$;
- $\theta(\neg U) = \neg\theta(U)$;
- $\theta(U \circ V) = \theta(U) \circ \theta(V)$, *unde* $\circ \in \{\vee,\wedge,\rightarrow,\leftrightarrow\}$.

Observație:

Instanța $\theta(U)$ a formulei U se obține prin înlocuirea simultană a fiecărei apariții a variabilei libere x_i din U cu t_i, pentru $i=1,...,k$.

Exemplul 2.11.

Fie substituția $\theta = [x \leftarrow a, y \leftarrow f(x), z \leftarrow g(a,t)]$ și formula predicativă deschisă $U(x,y,z,t) = p(x,b,t) \wedge q(y,z,x)$. Să se calculeze $\theta(U)$.

$$\theta(U) = \theta(p(x,b,t) \wedge q(y,z,x)) = \theta(p(x,b,t)) \wedge \theta(q(y,z,x)) =$$
$$= p(a,b,t) \wedge q(f(x),g(a,t),a)$$

Definiția 2.12.

Compunerea a două substituții $\theta_1 = [x_1 \leftarrow t_1,...,x_k \leftarrow t_k]$ și $\theta_2 = [y_1 \leftarrow s_1,...,y_n \leftarrow s_n]$ este definită astfel:

$$\theta = \theta_1\theta_2 = [x_i \leftarrow \theta_2(t_i) \mid x_i \in dom(\theta_1), x_i \neq \theta_2(t_i)] \cup$$
$$\cup [y_j \leftarrow s_j \mid y_j \in dom(\theta_2) \setminus dom(\theta_1)]$$

Observație:

Compunerea a două substituții poate sa aibă ca rezultat o mulțime de înlocuiri de variabile care nu sunt toate valide, deci rezultatul compunerii nu este totdeauna o substituție.

Exemplul 2.12.

Fie substituțiile: $\theta_1 = [x \leftarrow f(y,z)]$ și $\theta_2 = [y \leftarrow x, z \leftarrow a]$.

$$\theta_1\theta_2 = [x \leftarrow f(y,z)][y \leftarrow x, z \leftarrow a] = [x \leftarrow f(x,a), y \leftarrow x, z \leftarrow a]$$

În urma compunerii celor două substituții s-au obținut trei înlocuiri de variabile. Prima înlocuire: $x \leftarrow f(x,a)$ nu este validă deoarece x este subtermen al lui $f(x,a)$ și astfel conform definiției nu s-a obținut în urma compunerii o substituție.

Proprietățile operației de compunere a substituțiilor:

1. Elementul neutru: ε-substituția vidă, $\varepsilon\theta = \theta\varepsilon = \theta$, $\forall\theta$-substituție.
2. Asociativitatea: $\theta_1(\theta_2\theta_3) = (\theta_1\theta_2)\theta_3$, $\forall\theta_1,\theta_2,\theta_3$-substituții.
3. Compunerea a două substituții nu este în general comutativă, deci nu are loc $\theta_1\theta_2 = \theta_2\theta_1$, $\forall\theta_1,\theta_2$-substituții.

Exemplul 2.13.

Verificarea proprietății de asociativitate: $\theta_1(\theta_2\theta_3) = (\theta_1\theta_2)\theta_3$.

Considerăm substituțiile:

$\theta_1 = [x \leftarrow h(a,y), z \leftarrow f(b)]$

$\theta_2 = [y \leftarrow g(a), u \leftarrow x]$

$\theta_3 = [x \leftarrow f(a), y \leftarrow b, z \leftarrow b]$

Calculăm compunerea $\theta_1(\theta_2\theta_3)$.

$$\theta_2\theta_3 = [y \leftarrow g(a), u \leftarrow x]\ [x \leftarrow f(a), y \leftarrow b, z \leftarrow b] =$$
$$= [y \leftarrow g(a), u \leftarrow f(a), x \leftarrow f(a), z \leftarrow b] =$$
$$\theta_1(\theta_2\theta_3) = [x \leftarrow h(a,y), z \leftarrow f(b)]$$
$$[y \leftarrow g(a), u \leftarrow f(a), x \leftarrow f(a), z \leftarrow b] =$$
$$= [x \leftarrow h(a, g(a)), z \leftarrow f(b), y \leftarrow g(a), u \leftarrow f(a)]$$

Calculăm compunerea $(\theta_1\theta_2)\theta_3$.

$$\theta_1\theta_2 = [x \leftarrow h(a,y), z \leftarrow f(b)]\ [y \leftarrow g(a), u \leftarrow x] =$$
$$= [x \leftarrow h(a, g(a)), z \leftarrow f(b), y \leftarrow g(a), u \leftarrow x]$$
$$(\theta_1\theta_2)\theta_3 = [x \leftarrow h(a, g(a)), z \leftarrow f(b), y \leftarrow g(a), u \leftarrow x]$$
$$[x \leftarrow f(a), y \leftarrow b, z \leftarrow b] =$$
$$= [x \leftarrow h(a, g(a)), z \leftarrow f(b), y \leftarrow g(a), u \leftarrow f(a)]$$

Are loc $\theta_1(\theta_2\theta_3) = (\theta_1\theta_2)\theta_3$.

Astfel, la compunerea mai multor substituții putem asocia câte două substituții succesive în ordinea dorită fără a schimba rezultatul.

Exemplul 2.14.

Verificarea necomutativității. Se dau substituțiile:

$\theta_1 = [x \leftarrow f(y), y \leftarrow f(a), z \leftarrow u]$ și

$\theta_2 = [y \leftarrow g(a), u \leftarrow z, v \leftarrow f(f(a))]$

$$\lambda_1 = \theta_1\theta_2 =$$
$$= [x \leftarrow \theta_2(f(y)), y \leftarrow \theta_2(f(a)), z \leftarrow \theta_2(u)] \cup [u \leftarrow z, v \leftarrow f(f(a))] =$$
$$= [x \leftarrow f(g(a)), y \leftarrow f(a), z \leftarrow z, u \leftarrow z, v \leftarrow f(f(a))] =$$
$$= [x \leftarrow f(g(a)), y \leftarrow f(a), u \leftarrow z, v \leftarrow f(f(a))]$$
$$\lambda_2 = \theta_2\theta_1 = [y \leftarrow g(a), v \leftarrow f(f(a)), x \leftarrow f(y), z \leftarrow u]$$

Se observă că $\lambda_1 \neq \lambda_2$, deci compunerea a două substituții nu este în general comutativă.

Calculăm pentru formula $U = p(u,v,x,y,z)$ instanțele $\lambda_1(U)$ și $\lambda_2(U)$.

$\lambda_1(U) = p(z, f(f(a)), f(g(a)), f(a), z)$

$\lambda_2(U) = p(u, f(f(a)), f(y), g(a), u)$

Definiția 2.13.

- O substituție θ se numește ***unificator*** al termenilor t_1 și t_2 dacă $\theta(t_1) = \theta(t_2)$. Termenul $\theta(t_1)$ se numește ***instanța comună*** a termenilor unificați.
- Un ***unificator al mulțimii*** de formule predicative $\{U_1,...,U_n\}$ este o substituție θ cu proprietatea: $\theta(U_1) = ... = \theta(U_n)$.

Exemplul 2.15.

- Termenii *f(x,x)* și *f(a,b)* nu pot fi unificați, deoarece nu au o instanță comună dacă *a* și *b* sunt constante distincte.
- Atomii *p(x)* și *p(f(x))* nu sunt unificabili deoarece *x* este un subtermen al lui *f(x)*.
- Atomii *p(x)* și *q(a)* nu sunt unificabili deoarece nu au același simbol de predicat.
- Atomii *p(x,y,z)* și *p(a,b)* nu sunt unificabili deoarece cele două simboluri predicative cu același nume au aritate diferită.
- Termenii *t1=f(x,b)* și *t2=f(a,y)* au instanța comună *f(a,b)*= θ *(t1)*= θ *(t2)*, unde $\theta = [x \leftarrow a, y \leftarrow b]$ este unificatorul comun.
- Termenii *t1=g(g(x))* și *t2=g(y)* au mai mulți unificatori, dintre care amintim:

  $\mu 1 = [y \leftarrow g(x)]$, instanța comună este $\mu 1$ *(t1)*= $\mu 1$ *(t2)* = *g(g(x))*;

  $\mu 2 = [x \leftarrow 5, y \leftarrow g(5)]$, instanța comună este *g(g(5))*.

Se observă că unificatorul $\mu 1$ este mai general decât unificatorul $\mu 2$.

Definiția 2.14.

Cel mai general unificator (mgu) este un unificator μ cu proprietatea că orice alt unificator θ se obține din compunerea lui μ cu o altă substituție λ: $\theta = \mu\lambda$.

Algoritmul pentru determinarea celui mai general unificator a doi literali:

Date de intrare: $l1 = p_1(t_{11}, t_{12}, ..., t_{1n})$ și $l2 = p_2(t_{21}, t_{22}, ..., t_{2k})$ doi literali

Date de ieșire: *mgu(l1,l2)* sau mesaj: " *l1, l2 nu sunt unificabili*"

dacă ($p_1 \neq p_2$) // simbolurile predicative sunt diferite

  atunci scrie *"l1, l2 nu sunt unificabili"*; STOP;

sf_dacă

dacă($n \neq k$) // aritate diferită pentru același simbol predicativ

  atunci scrie *"l1, l2 nu sunt unificabili"*; STOP;

sf_dacă

$\theta = \varepsilon$; // inițializare cu substituția vidă

cât_timp ($\theta(l1) \neq \theta(l2)$) **execută**

Din $\theta(l1), \theta(l2)$ se determină cele mai din stânga simboluri de funcții, constante sau variabile diferite și notăm cu $t1$ și $t2$ termenii corespunzători.

dacă (*niciunul dintre* $t1$ *și* $t2$ *nu este variabilă* sau *unul este subtermenul celuilalt*)

atunci scrie *"l1, l2 nu sunt unificabili"*; STOP;

sf_dacă

dacă (*t1 este variabilă*)

atunci $\lambda = [t1 \leftarrow t2]$;

altfel $\lambda = [t2 \leftarrow t1]$;

sf_dacă

$\theta = \theta\lambda$;

dacă (θ *nu este o substituție*)

atunci scrie *"l1, l2 nu sunt unificabili"*; STOP;

sf_dacă

sf_cât_timp

scrie *"l1 și l2 sunt unificabili, mgu(l1,l2)="* θ;

Sf_algoritm

Observație: Cel mai general unificator a doi literali nu este unic.

Exemplul 2.16.

Să se determine cel mai general unificator al literalilor:

$l1=p(a,x,f(g(y)))$ și $l2=p(y,f(z),f(z))$, unde:

a-constantă, x,y,z-variabile, f,g-simboluri de funcții unare, iar p-simbol de predicat cu aritatea trei.

Deoarece simbolurile de predicate ale celor doi literali sunt identice si au aceeași aritate se continuă cu unificarea termenilor care reprezintă argumentele de la stânga spre dreapta. Sunt subliniați termenii care se unifică la fiecare pas.

$l1=p(\underline{a},x,f(g(y)))$ *și* $l2=p(\underline{y},f(z),f(z))$

$\theta = [y \leftarrow a]$, unificator al termenilor y (variabilă) și a(constantă)

$\theta(l1)=p(a,\underline{x},f(g(a)))$, $\theta(l2)=p(a,\underline{f(z)},f(z))$;

$[x \leftarrow f(z)]$, unificator al termenilor x și $f(z)$

$\theta = [y \leftarrow a][x \leftarrow f(z)] = [y \leftarrow a, x \leftarrow f(z)]$,

$\theta(l1)=p(a,f(z),f(\underline{g(a)}))$, $\theta(l2)=p(a,f(z),f(\underline{z}))$;

$[z \leftarrow g(a)]$, unificator al termenilor z și $g(a)$

$\theta = [y \leftarrow a, x \leftarrow f(z)][z \leftarrow g(a)] = [y \leftarrow a, x \leftarrow f(g(a)), z \leftarrow g(a)] =$
$= mgu(l1,l2)$

Instanța comună a celor doi literali: $\theta(l1)=\theta(l2)=p(a,f(g(a)),f(g(a)))$.

Exemplul 2.17.

Să se verifice dacă sunt unificabili literalii $l1$ și $l2$, unde:

$l1=q(x,a,f(x,y))$ și $l2=q(b,y,f(z,c))$

Cei doi literali au același simbol de predicat și aceeași aritate, deci se va încerca unificarea termenilor corespunzători argumentelor.

$l1= q(\underline{x},a,f(x,y))$ și $l2=q(\underline{b},y,f(z,c))$

$\theta = [x \leftarrow b]$, unificator al termenilor x (variabilă) și b (constantă)

$\theta(l1)=q(b,\underline{a},f(b,y))$, $\theta(l2)=q(b,\underline{y},f(z,c))$;

$[y \leftarrow a]$, unificator al termenilor y (variabilă) și a (constantă)

$\theta = [x \leftarrow b][y \leftarrow a] = [x \leftarrow b, y \leftarrow a]$,

$\theta(l1)= q(b,a,f(\underline{b},a))$, $\theta(l2)=q(b,a,f(\underline{z},c))$;

$[z \leftarrow b]$, unificator al termenilor z și b

$\theta = [x \leftarrow b, y \leftarrow a][z \leftarrow b] = [x \leftarrow b, y \leftarrow a, z \leftarrow b]$

$\theta(l1)= q(b,a,f(b,\underline{a}))$, $\theta(l2)=q(b,a,f(b,\underline{c}))$;

Termenii a și c, fiind constante distincte, nu sunt unificabili, deci termenii $f(b,a)$ *și* $f(z,c)$ corespunzători celui de-al treilea argument al literalilor nu sunt unificabili.

Concluzionăm că literalii din enunț nu sunt unificabili.

2.7. Procedura de demonstrare bazată pe teorema lui Herbrand

Nedecidabilitatea calculului predicatelor se datorează unui număr infinit de posibile interpretări pentru o formulă predicativă. Este oare necesară evaluarea unei formule în toate interpretările sale posibile pentru a decide consistența/inconsistența acesteia?

Prin aducerea la o formă normală clauzală, problema verificării consistenței/inconsistenței unei formule predicative a fost semnificativ simplificată. Herbrand a simplificat în continuare această problemă prin găsirea unui domeniu de interpretare special, numit ***domeniul (universul) Herbrand*** și arătând că o *formă normală clauzală este inconsistentă dacă și numai dacă este falsă în toate interpretările definite pe domeniul Herbrand.*

Definiția 2.15.

1. Un ***termen*** sau un ***atom*** este ***de bază*** dacă nu conține variabile.
2. O ***formulă*** este ***de bază*** dacă nu conține variabile și cuantificatori.

3. O formulă A^b este o ***instanță de bază*** a unei formule A fără cuantificatori dacă, A^b se obține prin substituirea variabilelor libere din A cu termeni de bază.

Exemplul 2.18.

- a, $f(a)$, $g(f(a),b)$ – termeni de bază, unde a,b-constante, f,g-simboluri de funcții;
- $p(a)$, $q(f(a),b)$ - atomi de bază, unde a,b - constante, f - simbol de funcție, p,q-simboluri de predicate;
- Din formula predicativă deschisă $A(x,y,z) = p(x) \wedge \neg q(y, f(z))$ se obține instanța de bază $A^b = p(a) \wedge \neg q(b, f(a))$ folosind substituția: $[x \leftarrow a, y \leftarrow b, z \leftarrow a]$.

Definiția 2.16.

Fie $S = \{U_1, U_2, ..., U_n\}$ o mulțime de formule predicative deschise.

1. ***Universul Herbrand*** asociat mulțimii S, notat H_S, este mulțimea tuturor termenilor de bază construiți folosind constantele și simbolurile de funcții din formulele mulțimii S astfel:

$H_S = \cup_{i \geq 0} H_i$, unde:

$H_0 = C$ = mulțimea constantelor din S

Dacă $C = \varnothing$ atunci $H_0 = \{c\}$;

$H_{i+1} = H_i \cup$

$\cup \{f(t_1,...,t_k) \mid t_1,...,t_k \in H_i, f \in F_k, f$ – simbol de functie din $S\}$

2. ***Baza Herbrand*** asociată mulțimii S, notată BH_S, este mulțimea tuturor atomilor de bază construiți folosind simbolurile predicative din S și termenii de bază din H_S.

$BH_S = \{p(t_1,...,t_k) \mid t_1,...,t_k \in H_S, p \in P_k, p$ – simbol de predicat din $S\}$

3. ***Sistemul Herbrand*** asociat mulțimii S, notat SH_S, este mulțimea tuturor instanțelor de bază ale formulelor din S obținute prin înlocuirea variabilelor libere cu termeni de bază din H_S.

$SH_S = \{U(t_1,...,t_k) \mid t_1,...,t_k \in H_S, U$ - formula din $S\}$.

4. O ***interpretare Herbrand*** a mulțimii S este o interpretare $I=<D,m>$, având ca și domeniu universul Herbrand, $D = H_S$ și funcția m definită astfel:

- $m(a) = a, \forall a \in H_0$;
- $m(f): H_S^k \to H_S, f \in F_k$, f - simbol de funcție din S, unde: $m(f)(t_1,...,t_k) = f(m(t_1),...,m(t_k)), t_1,...,t_k \in H_S$;
- nu există restricții privitoare la asignarea de predicate reale simbolurilor de predicate din S.

5. Un ***model Herbrand*** al mulțimii S, este o interpretare Herbrand care evaluează toate formulele din S ca adevărate, adică satisface S. Acesta se poate identifica cu o submulțime a bazei Herbrand ce conține atomii de bază care se consideră evaluați ca adevărați.

Exemplul 2.19.

1) Să se construiască un model Herbrand al mulțimii de formule predicative: $S_1 = \{ p(a) \vee \neg q(x), r(y) \}$.

$H_{S_1} = \{a\}$ - univers Herbrand finit

$BH_{S_1} = \{ p(a), q(a), r(a) \}$ - bază Herbrand finită

$SH_{S_1} = \{ p(a) \vee \neg q(a), r(a) \}$ - sistem Herbrand finit

Un model Herbrand al mulțimii S_1 este $I_1 =< D_1, m_1 >$ unde:

$D_1 = H_{S_1} = \{a\}$ și m_1 este:

- $m_1(a) = a$
- $m_1(p)(a)$ =T, $m_1(q)(a)$ = F, $m_1(r)(a)$ =T,

identificat simplu prin submulțimea $\{p(a), r(a)\}$ a bazei Herbrand.

Are loc: $v^{I_1}(S_1)$ =T, deci S_1 este o mulțime consistentă.

2) Să se construiască un model Herbrand al mulțimii de formule: $S_2 = \{ p(f(x)), \neg q(z,y) \}$.

$H_{S_2} = \{c, f(c), f(f(c)), ...\}$; c - constantă, universul Herbrand este infinit

$BH_{S_2} = \{p(f(c)), q(c,c), p(f(f(c))), q(c,f(c)), q(f(c),c), q(f(c),f(c)), ...\}$

- baza Herbrand este infinită

$SH_{S_2} = \{p(f(c)), \neg q(c,c), p(f(f(c))), \neg q(c, f(c)), \neg q(f(c), c),$
$\neg q(f(c), f(c)), ...\}$ - sistemul Herbrand este infinit

Un model Herbrand al mulțimii S_2 este $I_2 =< D_2, m_2 >$, unde:

$D_2 = H_{S_2}$ și m_2 este definită astfel:

- $m_2(c) = c;$
- $m_2(f)(c) = f(c)$, $m_2(f)(f(c)) = f(f(c))$, ...
- $m_2(p)(f(c)) = T$,

  $m_2(q)(c,c) = F$,

  $m_2(p)(f(f(c))) = T$,

  $m_2(q)(f(c), c) = F$,

  $m_2(q)(c, f(c)) = F$,

  $m_2(q)(f(c), f(c)) = F$,

  ...

Acest model este identificat prin submulțimea $\{p(f(c)), p(f(f(c))), \ldots\}$

Se observă că existența simbolurilor de funcții în mulțimea inițială de formule are ca și consecință faptul că universul Herbrand și sistemul Herbrand asociate sunt infinite.

Are loc: $v^{I_2}(S_2)$=T, deci S_2 este o mulțime consistentă.

Teorema 2.8. (teorema lui Herbrand)

Fie S o mulțime de formule predicative fără cuantificatori. S admite un model dacă și numai dacă SH_S (sistemul Herbrand asociat lui S) admite un model.

Teorema 2.9.

O mulțime S de formule predicative fără cuantificatori este inconsistentă dacă și numai dacă sistemul Herbrand asociat mulțimii S, adică SH_S. este inconsistent.

Observație:

Problema verificării inconsistenței unei mulțimi de formule predicative s-a redus la verificarea inconsistenței unei mulțimi (finite sau infinite) de formule în calculul propozițional conform:

"mulțimea S este inconsistentă *dacă și numai dacă* mulțimea SH_{S1} este inconsistentă, unde $S1 = S^c$ ".

Teorema lui Herbrand sugerează o procedură de demonstrare prin respingere în calculul predicatelor bazată pe următorul rezultat:

$U_1, U_2, \ldots, U_n \vdash V$ *dacă și numai dacă*

mulțimea $S = \{U_1^c, U_2^c, \ldots, U_n^c, (\neg V)^c\}$ este inconsistentă

dacă și numai dacă mulțimea SH_S este inconsistentă.

Algoritmul următor verifică dacă are loc deducția: $U_1, U_2, \ldots, U_n \vdash V$.

Algoritmul bazat pe teorema lui Herbrand:

Date de intrare: $U_1, U_2, \ldots, U_n$, V - formule predicative

Date de ieșire: mesaj: "are loc deducția: $U_1, U_2, \ldots, U_n \vdash V$" sau

"nu are loc deducția: $U_1, U_2, \ldots, U_n \vdash V$"

Se construiește mulțimea $S = \{U_1^c, U_2^c, \ldots, U_n^c, (\neg V)^c\}$;

Se construiește SH_S *(sistemul Herbrand asociat mulțimii S);*

dacă (SH_S este finit)

  atunci

    dacă (SH_S este inconsistent)

      atunci scrie "*are loc* $U_1, U_2, \ldots, U_n \vdash V$ "; STOP;

      altfel scrie " *nu are loc* $U_1, U_2, \ldots, U_n \vdash V$ "; STOP;

    sf_dacă

  altfel // $SH_S = \{Z_1, Z_2, \ldots, Z_m, \ldots\}$ infinit

    k=1;

    cât_timp ($Z_1 \wedge \ldots \wedge Z_k$ este consistentă) **execută** // (**)

      k=k+1;

    sf_cât_timp

    scrie " *nu are loc* $U_1, U_2, \ldots, U_n \vdash V$ "; STOP;

sf_dacă

Sf_algoritm

Acest algoritm este semi-decizional și în cazul mulțimii SH_S infinite consistente (**) nu se oprește.

Pentru implementarea propriu-zisă, condiția ciclului *cât_timp* trebuie să conțină și o restricție asupra numărului de iterații k pentru evitarea ciclării infinite. Dacă ieșirea din ciclu se face prin atingerea numărului maxim de iterații permise, atunci nu putem lua o decizie asupra consistenței sau inconsistenței mulțimii SH_S, deci nu putem decide dacă are loc sau nu deducția $U_1, U_2, \ldots, U_n \vdash V$.

Exemplul 2.20.

Folosind procedura bazată pe teorema lui Herbrand să se verifice validitatea formulei $U = (\forall x)(p(x) \to q(x)) \to ((\forall x)p(x) \to (\forall x)q(x))$.

Utilizăm rezultatele din Exemplul 2.8. unde $A = \neg U$.

$A_1^c = (\neg p(x) \vee q(x)) \wedge p(y) \wedge \neg q(a)$

$A_2^c = (\neg p(x) \vee q(x)) \wedge p(y) \wedge \neg q(f(x))$

Fie mulțimile de clauze: $S_1 = \{\neg p(x) \vee q(x), p(y), \neg q(a)\}$ și

$S_2 = \{ \neg p(x) \vee q(x), p(y), \neg q(f(x)) \}$

Vom folosi procedura prin respingere bazată pe teorema lui Herbrand, conform căreia:

"U este validă *dacă și numai dacă* sistemul Herbrand SH_S este o mulțime inconsistentă", unde S este S_1 sau S_2.

Avem două variante corespunzătoare celor două mulțimi S_1 și S_2, obținute din două forme normale Skolem ale formulei A.

Varianta 1: utilizăm mulțimea $S_1 = \{ \neg p(x) \vee q(x), p(y), \neg q(a) \}$

$H_{S_1} = \{a\}$,

$SH_{S_1} = \{ \neg p(a) \vee q(a), p(a), \neg q(a) \}$

Considerăm conjuncția formulelor din sistemul Herbrand, pe care o aducem la forma normală disjunctivă:

$X = (\neg p(a) \vee q(a)) \wedge p(a) \wedge \neg q(a) \equiv$

$\equiv (\underline{\neg p(a)} \wedge \underline{p(a)} \wedge \neg q(a)) \vee (\underline{q(a)} \wedge p(a) \wedge \underline{\neg q(a)})$ FND cu două cuburi inconsistente, deci X este inconsistentă și astfel sistemul Herbrand SH_{S_1} este inconsistent.

Concluzionăm că A este inconsistentă și U este o formulă validă.

Varianta 2: utilizăm mulțimea $S_2 = \{ \neg p(x) \vee q(x), p(y), \neg q(f(x)) \}$

$H_{S_2} = \{c, f(c), f(f(c)), \ldots\}$,

$SH_{S_2} = \{ \neg p(c) \vee q(c), p(c), \neg q(f(c)), \neg p(f(c)) \vee q(f(c)),$
$p(f(c)), \neg q(f(f(c))), \ldots \}$

Universul Herbrand și sistemul Herbrand asociate lui S_2 sunt infinite.

Considerăm submulțimea finită a lui SH_{S_2}:

$S_f = \{ p(f(c)), \neg q(f(c)), \neg p(f(c)) \vee q(f(c)) \}$

Fie X conjuncția formulelor mulțimii S_f:

$X = p(f(c)) \wedge \neg q(f(c)) \wedge (\neg p(f(c)) \vee q(f(c))) \equiv$

$\equiv (\underline{p(f(c))} \wedge \neg q(f(c)) \wedge \underline{\neg p(f(c))}) \vee (p(f(c)) \wedge \underline{\neg q(f(c))} \wedge \underline{q(f(c))})$

S-a obținut FND cu două cuburi inconsistente (fiecare cub are o pereche de literali opuși). Deci X este inconsistentă și astfel mulțimea SH_{S_2} este inconsistentă având submulțimea S_f inconsistentă.

În concluzie formula A este inconsistentă și U este o formulă validă.

Se observă că indiferent de forma normală Skolem aleasă, se poate demonstra inconsistența sistemului Herbrand corespunzător. Recomandarea este de a utiliza forma normală Skolem fără simboluri de funcții, pentru a obține un sistem Herbrand finit.

Exemplul 2.21.

Folosind procedura Herbrand să se verifice dacă mulțimea $S = \{ p(x) \rightarrow q(x) \vee r(x), q(y) \rightarrow r(y), r(a), \neg p(a) \}$ este inconsistentă.

Aducem formulele din mulțimea S la forma normală clauzală:

$S = \{ \neg p(x) \vee q(x) \vee r(x), \neg q(y) \vee r(y), r(a), \neg p(a) \}$

$H_S = \{a\}$ - univers Herbrand finit

$BH_S = \{ p(a), q(a), r(a) \}$ - bază Herbrand finită

$SH_S =$

$\{ C1 = \neg p(a) \vee q(a) \vee r(a), C2 = \neg q(a) \vee r(a), C3 = r(a), C4 = \neg p(a) \}$

Fie X conjuncția clauzelor din sistemul Herbrand:

$X = \underline{(\neg p(a) \vee q(a) \vee r(a))} \wedge (\neg q(a) \vee r(a)) \wedge r(a) \wedge \underline{\neg p(a)}$

Simplificăm formula X aplicând legea absorbției de două ori astfel:

$X \equiv \underline{(\neg q(a) \vee r(a))} \wedge \underline{r(a)} \wedge \neg p(a) \equiv r(a) \wedge \neg p(a)$ --- FND

Forma normală disjunctivă obținută nu este o disjuncție de cuburi inconsistente, deci X este o formulă contingentă. Astfel, SH_S este o mulțime consistentă și conform Teoremei 2.8. mulțimea S este consistentă.

Un model Herbrand pentru mulțimea S este identificat prin submulțimea $\{r(a)\} \subset BH_S$ reprezentând:

$I_1 = <D, m_1>, D = H_S = \{a\}$,

$m_1(r)(a) = T, m_1(p)(a) = F, m_1(q)(a) = F$.

Are loc $v^{I_1}(S) = T$.

Un alt model Herbrand al mulțimii S este identificat prin submulțimea $\{r(a), q(a)\} \subset BH_S$ și reprezintă:

$I_2 = <D, m_2>, D = H_S = \{a\}$,

$m_2(r)(a) = T, m_2(p)(a) = F, m_2(q)(a) = T$.

Are loc $v^{I_2}(S) = T$.

3. METODA TABELELOR SEMANTICE ÎN LOGICILE CLASICE

Această metodă, introdusă de R. Smullyan în anul 1968, este o metodă de demonstrare semantică, prin respingere, foarte eficientă şi atractivă, fiind abordată de mulţi cercetători [1, 16, 21, 28, 34, 39, 40].

Există două forme echivalente ale acestei metode:

- formulele apar prefixate de valorile logice T(true) şi F(false), formă care a putut fi uşor generalizată pentru logicile multivalente şi cele modale.
- formulele sunt fără semn, această formă fiind cea prezentată în continuare.

Metoda are în denumirea ei termenul "semantic", deoarece se bazează pe consideraţii semantice şi încearcă să construiască modelele unei formule date. Ideea de bază a acestei metode constă în descompunerea formulei iniţiale în subformule până la nivel de literali, cu scopul determinării modelelor sale. Demonstraţia prin respingere pentru o formulă U, constă în a căuta modelele pentru $\neg U$ şi a arăta că acestea nu există.

3.1. Abordarea clasică a metodei tabelelor semantice

Formulele predicative se împart în patru clase în funcţie de conectiva/cuantificatorul principal astfel:

- clasa α conţine ***formule de tip conjunctiv***:
  $A \wedge B$, $\quad \neg(A \vee B) \equiv \neg A \wedge \neg B$, $\quad \neg(A \rightarrow B) \equiv A \wedge \neg B$
- clasa β conţine ***formule de tip disjunctiv***:
  $A \vee B$, $\quad \neg(A \wedge B) \equiv \neg A \vee \neg B$, $\quad A \rightarrow B \equiv \neg A \vee B$
- clasa γ conţine ***formule cuantificate universal***:
  $(\forall x)\, A(x)$, $\quad \neg(\exists x)\, A(x) \equiv (\forall x) \neg A(x)$
- clasa δ conţine ***formule cuantificate existenţial***:
  $(\exists x)\, A(x)$, $\quad \neg(\forall x)\, A(x) \equiv (\exists x) \neg A(x)$

Corespunzător există patru tipuri de **reguli de descompunere** a formulelor:

1. **regula α**

$$
\begin{array}{ccc}
A \wedge B & \neg(A \vee B) \equiv \neg A \wedge \neg B & \neg(A \rightarrow B) \equiv A \wedge \neg B \\
| & | & | \\
A & \neg A & A \\
| & | & | \\
B & \neg B & \neg B
\end{array}
$$

O formulă conjunctivă este satisfăcută dacă toate componentele sale sunt satisfăcute. Descompunerea este simbolizată grafic prin adăugarea pe aceeaşi ramură cu formula iniţială a subformulelor sale componente.

2. **regula β**

$$
\begin{array}{ccc}
A \vee B & \neg(A \wedge B) \equiv \neg A \vee \neg B & A \rightarrow B \equiv \neg A \vee B \\
/ \quad \backslash & / \quad \backslash & / \quad \backslash \\
A \quad B & \neg A \quad \neg B & \neg A \quad B
\end{array}
$$

O formulă disjunctivă este descompusă, prin ramificare, în cele două subformule componente. Consistenţa unei formule disjunctive se reduce la consistenţa a cel puţin uneia dintre cele două subformule ale sale.

3. **regula γ**

$$
\begin{array}{cc}
(\forall x)\, A(x) & \neg(\exists x)\, A(x) \equiv (\forall x) \neg A(x) \\
| & | \\
A(c_1) & \neg A(c_1) \\
| & | \\
\ldots & \ldots \\
| & | \\
A(c_n) & \neg A(c_n) \\
| & | \\
(\forall x)\, A(x) \text{ - copie formulă} & \neg(\exists x)\, A(x) \text{ - copie formulă}
\end{array}
$$

- unde $c_1, \ldots, c_n$ sunt toate constantele existente pe ramură. Dacă nu există nicio constantă pe ramură se introduce o constantă nouă pentru instanţiere.

Această regulă rezidă din faptul că o formulă cuantificată universal este logic echivalentă cu conjuncţia tuturor instanţelor sale care utilizează constantele din domeniul de interpretare. Copia formulei descompuse este utilă pentru instanţieri ulterioare în cazul în care se introduc noi constante în domeniul de interpretare.

4. regula δ

$$(\exists\, x)\, A(x) \qquad\qquad \neg(\forall\, x)\, A(x) \equiv (\exists\, x) \neg A(x)$$
$$|\qquad\qquad\qquad\qquad |$$
$$A(a) \qquad\qquad\qquad \neg A(a)$$

- unde a este o constantă nou introdusă în domeniul de interpretare.

O formulă cuantificată existențial se reduce astfel la o instanță a sa, obținută prin introducerea unei noi constante.

Definiția 3.1. (construcția tabelei semantice)

Unei formule U din logica predicatelor de ordinul I, i se poate asocia o ***tabelă semantică***, care este de fapt un arbore binar ce conține în nodurile sale formule și se construiește astfel:

1. rădăcina arborelui este etichetată cu formula U;
2. fiecare ramură a arborelui ce conține o formulă nedescompusă (care nu este literal) va fi extinsă cu subarborele corespunzător regulii de descompunere care se aplică formulei;
3. extinderea unei ramuri se încheie în două situații:
   a) dacă pe ramură apare o formulă şi negația sa;
   b) dacă au fost descompuse toate formulele de pe acea ramură sau prin aplicarea regulilor de descompunere nu se obțin formule noi pe acea ramură.

Definiția 3.2.

1. O ***ramură*** a tabelei semantice se numește ***închisă*** (simbolizată grafic prin $\otimes$) dacă aceasta conține o formulă și negația sa, în caz contrar ***ramura*** se numește ***deschisă*** (simbolizată grafic prin $\circ$).
2. O ***ramură*** a tabelei semantice se numește ***completă*** dacă este fie închisă, fie toate formulele de pe acea ramură au fost descompuse.
3. O ***tabelă*** semantică se numește ***închisă*** dacă toate ramurile sale sunt închise. Dacă o ***tabelă*** semantică are cel puțin o ramură deschisă, atunci aceasta se numește ***deschisă***.
4. O ***tabelă*** semantică se numește ***completă*** dacă toate ramurile sale sunt complete.

Procesul de construire a unei tabele semantice este unul nedeterminist deoarece, la un moment dat se pot alege mai multe ramuri pentru extindere, iar în cadrul unei ramuri pot exista mai multe formule care trebuie descompuse. Astfel, unei formule i se pot asocia mai multe tabele semantice, acestea fiind echivalente.

Pentru a obține tabele semantice cât mai simple (mai puțin ramificate) se recomandă:

- utilizarea regulilor de tip α și δ înaintea regulilor de tip β care realizează o ramificare;

- utilizarea regulilor de tip δ (care introduc noi constante) înaintea regulilor de tip γ care utilizează toate constantele de pe ramura respectivă.

Formulele de pe aceeaşi ramură a unei tabele semantice sunt legate între ele prin conectiva logică $\wedge$, iar ramificarea corespunde conectivei logice $\vee$. Tabela semantică asociată unei formule propoziționale este o reprezentare grafică a formei sale normale disjunctive. Fiecare ramură reprezintă un cub (conjuncția tuturor literalilor de pe ramură), iar arborele este disjuncția tuturor ramurilor sale.

Dacă o formulă are asociată o tabelă semantică completă deschisă, atunci formula este consistentă, iar fiecare ramură deschisă a tabelei furnizează un model parțial (prin atribuirea valorii de adevăr "true" fiecărui literal de pe ramură) pentru formula respectivă. Chiar dacă toate ramurile unei tabele semantice sunt deschise, nu există garanția că formula este validă. Dacă numărul modelelor obținute din tabela semantică deschisă este egal cu numărul tuturor interpretărilor posibile, atunci formula este validă, în caz contrar aceasta este doar contingentă.

O tabelă semantică închisă asociată unei formule indică faptul că formula este inconsistentă, adică nu există nicio interpretare în care formula să fie evaluată ca adevărată.

Se observă că este mai uşor să se verifice inconsistența unei formule decât validitatea sa, din tabela semantică asociată. Din motive de eficiență, metoda tabelelor semantice este utilizată ca metodă de demonstrare prin respingere conform teoremelor următoare.

Teorema 3.1. [27]
(corectitudinea și completitudinea metodei tabelelor semantice)

O formulă U este tautologie (teoremă) dacă şi numai dacă există o tabelă semantică închisă pentru formula $\neg U$.

Teorema 3.2.

$U_1,\dots,U_n \models V$ *(echivalent cu* $U_1,\dots,U_n \vdash V$*) dacă şi numai dacă există o tabelă semantică închisă pentru formula* $U_1 \wedge \dots \wedge U_n \wedge \neg V$.

Pentru cazul propozițional, tabela semantică asociată negației unei formule este finită, deci tabela semantică este fie închisă, fie deschisă și astfel se poate decide dacă formula respectivă este sau nu tautologie.

Pentru cazul logicii predicatelor de ordinul I, arborele poate fi infinit datorită combinării regulilor de tip γ și δ. *Calculul predicatelor de ordinul I nu este decidabil, este doar semi-decidabil.* Dacă arborele asociat negației unei formule predicative este finit, atunci se poate decide dacă formula respectivă este tautologie sau nu, dar dacă arborele este infinit, atunci nu se poate decide nimic asupra validității formulei.

Exemplul 3.1.

Construiți două tabele semantice diferite pentru formula:
$U=(p\vee q)\wedge\neg(q\to r)\wedge(p\to q\wedge r)$.

Cele două tabele semantice se obțin prin ordinea diferită de aplicare a regulilor de descompunere asupra subformulelor formulei inițiale.

Tabela semantică 1

```
(p∨q)∧¬(q→r)∧(p→q∧r) (1)
     |--- regula α pt. (1)
   p∨q (2)
     |
  ¬(q→r) (3)
     |
   p→q∧r (4)
     |--- regula α pt. (3)
     q
     |
     ¬r
    /     \--- β pt. (2)
   p        q
  / \      / \ --- β pt. (4)
¬p  q∧r  ¬p  q∧r (5)
⊗    |   ○    |--- α pt. (5)
     q        q
     |        |
     r        r
     ⊗        ⊗
```

Tabela semantică 2

```
(p∨q)∧¬(q→r)∧(p→q∧r) (1)
          |--- regula α pt. (1)
         p∨q (2)
          |
        ¬(q→r) (3)
          |
        p→q∧r (4)
        /          \--- regula β pt. (2)
       p             q
β pt. (4) --/  \      /  \--- β pt. (4)
     ¬p    q∧r     ¬p    q∧r (5)
     ⊗      |       |     |--- α pt. (3)
            q       q     q
            |       |     |
            ¬r      ¬r    ¬r
α pt. (5) --|       ○     |--- α pt. (5)
            q             q
            |             |
            r             r
            ⊗             ⊗
```

Se observă că Tabela 1 este mai simplă (mai puțin ramificată) decât Tabela 2. Tabela 1 s-a obținut prin descompunerea subformulelor conjunctive înaintea celor disjunctive. Cele două tabele semantice sunt echivalente, reprezentând grafic forme normale disjunctive, logic echivalente cu U.

Forma normală disjunctivă simplificată obținută prin eliminarea cuburilor inconsistente corespunzătoare ramurilor închise este: $\neg p\wedge q\wedge\neg r$. Acest cub corespunde ramurii deschise din fiecare tabelă semantică și furnizează unicul model al formulei contingente U:

$i:\{p,q,r\}\to\{T,F\}$, $i(p)$=F, $i(q)$=T, $i(r)$=F și astfel $i(U)$=T .

Exemplul 3.2.

Să se construiască o tabelă semantică asociată formulei:

$U=(q\wedge r\to p)\to(p\to r)\wedge q$

și să se decidă ce fel de formulă este. În cazul consistenței să se furnizeze toate modelele formulei.

Tabela semantică atașată formulei U este simbolizată grafic prin arborele de mai jos.

```
                  U=(q∧r→p)→(p→r)∧q   (1)
                        /           \ --- regula β pt. (1)
                ¬(q∧r→p) (2)       (p→r)∧q   (3)
regula α pt. (2) --- |                 |--- regula α pt. (3)
                   q∧r (4)           p→r (5)
                     |                 |
                    ¬p                 q
regula α pt. (4) --- |                /  \ --- regula β pt. (5)
                     q              ¬p    r
                     |              ○     ○
                     r
                     ○
```

Formula U are atașată o tabelă semantică completă și deschisă (toate cele trei ramuri sunt deschise), deci este consistentă.

$\mathrm{FND}(U)=(\neg p\wedge q\wedge r)\vee(q\wedge\neg p)\vee(q\wedge r)\equiv(q\wedge\neg p)\vee(q\wedge r)$

Forma simplificată s-a obținut prin aplicarea legii absorbției. Prima ramură este acoperită de oricare dintre celelalte două ramuri.

Cele două cuburi din FND simplificată corespund celor două ramuri din dreapta ale tabelei semantice.

Ramura reprezentată de cubul $q\wedge\neg p$ furnizează două modele (literalului r care lipsește i se poate atribui atât valoarea de adevăr T, cât și F) pentru formula U:

i1:$\{p,q,r\}\to\{T,F\}$, i1(p)=F, i1(q)=T, i1(r)=T
i2:$\{p,q,r\}\to\{T,F\}$, i2(p)=F, i2(q)=T, i2(r)=F

Ramura reprezentată de cubul $q\wedge r$ furnizează două modele pentru formula U:

i3:$\{p,q,r\}\to\{T,F\}$, i3(p)=T, i3(q)=T, i3(r)=T
i4:$\{p,q,r\}\to\{T,F\}$, i4(p)=F, i4(q)=T, i4(r)=T

Deoarece i1=i4, formula U are doar 3 modele: i1, i2 şi i3. Astfel, i1(U)=i2(U)=i3(U)=T şi concluzionăm că U nu este o formulă validă ci doar contingentă.

Exemplul 3.3.

Să se demonstreze că formula $U=(\neg B \to \neg A) \to ((\neg B \to A) \to B)$ este tautologie utilizând metoda tabelelor semantice.

Aplicăm această metodă ca metodă prin respingere, deci se construieşte tabela semantică asociată formulei $\neg U$.

$\neg U = \neg((\neg B \to \neg A) \to ((\neg B \to A) \to B))$ (1)

| --- regula α pentru (1)

$\neg B \to \neg A$ (2)

|

$\neg((\neg B \to A) \to B)$ (3)

| --- regula α pentru (3)

$\neg B \to A$ (4)

|

$\neg B$

/ \ --- regula β pentru (2)

/ \

B $\neg A$

ramură închisă ⊗ / \ --- regula β pentru (4)

B A

ramură închisă ⊗ ⊗ ramură închisă

Cele trei ramuri ale tabelei semantice sunt închise deoarece conţin perechi de literali opuşi: $(\neg B, B)$, $(\neg A, A)$, deci formula $\neg U$ nu are niciun model, fiind astfel o formulă inconsistentă. Conform teoremei de corectitudine şi completitudine a metodei concluzionăm că U este o tautologie.

Tabela semantică corespunde formei normale disjunctive a formulei $\neg U$. FND($\neg U$) $=(B \wedge \neg B) \vee (B \wedge \neg A \wedge \neg B) \vee (A \wedge \neg A \wedge \neg B)$, fiecare cub reprezentând o ramură a arborelui binar. Toate cuburile sunt inconsistente, deci $\neg U$ este inconsistentă şi astfel U este o formulă validă.

Exemplul 3.4.

Utilizând metoda tabelelor semantice să se verifice dacă are loc deducţia: $p \wedge (q \to r), r \wedge s \to t, u \to s \wedge \neg t \vdash p \wedge q \to \neg u$.

Aplicăm Teorema 3.2. Se construieşte tabela semantică asociată formulei X obţinută prin conjuncţia ipotezelor şi negaţia concluziei.

$X = p \wedge (q \to r) \wedge (r \wedge s \to t) \wedge (u \to s \wedge \neg t) \wedge \neg(p \wedge q \to \neg u)$ (1)

| --- α pt. (1), conjuncţie cu 5 componente

p

|

$q \to r$ (2)

|

$r \wedge s \to t$ (3)

|

$u \to s \wedge \neg t$ (4)

|

$\neg(p \wedge q \to \neg u)$ (5)

| --- regula α pentru (5)

$p \wedge q$ (6)

|

u

| --- regula α pentru (6)

p

|

q

/ \ --- regula β pentru (2)

$\neg q$ r

⊗ / \ --- regula β pentru (4)

$\neg u$ $s \wedge \neg t$ (7)

⊗ | --- regula α pentru (7)

s

|

$\neg t$

/ \ --- regula β pentru (3)

(8) $\neg(r \wedge s)$ t

regula β pentru (8) --- / \ ⊗

⊗ $\neg r$ $\neg s$ ⊗

Tabela semantică asociată formulei X este închisă, deci X este inconsistentă şi conform Teoremei 3.2, are loc deducţia din enunţ.

Exemplul 3.5.

Fie formula predicativă închisă $U=(\exists x)A(x) \to (\forall x)A(x)$. Dorim să verificăm dacă această formulă este tautologie în logica predicatelor de ordinul I.

Se construieşte tabela semantică ataşată formulei $\neg U$.

$\neg U = \neg((\exists x)A(x) \rightarrow (\forall x)A(x))$ (1)

| --- regula α pentru (1)

$(\exists x)A(x)$ (2)

|

$\neg(\forall x)A(x)$ (3)

| --- regula δ pentru (2)

$A(a)$

| --- regula δ pentru (3)

$\neg A(b)$

o ramură deschisă

S-a obținut o tabelă semantică completă și deschisă, deci formula U nu este tautologie. Există o infinitate de anti-modele pentru U, cu domenii de interpretare finite și infinite.

Un anti-model pentru U, cu domeniul de interpretare infinit este următorul:

$I_1 = < D_1, m_1 >$ unde $D_1 = \mathrm{N}$ (mulțimea numerelor naturale)

$m_1(A)$: $\mathrm{N} \rightarrow \{T,F\}$, $m_1(A)(x)$: "$x \vdots 2$"

$v^{I_1}(U) = v^{I_1}((\exists x)A(x) \rightarrow (\forall x)A(x)) =$

$= v^{I_1}((\exists x)A(x)) \rightarrow v^{I_1}((\forall x)A(x)) =$

$= (\exists x)_{x\in \mathrm{N}}(x \vdots 2) \rightarrow (\forall x)_{x\in \mathrm{N}}(x \vdots 2) = \mathrm{T} \rightarrow \mathrm{F} = \mathrm{F}$

Tabela semantică precedentă furnizează pentru U anti-modelul având domeniul de interpretare cu cel mai mic număr de elemente. Un model pentru $\neg U$, care este anti-model pentru U, se obține din ramura deschisă astfel:

$I_2 = < D_2, m_2 >$ unde:

$D_2 = \{a,b\}$ – constantele de pe ramură și $m_2(A)$:$\{a,b\} \rightarrow \{T,F\}$.

Notăm predicatul asociat simbolului de predicat A tot cu A și alegem de pe ramură:

$A(a)$=T și $A(b)$=F, adică $\neg A(b)$=T.

Din tabela semantică se obține $v^{I_2}(\neg U) = \mathrm{T}$ și astfel $v^{I_2}(U) = \mathrm{F}$.

Pentru a verifica rezultatul obținut, evaluăm formula U în interpretarea I_2 utilizând observația conform căreia o formulă cuantificată universal/existențial este înlocuită cu conjuncția/disjuncția instanțelor acesteia, folosind toate elementele domeniului finit de interpretare.

$v^{I_2}(U) = v^{I_2}((\exists x)A(x) \rightarrow (\forall x)A(x)) = v^{I_2}((\exists x)A(x)) \rightarrow v^{I_2}((\forall x)A(x)) =$

$= A(a) \vee A(b) \rightarrow A(a) \wedge A(b) = \mathrm{T} \vee \mathrm{F} \rightarrow \mathrm{T} \wedge \mathrm{F} = \mathrm{T} \rightarrow \mathrm{F} = \mathrm{F}$.

Exemplul 3.6.

Folosind metoda tabelelor semantice verificați validitatea formulei:

$U = (\exists x)A(x) \wedge (\exists x)B(x) \rightarrow (\exists x)(A(x) \wedge B(x))$.

Tabela semantică asociată formulei $\neg U$ este simbolizată grafic astfel:

$\neg U = \neg((\exists x)A(x) \wedge (\exists x)B(x) \rightarrow (\exists x)(A(x) \wedge B(x)))$ (1)

| --- regula α pentru (1)

$(\exists x)A(x) \wedge (\exists x)B(x)$ (2)

|

$\neg((\exists x)(A(x) \wedge B(x))) \equiv (\forall x)(\neg A(x) \vee \neg B(x))$ (3)

| --- regula α pentru (2)

$(\exists x)A(x)$ (4)

|

$(\exists x)B(x)$ (5)

| --- regula δ pentru (4), ***a*** - constantă nouă

$A(a)$

| --- regula δ pentru (5), ***b*** - constantă nouă

$B(b)$

| --- regula γ pt. (3), utilizare constante ***a,b***

$\neg A(a) \vee \neg B(a)$ (6)

|

$\neg A(b) \vee \neg B(b)$ (7)

|

$(\forall x)(\neg A(x) \vee \neg B(x))$ copia formulei (3)

/ \ --- regula β pentru (6)

$\neg A(a)$ $\neg B(a)$

ramură închisă ⊗ / \ --- regula β pentru (7)

$\neg A(b)$ $\neg B(b)$

ramură deschisă o ⊗ ramură închisă

S-a obținut o tabelă semantică completă (deoarece nu mai există formule care prin descompunere să furnizeze formule noi) și deschisă. Copia formulei (3), prin descompunere, nu generează formule noi, deoarece nu au fost introduse noi constante de la instanțierea precedentă a formulei (3).

Existența unei ramuri deschise indică faptul că formula $\neg U$ este consistentă, deci U nu este validă.

Modelul formulei $\neg U$ furnizat de ramura deschisă va fi o interpretare care evaluează formula U ca falsă.
Construim din tabela semantică modelul pentru $\neg U$ (anti-model pentru U), având domeniul de interpretare cu cel mai mic număr de elemente.

$I_1 = < D_1, m_1 >$, $D_1 = \{a,b\}$ - domeniul de interpretare

(constantele de pe ramură)

$m_1(A)$: $D_1 \to \{T,F\}$, $m_1(A)(a)$=T, $m_1(A)(b)$=F

$m_1(B)$: $D_1 \to \{T,F\}$, $m_1(B)(a)$=F, $m_1(B)(b)$=T .

Au loc: $v^{I_1}(\neg U) = T$ și $v^{I_1}(U) = F$.

I_1 este un model pentru $\neg U$ și un anti-model pentru U.

Un alt anti-model pentru formula U este interpretarea: $I_2 = < D_2, m_2 >$.
D_2 = mulțimea locuitorilor orașului Cluj-Napoca – domeniul de interpretare

$m_2(A)$: $D_2 \to \{T,F\}$, $m_2(A)(x)$: "*x este brunet*"

$m_2(B)$: $D_2 \to \{T,F\}$, $m_2(B)(x)$: "*x este blond*"

$$v^{I_2}(U) = v^{I_2}((\exists x)A(x) \wedge (\exists x)B(x)) \to v^{I_2}((\exists x)(A(x) \wedge B(x))) =$$
$$= v^{I_2}((\exists x)A(x)) \wedge v^{I_2}((\exists x)B(x)) \to v^{I_2}((\exists x)(A(x) \wedge B(x))) =$$
$$= (\exists x)_{x \in D_2} \text{"x este brunet"} \wedge (\exists x)_{x \in D_2} \text{"x este blond"} \to$$
$$\to (\exists x)_{x \in D_2} \text{"x este brunet} \wedge \text{x este blond"} =$$
$$= T \wedge T \to F = T \to F = F.$$

Exemplul 3.7.

Fie formula închisă $U = (\exists y)(\forall x)p(x,y)$. Verificăm validitatea formulei U construind tabela semantică asociată negației sale.

După cum se observă din tabela următoare formula $\neg U$ are asociată o tabelă semantică infinită deoarece formula cuantificată universal se va multiplica. Aceasta va genera la fiecare multiplicare o formulă cuantificată existențial de forma: $(\exists x)\neg p(x,c_i)$, unde c_i este constanta introdusă de formula existențială de la etapa precedentă.

Deoarece tabela este infinită nu putem decide asupra validității lui U. Totuși se observă că nu există posibilitatea obținerii de literali opuși pentru a închide tabela.

$\neg U = \neg((\exists y)(\forall x)p(x,y)) \equiv (\forall y)(\exists x)\neg p(x,y)$ (1)

| --- regula γ pentru (1), c_0 - constantă nouă

$(\exists x)\neg p(x,c_0)$ (2)

|

$(\forall y)(\exists x)\neg p(x,y)$ (3) copia formulei (1)

| --- regula δ pentru (2), c_1 - constantă nouă

$\neg p(c_1,c_0)$

| --- regula γ pentru (3), se utilizează c_1

$(\exists x)\neg p(x,c_1)$ (4)

| --- regula δ pentru (4), c_2 - constantă nouă

$\neg p(c_2,c_1)$

...

O interpretare foarte cunoscută, anti-model pentru formula U, este I=<D,m>, cu domeniul de interpretare D = N (mulțimea numerelor naturale), iar $m(p)$: NxN $\to$ {T,F}, $m(p)(x,y)$: "$x<y$".

Formula $(\exists y)(\forall x)p(x,y)$, evaluată în interpretarea I, se traduce în limbaj natural astfel: "*există un cel mai mare număr natural*", ceea ce este evident fals. Concluzionăm că formula dată nu este validă.

Exemplul 3.8.

Să se verifice validitatea formulei predicative închise:

$U = (\forall x)(\forall y)p(x,y) \to (\exists x)(\exists y)p(x,y)$.

$\neg U = \neg((\forall x)(\forall y)p(x,y) \to (\exists x)(\exists y)p(x,y))$ (1)

| --- regula α pentru (1)

$(\forall x)(\forall y)p(x,y)$ (2)

|

$\neg(\exists x)(\exists y)p(x,y)$ (3)

| --- regula γ pentru (2), ***a***- constantă nouă

$(\forall y)p(a,y)$ (4)

| --- regula γ pentru (4)

$p(a,a)$ instanțiere (4), utilizare ***a***

| --- regula γ pentru (3), utilizare ***a***

$\neg(\exists y)p(a,y)$ (5)

| --- regula γ pentru (5), utilizare ***a***

$\neg p(a,a)$

⊗ ramură închisă

Tabela semantică asociată formulei $\neg U$ este închisă, deci U este o formulă validă (tautologie).

Se observă că la prima aplicare a regulii de descompunere γ pentru formula (2), deoarece nu există nicio constantă în domeniul de interpretare, s-a introdus una cu care s-a făcut instanțierea. Totodată nu a mai fost nevoie de copiile formulelor universale (2), (3), (4) și (5) pentru instanțieri ulterioare, deoarece tabela s-a închis.

Exemplul 3.9.

Verificați proprietatea de distributivitate a cuantificatorului universal față de conjuncție: $(\forall x)(p(x) \wedge q(x)) \equiv (\forall x)p(x) \wedge (\forall x)q(x)$.

Această echivalență logică are loc *dacă și numai dacă*:

$\models (\forall x)(p(x) \wedge q(x)) \leftrightarrow (\forall x)p(x) \wedge (\forall x)q(x)$ *dacă și numai dacă* $\models U_1$ și $\models U_2$, unde:

$U_1 = (\forall x)(p(x) \wedge q(x)) \to (\forall x)p(x) \wedge (\forall x)q(x)$ și

$U_2 = (\forall x)p(x) \wedge (\forall x)q(x) \to (\forall x)(p(x) \wedge q(x))$.

Construim tabela semantică atașată formulei $\neg U_1$:

$\neg U_1 = \neg((\forall x)(p(x) \wedge q(x)) \to (\forall x)p(x) \wedge (\forall x)q(x))$ (1)

| --- regula α pentru (1)

$(\forall x)(p(x) \wedge q(x))$ (2)

|

$\neg((\forall x)p(x) \wedge (\forall x)q(x))$ (3)

/ \ --- regula β pentru (3)

(4) $\neg(\forall x)p(x))$ $\neg(\forall x)q(x))$ (5)

regula δ pt. (4), *a*-const --- |

$\neg p(a)$

regula γ pt. (2) --- |

(6) $p(a) \wedge q(a)$

|

$(\forall x)(p(x) \wedge q(x))$ - copie (2)

regula α pt. (6) --- |

$p(a)$

|

ramură închisă $\otimes$ $q(a)$

| --- regula δ pt. (5), *b*-const.

$\neg q(b)$

| --- regula γ pt. (2)

$p(b) \wedge q(b)$ (7)

|

$(\forall x)(p(x) \wedge q(x))$ - copie (2)

| --- regula α pt. (7)

$p(b)$

|

$q(b)$ $\otimes$ ramură închisă

Formula $\neg U_1$ este inconsistentă deoarece are asociată o tabelă semantică închisă (ramura din stânga conține literalii opuși $p(a)$ și $\neg p(a)$, iar ramura din dreapta conține literalii opuși $q(b)$ și $\neg q(b)$

Astfel, conform corectitudinii metodei are loc $\models U_1$.

Tabela semantică atașată formulei $\neg U_2$ este următoarea:

$\neg U_2 = \neg((\forall x)p(x) \wedge (\forall x)q(x) \to (\forall x)(p(x) \wedge q(x)))$ (1)

| --- regula α pentru (1)

$(\forall x)p(x) \wedge (\forall x)q(x)$ (2)

|

$\neg(\forall x)(p(x) \wedge q(x))$ (3)

| --- regula α pentru (2)

$(\forall x)p(x)$ (4)

|

$(\forall x)q(x)$ (5)

| --- regula δ pentru (3), *a*- constantă

$\neg(p(a) \wedge q(a))$ (6)

| --- regula γ pentru (4)

$p(a)$ instanțiere (4), utilizare *a*

|

$(\forall x)p(x)$ copie formula (4)

| --- regula γ pentru (5)

$q(a)$ instanțiere (5), utilizare *a*

|

$(\forall x)q(x)$ copie formula (5)

/ \ --- regula β pentru (6)

$\otimes$ $\neg p(a)$ $\neg q(a)$ $\otimes$

Tabela semantică obținută este închisă, deci $\neg U_2$ este inconsistentă și $\models U_2$.

S-a demonstrat astfel proprietatea de "distributivitate a cuantificatorului universal față de conjuncție", ceea ce era de așteptat deoarece, cuantificatorul universal este o generalizare a conjuncției.

Duala acestei proprietăți este "distributivitatea cuantificatorului existențial față de disjuncție".

Exemplul 3.10.

Verificați proprietatea de distributivitate a cuantificatorului universal față de implicație: $(\forall x)(p(x) \to q(x)) \equiv (\forall x)p(x) \to (\forall x)q(x)$.

Această echivalenţă logică are loc *dacă şi numai dacă*

$\models (\forall x)(p(x) \to q(x)) \leftrightarrow ((\forall x)p(x) \to (\forall x)q(x))$ *dacă şi numai dacă* $\models U_1$ şi $\models U_2$ unde:

$U_1 = (\forall x)(p(x) \to q(x)) \to ((\forall x)p(x) \to (\forall x)q(x))$ şi
$U_2 = ((\forall x)p(x) \to (\forall x)q(x)) \to (\forall x)(p(x) \to q(x))$.

Vom arăta că formula U_1 este tautologie, dar U_2 nu este tautologie şi astfel nu are loc proprietatea de distributivitate, ci doar semidistributivitatea cuantificatorului universal faţă de implicaţie.

$\neg U_1 = \neg((\forall x)(p(x) \to q(x)) \to ((\forall x)p(x) \to (\forall x)q(x)))$ (1)
| --- regula α pentru (1)
$(\forall x)(p(x) \to q(x))$ (2)
|
$\neg((\forall x)p(x) \to (\forall x)q(x))$ (3)
| --- regula α pentru (3)
$(\forall x)p(x)$ (4)
|
$\neg(\forall x)q(x)$ (5)
| --- regula δ pt. (5), *a*- constantă nouă
$\neg q(a)$
| --- regula γ pentru (4)
$p(a)$
| --- regula γ pentru (2)
$p(a) \to q(a)$ (6)
/ \ --- regula β pentru (6)
ramură închisă ⊗ $\neg p(a)$ $q(a)$ ⊗ramură închisă

Tabela semantică ataşată formulei $\neg U_1$ este închisă, deci $\models U_1$.

Observaţie:

Nu s-au mai adăugat în tabela semantică copiile formulelor universale (2) şi (4), deoarece tabela s-a închis, deci nu a mai fost nevoie de instanţierile acestora.

Tabela semantică asociată formulei $\neg U_2$, reprezentată grafic mai jos, este completă şi deschisă, deci formula U_2 nu este tautologie.

$\neg U_2 = \neg(((\forall x)p(x) \to (\forall x)q(x)) \to (\forall x)(p(x) \to q(x)))$ (1)
| --- regula α pentru (1)
$(\forall x)p(x) \to (\forall x)q(x)$ (2)
|
$\neg(\forall x)(p(x) \to q(x))$ (3)
|--- regula δ pt. (3), *a*- constantă nouă
$\neg(p(a) \to q(a))$ (4)
| --- regula α pentru (4)
$p(a)$
|
$\neg q(a)$
/ \ --- regula β pentru (2)
(5) $\neg(\forall x)p(x)$ $(\forall x)q(x)$ (6)
regula δ pt.(5), *b*-const. nouă --- | | --- regula γ pentru (6)
$\neg p(b)$ $q(a)$
ramură deschisă o ⊗ ramură închisă

Ramura deschisă furnizează două modele, I_1 şi I_2, pentru formula $\neg U_2$, care sunt anti-modele pentru formula U_2. Cele două interpretări I_1 şi I_2 au acelaşi domeniu de interpretare şi diferă doar prin valoarea de adevăr ce se asignează literalului $q(b)$ care nu apare pe ramura deschisă.

$I_1 = \langle D, m_1 \rangle$ şi $I_2 = \langle D, m_2 \rangle$ unde:

$D = \{a,b\}$ - domeniul de interpretare ce conţine toate constantele de pe ramura deschisă.

$m_1(p): D \to \{T,F\}$, $m_1(p)(a)=T$, $m_1(p)(b)=F$

$m_1(q): D \to \{T,F\}$, $m_1(q)(a)=F$, $m_1(q)(b)=T$

Au loc: $v^{I_1}(\neg U_2) = T$ şi $v^{I_1}(U_2) = F$.

$m_2(p): D \to \{T,F\}$, $m_2(p)(a)=T$, $m_2(p)(b)=F$

$m_2(q): D \to \{T,F\}$, $m_2(q)(a)=F$, $m_2(q)(b)=F$

Au loc: $v^{I_2}(\neg U_2) = T$ şi $v^{I_2}(U_2) = F$.

Astfel are loc doar proprietatea de semidistributivitate a cuantificatorului universal faţă de implicaţie:

$\models (\forall x)(p(x) \to q(x)) \to ((\forall x)p(x) \to (\forall x)q(x))$.

3.2. O nouă viziune asupra metodei tabelelor semantice

În lucrările [21, 34, 39], este prezentată metoda tabelelor semantice, într-o nouă viziune și este utilizată pentru generarea extensiilor clasice ale unei teorii implicite. Prezentăm în continuare bazele teoretice ale metodei tabelelor semantice conform [34], cu modificări în definirea funcției TP pentru a surprinde toate combinațiile de aplicare a conectivelor și cuantificatorilor în formulele predicative.

Definiția 3.3.

O ***tabelă semantică*** este concepută ca o mulțime având ca elemente mulțimi de literali: o ramură este reprezentată de o mulțime de literali, iar reuniunea acestor mulțimi este tabela. Această tabelă se construiește cu ajutorul unei funcții TP definită pe mulțimi de formule cu valori în mulțimi de mulțimi de literali. TP se definește recursiv folosind regulile de descompunere $\alpha, \beta, \gamma, \delta$ astfel:

TP(M) = {M}, dacă M este o mulțime de literali.
TP(M) = TP(M' $\cup$ {p}), dacă $\neg\neg p \in M$ și M'= M\{$\neg\neg p$}

regula α --- descompunerea formulelor conjunctive

TP(M) = TP(M' $\cup$ {p} $\cup$ {q}), dacă $p \wedge q \in M$ și M'= M\{ $p \wedge q$}
TP(M)= TP(M' $\cup$ {$\neg$ p} $\cup$ {$\neg$ q}), dacă $\neg(p \vee q) \in M$ și M'= M\{ $\neg(p \vee q)$}
TP(M)= TP(M' $\cup$ {p} $\cup$ {$\neg$ q}), dacă $\neg(p \rightarrow q) \in M$ și M'= M\{ $\neg(p \rightarrow q)$}

regula β --- descompunerea formulelor disjunctive

TP(M) = TP(M' $\cup$ {p}) $\cup$ TP(M' $\cup$ {q}), dacă $p \vee q \in M$ și M'=M\{$p \vee q$}
TP(M) = TP(M' $\cup$ {$\neg$ p}) $\cup$ TP(M' $\cup$ {$\neg$ q}), dacă $\neg(p \wedge q) \in M$ și M'= M\{ $\neg(p \wedge q)$}
TP(M) = TP(M' $\cup$ {$\neg$ p}) $\cup$ TP(M' $\cup$ {q}), dacă $p \rightarrow q \in M$ și M'= M\{$p \rightarrow q$}

regula δ --- descompunerea formulelor existențiale

TP(M) = TP(M' $\cup$ {A(c)}), dacă $(\exists x)A(x) \in M$, M'= M\{$(\exists x)A(x)$}
TP(M) = TP(M' $\cup$ {$\neg$ A(c)}), dacă $\neg(\forall x)A(x) \in M$, M'= M\{ $\neg(\forall x) A(x)$}

- unde c este o constantă nou introdusă în domeniul de interpretare

regula γ --- descompunerea formulelor universale

TP(M) = TP(M $\cup$ {A(c) | c $\in$ C}), dacă $(\forall x) A(x) \in M$
TP(M) = TP(M $\cup$ {$\neg$ A(c) | c $\in$ C}), dacă $\neg(\exists x)A(x) \in M$

- unde C este mulțimea constantelor din domeniul de interpretare. Dacă C=$\varnothing$ atunci se introduce o constantă nouă , **c**, deci mulțimea constantelor devine C={c}.

Definiția 3.4.

1. O ***mulțime de literali*** este ***închisă*** dacă aceasta conține doi literali complementari (opuși). În caz contrar ***mulțimea de literali*** se numește ***deschisă***.
2. O ***tabelă*** este ***închisă***, dacă fiecare element al său este o mulțime închisă de literali.
3. Dacă o tabelă are cel puțin un element care este o mulțime deschisă, atunci ***tabela*** se numește ***deschisă***.

Tabela semantică atașată unei formule corespunde formei sale normale disjunctive astfel: fiecare mulțime reprezintă conjuncția elementelor sale (cubul format din literalii mulțimii), iar tabela reprezintă disjuncția elementelor sale (mulțimilor de literali).

Fie tabela semantică $\cup_{i=1}^{m} \{\cup_{k=1}^{n_i} \{a_{ik}\}\}$, numărul ramurilor sale este m și ramurile sale $i=1,m$ au respectiv n_i elemente care sunt literali. Această tabelă corespunde unei formule aflate în forma normală disjunctivă $\vee_{i=1}^{m} \wedge_{k=1}^{n_i} a_{ik}$.

Această nouă abordare a metodei tabelelor semantice permite, prin intermediul funcției TP, care are numeroase proprietăți interesante, manevrarea eficientă a tabelelor și operațiilor cu tabele: produsul a două tabele, obținerea unei subtabele, deschiderea unei tabele, minimizarea unei tabele.

Dacă tabela rezultată este foarte stufoasă, ne interesează să o aducem la o formă minimală, prin eliminarea la fiecare pas al construcției tabelei semantice, a mulțimilor închise și a acelor mulțimi care conțin alte mulțimi din tabelă, fapt care nu modifică consistența/inconsistența mulțimii inițiale de formule.

Definiția 3.5.

Forma minimală a tabelei semantice asociată mulțimii S de formule se notează cu TS(S) și se obține astfel:
TS(S) = TP(S) \{X | X $\in$ TP(S), X -închisă *sau* $\exists$ Y $\in$ TP(S) a.î. $\varnothing \neq Y \subset X$}

Observație:

Dacă TS(S)=$\varnothing$ atunci S este o mulțime inconsistentă de formule, în caz contrar S este consistentă.

Teorema 3.3. (teorema de completitudine și corectitudine pentru TP)
Formula U este o tautologie dacă și numai dacă TP({$\neg$ *U*}) *este o tabelă semantică închisă.*

Teorema 3.4.
Fie S o mulțime finită de formule și U o formulă, atunci S |= U, adică U este consecință logică a mulțimii S, dacă și numai dacă TP(*S*$\cup${$\neg$ *U*}) *este o tabelă semantică închisă.*

$= \mathrm{TP}(\{\, \underline{p \to (q \to r)}, q, p, \neg r \,\})$ (regula β)

$= \mathrm{TP}(\{\, \neg p, q, p, \neg r \,\}) \cup \mathrm{TP}(\{\, \underline{q \to r}, q, p, \neg r \,\})$ (regula β)

$= \{\{\, \neg p, q, p, \neg r \,\}\} \cup \mathrm{TP}(\{\, \neg q, q, p, \neg r \,\}) \cup \mathrm{TP}(\{\, r, q, p, \neg r \,\}) =$

$= \{\{\, \neg p, q, p, \neg r \,\}\} \cup \{\{\, \neg q, q, p, \neg r \,\}\} \cup \{\{\, r, q, p, \neg r \,\}\} =$

$= \{\{\, \underline{\neg p}, q, \underline{p}, \neg r \,\}, \{\, \underline{\neg q}, \underline{q}, p, \neg r \,\}, \{\, \underline{r}, q, p, \underline{\neg r} \,\}\}$

Tabela semantică obţinută este închisă având toate ramurile închise (literalii opuşi sunt subliniaţi), astfel $\neg U$ este o formulă inconsistentă şi U este o tautologie, conform Teoremei 3.3.

Exemplul 3.14.

Verificaţi dacă are loc proprietatea de distributivitate a cuantificatorului existenţial faţă de implicaţie.

Această proprietate se exprimă sub forma următoarei echivalenţe logice $((\exists x)A(x) \to (\exists x)B(x)) \equiv (\exists x)(A(x) \to B(x))$ care trebuie verificată.

Similar cu Exemplele 3.9 şi 3.10, problema se reduce la verificarea validităţii formulelor U_1 şi U_2, unde:

$U_1 = ((\exists x)A(x) \to (\exists x)B(x)) \to (\exists x)(A(x) \to B(x))$

$U_2 = (\exists x)(A(x) \to B(x)) \to ((\exists x)A(x) \to (\exists x)B(x))$

Se construieşte $\mathrm{TP}(\{\neg U_1\})$:

$\mathrm{T1} = \mathrm{TP}(\{\neg U_1\}) =$

$= \mathrm{TP}(\{\neg\, (((\exists x)A(x) \to (\exists x)B(x)) \to (\exists x)(A(x) \to B(x)))\})$

- descompunerea formulei conjunctive $\neg U_1$:
  $\mathrm{T1} = \mathrm{TP}(\{\, (\exists x)A(x) \to (\exists x)B(x), \neg(\exists x)(A(x) \to B(x)) \,\})$
- descompunerea formulei disjunctive $(\exists x)A(x) \to (\exists x)B(x)$:
  $\mathrm{T1} = \mathrm{TP}(\{\, \neg(\exists x)A(x), \neg(\exists x)(A(x) \to B(x)) \,\}) \cup$
  $\cup\ \mathrm{TP}(\{\, (\exists x)B(x), \neg(\exists x)(A(x) \to B(x)) \,\})$
- descompunerea formulei existenţiale: $(\exists x)B(x)$, b-constantă nouă, pe a doua ramură şi descompunerea formulei universale $\neg(\exists x)A(x)$, a-constantă nou introdusă, deoarece nu există constante pe prima ramură:
  $\mathrm{T1} = \mathrm{TP}(\{\, \neg A(a), \neg(\exists x)A(x), \neg(\exists x)(A(x) \to B(x)) \,\}) \cup$
  $\cup\ \mathrm{TP}(\{\, B(b), \neg(\exists x)(A(x) \to B(x)) \,\})$

- descompunerea formulei universale $\neg(\exists x)(A(x) \to B(x))$ pe cele două ramuri, cu instanţierile corespunzătoare constantelor de pe fiecare ramură:
  $\mathrm{T1} = \mathrm{TP}(\{\, \neg A(a), \neg(\exists x)A(x), \underline{\neg(A(a) \to B(a))}, \neg(\exists x)(A(x) \to B(x)) \,\}) \cup$
  $\cup\ \mathrm{TP}(\{\, B(b), \underline{\neg(A(b) \to B(b))}, \neg(\exists x)(A(x) \to B(x)) \,\})$
- descompunerea formulelor conjunctive de pe cele două ramuri:
  $\mathrm{T1} = \mathrm{TP}(\{\, \underline{\neg A(a)}, \neg(\exists x)A(x), \underline{A(a)}, \neg B(a), \neg(\exists x)(A(x) \to B(x)) \,\}) \cup$
  $\cup\ \mathrm{TP}(\{\, \underline{B(b)}, A(b), \underline{\neg B(b)}, \neg(\exists x)(A(x) \to B(x)) \,\})$

T1 este o tabelă semantică închisă deoarece fiecare din cele două ramuri conţine o pereche de literali opuşi (cei subliniaţi). Astfel, formula $\neg U_1$ este inconsistentă şi U_1 este validă.

Vom demonstra că formula U_2 nu este validă, $\mathrm{TP}(\neg U_2)$ fiind deschisă.

$\mathrm{TP}(\{\neg U_2\}) = \mathrm{TP}(\{\, \underline{\neg((\exists x)(A(x) \to B(x)) \to ((\exists x)A(x) \to (\exists x)B(x)))} \,\})$ (regula α)

$= \mathrm{TP}(\{\, \underline{(\exists x)(A(x) \to B(x))}, \neg((\exists x)A(x) \to (\exists x)B(x)) \,\})$ (regula δ, a-constantă nouă)

$= \mathrm{TP}(\{\, A(a) \to B(a), \underline{\neg((\exists x)A(x) \to (\exists x)B(x))} \,\})$ (regula α)

$= \mathrm{TP}(\{\, A(a) \to B(a), \underline{(\exists x)A(x)}, \neg(\exists x)B(x) \,\})$ (regula δ, b-constantă nouă)

$= \mathrm{TP}(\{\, A(a) \to B(a), A(b), \underline{\neg(\exists x)B(x)} \,\})$ (regula γ, instanţierile formulei universale cu constantele a şi b)

$= \mathrm{TP}(\{\, \underline{A(a) \to B(a)}, A(b), \neg B(a), \neg B(b), \neg(\exists x)B(x) \,\})$ (regula β)

$= \mathrm{TP}(\{\, \neg A(a), A(b), \neg B(a), \neg B(b), \neg(\exists x)B(x) \,\}) \cup$

$\cup\ \mathrm{TP}(\{\, \underline{B(a)}, A(b), \underline{\neg B(a)}, \neg B(b), \neg(\exists x)B(x) \,\})$

S-a obţinut o tabelă semantică completă (formula cuantificată universal a fost instanţiată folosind toate constantele de pe ramură) şi deschisă, având prima ramură deschisă, iar cea de-a doua închisă (literalii subliniaţi sunt opuşi), deci U_2 nu este o formulă validă.

Concluzionăm că nu are loc proprietatea de distributivitate, ci doar semidistributivitatea cuantificatorului existenţial faţă de implicaţie:

$\models ((\exists x)A(x) \to (\exists x)B(x)) \to (\exists x)(A(x) \to B(x))$

Exemplul 3.15.

Să se verifice validitatea formulei predicative:

$$U = (\forall x)(\forall y)p(x,y) \to (\exists x)(\forall y)p(x,y).$$

Se construieşte tabela semantică asociată formulei $\neg U$ utilizând funcţia TP.

Subformulele care se descompun la fiecare pas sunt subliniate.

$$\mathrm{TP}(\{\neg U\}) = \mathrm{TP}(\{\underline{\neg((\forall x)(\forall y)p(x,y) \to (\exists x)(\forall y)p(x,y))}\})$$

(regula α)

$$= \mathrm{TP}(\{\underline{(\forall x)(\forall y)p(x,y)}, \neg(\exists x)(\forall y)p(x,y)\})$$

(regula γ, a- constantă nouă)

$$= \mathrm{TP}(\{\underline{(\forall y)p(a,y)}, (\forall x)(\forall y)p(x,y), \neg(\exists x)(\forall y)p(x,y)\})$$

(regula γ, utilizare a)

$$= \mathrm{TP}(\{p(a,a), (\forall y)p(a,y), (\forall x)(\forall y)p(x,y), \underline{\neg(\exists x)(\forall y)p(x,y)}\})$$

(regula γ, utilizare a)

$$= \mathrm{TP}(\{p(a,a), (\forall y)p(a,y), (\forall x)(\forall y)p(x,y), \underline{\neg(\forall y)p(a,y)}, \neg(\exists x)(\forall y)p(x,y)\})$$

(regula δ, b- constantă nouă)

$$= \mathrm{TP}(\{p(a,a), \underline{(\forall y)p(a,y)}, (\forall x)(\forall y)p(x,y), \neg p(a,b), \neg(\exists x)(\forall y)p(x,y)\}) =$$

(regula γ, utilizare b)

$$\mathrm{TP}(\{p(a,a), \underline{p(a,b)}, (\forall y)p(a,y), (\forall x)(\forall y)p(x,y), \underline{\neg p(a,b)}, \neg(\exists x)(\forall y)p(x,y)\})$$

Tabela semantică obţinută este formată dintr-o singură mulţime închisă de literali, având doi literali opuşi (subliniaţi în ultima tabelă), deci formula $\neg U$ este inconsistentă şi astfel U este validă.

Concluzii:

Scopul metodei tabelelor semantice este de a construi modelele unei formule propoziţionale/predicative, oferind totodată o metodă prin respingere pentru rezolvarea celor două probleme decizionale ale calculului propoziţional/predicativ.

Abordarea clasică, prin intermediul simbolizării grafice, este mai sugestivă, dar mai greu de implementat, necesitând manipularea arborilor binari.

Noua abordare, utilizând funcţia TP, care are proprietăţi interesante, permite o implementare mai uşoară şi eficientă a acestei metode de demonstrare.

În ambele abordări procesul nedeterminist de construire a tabelelor semantice poate fi eficientizat prin introducerea de euristici.

4. METODA SECVENŢELOR ŞI METODA ANTI-SECVENŢELOR

Acest capitol prezintă două sisteme axiomatice complementare: *calculul secvenţelor* şi *calculul anti-secvenţelor*, utilizate pentru a demonstra validitatea/derivabilitatea, respectiv nevaliditatea/nederivabilitatea în logicile clasice. S-au utilizat ca şi referinţe bibliografice lucrările [1, 3, 5, 22, 28, 30].

4.1. Metoda secvenţelor

Metoda secvenţelor este o metodă de demonstrare directă şi sintactică, bazată pe sistemul axiomatic numit ***calculul secvenţelor***. Este o îmbunătăţire a metodei deducţiei naturale şi este utilizată pentru a verifica validitatea/derivabilitatea în logica propoziţiilor şi logica predicatelor de ordinul I.

Definiţia 4.1. (sintaxa)

O ***secvenţă*** are forma: $X \Rightarrow Y$, unde X şi Y sunt mulţimi finite de formule predicative. X se numeşte ***antecedent***, iar Y se numeşte ***consecvent***.

În secvenţa $U_1, U_2, \ldots, U_n \Rightarrow V_1, V_2, \ldots, V_m$, *unde* $U_1, U_2, \ldots, U_n, V_1, V_2, \ldots, V_m \in F_{\mathrm{Pr}}$ au loc:

- simbolul "$\Rightarrow$" are semnificaţia "***este demonstrabil***";
- $U_1, U_2, \ldots, U_n$ se numesc ***ipoteze:***
- $V_1, V_2, \ldots, V_m$ sunt ***formulele ce trebuie demonstrate.***

Se numesc ***secvenţe de bază***, secvenţele de forma: $X, A \Rightarrow Y, A$, adică acestea conţin aceeaşi formulă, A, în antecedent şi în consecvent.

Definiţia 4.2. (semantica)

Secvenţa $U_1, U_2, \ldots, U_n \Rightarrow V_1, V_2, \ldots, V_m$ ***este adevărată*** dacă şi numai dacă formula $U_1 \wedge U_2 \wedge \ldots \wedge U_n \to V_1 \vee V_2 \vee \ldots \vee V_m$ este validă (tautologie), adică din conjuncţia ipotezelor se poate demonstra cel puţin una dintre formulele din consecvent.

Observaţii:

- Secvenţele de bază sunt secvenţe adevărate.
- Într-o secvenţă, formulele din antecedent se consideră conectate prin conjuncţie, iar cele din consecvent conectate prin disjuncţie.

Sistemul axiomatic - calculul secvenţelor

$Seq = (\Sigma_{Seq}, F_{Seq}, A_{Seq}, R_{Seq})$, unde:

- $\Sigma_{Seq} = \Sigma_{\Pr} \cup \{\Rightarrow\}$ - ***alfabetul***
- $F_{Seq} = \{X \Rightarrow Y \mid X, Y \subset F_{\Pr}\}$ - ***totalitatea secvenţelor***
- $A_{Seq} = \{X, A \Rightarrow Y, A \mid A \in F_{\Pr}, X, Y \subset F_{\Pr}\}$ -***mulţimea axiomelor,*** formată din totalitatea secvenţelor de bază
- $R_{Seq} = \{\neg_s, \neg_d, \wedge_s, \wedge_d, \vee_s, \vee_d, \rightarrow_s, \rightarrow_d, \forall_s, \forall_d, \exists_s, \exists_d\}$ - ***regulile de inferenţă***

Regulile calculului secvenţelor sunt:

	Introducerea în antecedent (reguli de stânga)	**Introducerea în consecvent (reguli de dreapta)**
$\neg$	$(\neg_s)\ \dfrac{X \Rightarrow A, Y}{X, \neg A \Rightarrow Y}$	$(\neg_d)\ \dfrac{X, A \Rightarrow Y}{X \Rightarrow \neg A, Y}$
$\wedge$	$(\wedge_s)\ \dfrac{X, A, B \Rightarrow Y}{X, A \wedge B \Rightarrow Y}$	$(\wedge_d)\ \dfrac{X \Rightarrow A, Y \qquad X \Rightarrow B, Y}{X \Rightarrow A \wedge B, Y}$
$\vee$	$(\vee_s)\ \dfrac{X, A \Rightarrow Y \qquad X, B \Rightarrow Y}{X, A \vee B \Rightarrow Y}$	$(\vee_d)\ \dfrac{X \Rightarrow A, B, Y}{X \Rightarrow A \vee B, Y}$
$\rightarrow$	$(\rightarrow_s)\ \dfrac{X \Rightarrow A, Y \qquad X, B \Rightarrow Y}{X, A \rightarrow B \Rightarrow Y}$	$(\rightarrow_d)\ \dfrac{X, A \Rightarrow B, Y}{X \Rightarrow A \rightarrow B, Y}$
$\forall$	$(\forall_s)\ \dfrac{X, A[x \leftarrow t], (\forall x)A(x) \Rightarrow Y}{X, (\forall x)A(x) \Rightarrow Y}$	$(\forall_d)\ \dfrac{X \Rightarrow A(x), Y}{X \Rightarrow (\forall x)A(x), Y}$
$\exists$	$(\exists_s)\ \dfrac{X, A(x) \Rightarrow Y}{X, (\exists x)A(x) \Rightarrow Y}$	$(\exists_d)\ \dfrac{X \Rightarrow A[x \leftarrow t], (\exists x)A(x), Y}{X \Rightarrow (\exists x)A(x), Y}$

În regulile de mai sus, X şi Y sunt mulţimi finite de formule, chiar vide, neafectate de aplicarea regulilor. t este un termen (constantă sau variabilă liberă din X, Y sau A) ales adecvat pentru instanţiere, cu scopul obţinerii de secvenţe de bază. Regulile au una sau două ***premise*** (aflate deasupra liniei orizontale) şi o ***concluzie*** (aflată sub linia orizontală).

Reguli duale: $(\forall_d)$ şi $(\exists_s)$, $(\exists_d)$ şi $(\forall_s)$, $(\vee_d)$ şi $(\wedge_s)$, $(\wedge_d)$ şi $(\vee_s)$.

Dacă aceste reguli sunt aplicate dinspre premise spre concluzie, se numesc ***reguli de inferenţă***, iar dacă sunt aplicate invers, dinspre concluzie spre premise, acestea se numesc ***reguli de reducere***. În general regulile din calculul secvenţelor se aplică ca reguli de reducere.

Metoda secvenţelor constă în simplificarea (reducerea) secvenţei iniţiale (care se doreşte a fi demonstrată) prin aplicarea succesivă a regulilor de reducere din calculul seventelor, în scopul obţinerii de secvenţe de bază.

Se construieşte un arbore binar (cu rădăcina în jos) asociat acestui proces de reducere, care are ca nod rădăcină secvenţa inţială. Este importantă redenumirea variabilelor legate din formulele secvenţei iniţiale astfel ca acestea să fie distincte.

Aplicarea unei reguli de reducere este simbolizată prin construirea de succesori (etichetaţi cu premisele regulii) ai nodului etichetat cu concluzia regulii aplicate. Construirea arborelui binar se încheie dacă nodurile frunză sunt etichetate cu secvenţe de bază, sau cu secvenţe care nu mai pot fi reduse. Nodurile frunză care conţin secvenţe de bază sunt barate.

Se observă că acest proces de reducere a secvenţelor la secvenţe mai simple este unul nedeterminist, deoarece la un moment dat există posibilitatea alegerii mai multor reguli de reducere. Recomandările următoare permit alegerea acelor variante care simplifică procesul de reducere:

- utilizarea regulilor pentru conective care au o premisă înaintea regulilor pentru conective care au două premise, pentru a se întârzia cât de mult este posibil ramificarea arborelui de reducere;
- utilizare regulilor $(\forall_d)$,$(\exists_s)$ care transformă variabilele legate în variabile libere, înaintea regulilor $(\exists_d)$,$(\forall_s)$ care instanţiază formulele respective cu termeni adecvaţi.

Teorema 4.1. (corectitudinea şi completitudinea calculului secvenţelor)

Secvenţa $X \Rightarrow Y$ este adevărată dacă şi numai dacă aceasta poate fi redusă (prin aplicarea regulilor de reducere) la secvenţe de bază.

sau

Secvenţa $X \Rightarrow Y$ este adevărată dacă şi numai dacă aceasta poate fi derivată (prin aplicarea regulilor de inferenţă) din secvenţe de bază.

Observaţie:

Cele două formulări echivalente din teorema precedentă pun în evidenţă cele două moduri posibile de utilizare a regulilor: *inversă* (ca reguli de reducere) sau *normală* (ca reguli de inferenţă).

Rezultatele teoretice introduse în teorema următoare sunt consecinţe ale Teoremei 4.1. sugerând modalitatea de a rezolva *direct* cele două probleme decizionale din logicile clasice

Teorema 4.2. Fie $U_1, U_2, ..., U_n, V \in F_{\Pr}$.

1. *Formula V este o teoremă ($\vdash V$) dacă şi numai dacă secvenţa $\Rightarrow V$ este adevărată (poate fi redusă la secvenţe de bază).*
2. $U_1, U_2, ..., U_n \vdash V$ *dacă şi numai dacă secvenţa* $U_1, U_2, ..., U_n \Rightarrow V$ *este adevărată.*

Exemplul 4.1.

Demonstrați că regula silogismului $U=(p \to q) \to ((q \to r) \to (p \to r))$ este o teoremă.

Vom construi arborele de reducere a secvenței inițiale: $\Rightarrow (p \to q) \to ((q \to r) \to (p \to r))$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\overline{\underline{p}, q \to r \Rightarrow \underline{p}, r} \qquad \dfrac{\overline{p, \underline{q} \Rightarrow \underline{q}, r} \qquad \overline{p, q, \underline{r} \Rightarrow \underline{r}}}{p, q, q \to r \Rightarrow r}\,(\to_s)}{p, p \to q, q \to r \Rightarrow r}\,(\to_s)\ pt.\ p \to q}{p \to q, q \to r \Rightarrow p \to r}\,(\to_d)}{p \to q \Rightarrow (q \to r) \to (p \to r)}\,(\to_d)$$

$$\dfrac{p \to q \Rightarrow (q \to r) \to (p \to r)}{\Rightarrow (p \to q) \to ((q \to r) \to (p \to r))}\,(\to_d)$$

Cele trei noduri frunză conțin secvențe de bază, deci $\Rightarrow U$ este o secvență adevărată și astfel formula $U=(p \to q) \to ((q \to r) \to (p \to r))$ este o teoremă.

Exemplul 4.2.

Demonstrați că formula $U=\neg(p \wedge q) \to \neg p \wedge \neg q$ nu este validă.

Arborele de reducere a secvenței inițiale: $\Rightarrow \neg(p \wedge q) \to \neg p \wedge \neg q$ este următorul:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\overline{\underline{p} \Rightarrow \underline{p}}}{\Rightarrow p, \neg p}\,(\neg_d) \quad \dfrac{q \Rightarrow p}{\Rightarrow p, \neg q}\,(\neg_d)}{\Rightarrow p, \neg p \wedge \neg q}\,(\wedge_d) \qquad \dfrac{\dfrac{p \Rightarrow q}{\Rightarrow q, \neg p}\,(\neg_d) \quad \dfrac{\overline{\underline{q} \Rightarrow \underline{q}}}{\Rightarrow q, \neg q}\,(\neg_d)}{\Rightarrow q, \neg p \wedge \neg q}\,(\wedge_d)}{\Rightarrow p \wedge q, \neg p \wedge \neg q}\,(\wedge_d)\ pt.\ p \wedge q}{\neg(p \wedge q) \Rightarrow \neg p \wedge \neg q}\,(\neg_s)}{\Rightarrow \neg(p \wedge q) \to \neg p \wedge \neg q}\,(\to_d)$$

Arborele are două noduri frunză ce conțin secvențe de bază ($p \Rightarrow p$, $q \Rightarrow q$) și două noduri frunză cu *secvențe* ($q \Rightarrow p$, $p \Rightarrow q$) ce nu pot fi reduse la secvențe de bază, deci $\Rightarrow U$ nu este adevărată *și formula U nu este validă.*

Exemplul 4.3.

Să se verifice dacă are loc proprietatea de distributivitate la stânga a implicației față de conjuncție.

Această proprietate este exprimată prin: $\models (p \to q \wedge r) \leftrightarrow (p \to q) \wedge (p \to r)$.

Se înlocuiește conectiva echivalență cu conjuncția implicațiilor în cele două sensuri:

$$\models ((p \to q \wedge r) \to (p \to q) \wedge (p \to r)) \wedge ((p \to q) \wedge (p \to r) \to (p \to q \wedge r))$$

și astfel este necesară verificarea validității formulelor:

$U_1=(p \to q \wedge r) \to (p \to q) \wedge (p \to r)$ și $U_2=(p \to q) \wedge (p \to r) \to (p \to q \wedge r)$.

Arborele de reducere a secvenței inițiale $\Rightarrow (p \to q \wedge r) \to (p \to q) \wedge (p \to r)$ corespunzătoare formulei U_1 este:

$$\dfrac{\dfrac{\dfrac{\dfrac{\overline{\underline{p} \Rightarrow q, \underline{p}} \quad \dfrac{\overline{p, \underline{q}, r \Rightarrow \underline{q}}}{p, q \wedge r \Rightarrow q}\,(\wedge_s)}{p \to q \wedge r, p \Rightarrow q}\,(\to s)}{p \to q \wedge r \Rightarrow p \to q}\,(\to_d) \qquad \dfrac{\dfrac{\overline{\underline{p} \Rightarrow r, \underline{p}} \quad \dfrac{\overline{p, q, \underline{r} \Rightarrow \underline{r}}}{p, q \wedge r \Rightarrow r}\,(\wedge_s)}{p \to q \wedge r, p \Rightarrow r}\,(\to_s)}{p \to q \wedge r \Rightarrow p \to r}}{\dfrac{p \to q \wedge r \Rightarrow (p \to q) \wedge (p \to r)}{\Rightarrow (p \to q \wedge r) \to (p \to q) \wedge (p \to r)}\,(\to_d)}\,(\wedge_d)$$

Secvența inițială s-a redus la patru secvențe de bază: $\underline{p} \Rightarrow q, \underline{p}$; $p, \underline{q}, r \Rightarrow \underline{q}$; $\underline{p} \Rightarrow r, \underline{p}$ și $p, q, \underline{r} \Rightarrow \underline{r}$, deci formula U_1 este validă.

Similar se demonstrează că formula U_2 este validă și astfel are loc proprietatea de distributivitate din enunțul problemei.

Exemplul 4.4.

Utilizând metoda secvențelor verificați dacă are loc deducția: $\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \vdash \neg p$.

Această problemă se reduce la a verifica dacă secvența $\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \Rightarrow \neg p$ este adevărată sau nu.

$$
\dfrac{\dfrac{\overline{\neg p \vee q, r \vee \neg q, \underline{p} \Rightarrow \underline{p}} \qquad \dfrac{\overline{\neg p \vee q, \underline{r}, p \Rightarrow \underline{r}} \qquad \dfrac{\dfrac{\overline{\neg q, \underline{p} \Rightarrow \underline{p}, r}}{\neg p, \neg q, p \Rightarrow r}(\neg_s) \qquad \dfrac{\overline{\underline{q}, p \Rightarrow \underline{q}, r}}{q, \neg q, p \Rightarrow r}(\neg_s)}{\neg p \vee q, \neg q, p \Rightarrow r}(\vee_s)}{\neg p \vee q, r \vee \neg q, p \Rightarrow r}(\vee_s)}{\neg p \vee q, r \vee \neg q, p \Rightarrow p \wedge r}(\wedge_d)}{\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \Rightarrow \neg p}(\neg_s), (\neg_d)
$$

Arborele de reducere obținut are patru noduri frunză care conțin secvențe de bază. Conform teoremei de corectitudine și completitudine a metodei, secvența inițială este adevărată, deci are loc: $\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \vdash \neg p$.

Exemplul 4.5.

Demonstrați validitatea formulei predicative închise: $U = (\forall x)p(x) \to p(a) \wedge p(b)$.

Construim arborele de reducere a secvenței inițiale $\Rightarrow U$.

$$
\dfrac{\dfrac{\dfrac{\overline{(\forall x)p(x), p(a) \Rightarrow p(a)}}{(\forall x)p(x) \Rightarrow p(a)}(\forall_s)[x \leftarrow a] \qquad \dfrac{\overline{(\forall x)p(x), p(b) \Rightarrow p(b)}}{(\forall x)p(x) \Rightarrow p(b)}(\forall_s)[x \leftarrow b]}{(\forall x)p(x) \Rightarrow p(a) \wedge p(b)}(\wedge_d)}{\Rightarrow (\forall x)p(x) \to p(a) \wedge p(b)}(\to_d)
$$

Formulele cuantificate universal s-au instanțiat pe cele două ramuri cu constantele a, respectiv b. Se observă că secvența inițială s-a redus la secvențe de bază, deci formula U este validă.

Exemplul 4.6.

Verificați dacă are loc: $\models (\forall x)p(x) \vee (\forall x)q(x) \to (\forall x)(p(x) \vee q(x))$.

Este importantă redenumirea variabilelor legate astfel încât acestea să fie distincte în secvența inițială: $\Rightarrow (\forall x)p(x) \vee (\forall y)q(y) \to (\forall z)(p(z) \vee q(z))$.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\overline{(\forall x)p(x), p(z) \Rightarrow p(z), q(z)}}{(\forall x)p(x) \Rightarrow p(z), q(z)}(\forall_s),[x \leftarrow z] \qquad \dfrac{\overline{(\forall y)q(y), q(z) \Rightarrow p(z), q(z)}}{(\forall y)q(y) \Rightarrow p(z), q(z)}(\forall_s),[y \leftarrow z]}{(\forall x)p(x) \vee (\forall y)q(y) \Rightarrow p(z), q(z)}(\vee_s)}{(\forall x)p(x) \vee (\forall y)q(y) \Rightarrow p(z) \vee q(z)}(\vee_d)}{(\forall x)p(x) \vee (\forall y)q(y) \Rightarrow (\forall z)(p(z) \vee q(z))}(\forall_d)}{\Rightarrow (\forall x)p(x) \vee (\forall y)q(y) \to (\forall z)(p(z) \vee q(z))}(\to_d)
$$

Aplicarea regulii de reducere $(\forall_d)$ realizează transformarea variabilei z din variabilă legată în variabilă liberă (este eliminat cuantificatorul universal). Rezultatul aplicării regulii $(\forall_s)$ este instanțierea formulei universale cu termeni adecvați (în acest exemplu variabilele x, y sunt substituite cu z) și duplicarea formulei universale pentru a putea fi ulterior instanțiată. Cele două noduri frunză ale arborelui de reducere conțin secvențe de bază, deci formula inițială este tautologie.

Exemplul 4.7.

Verificați dacă are loc proprietatea de distributivitate a cuantificatorului existențial față de conjuncție

Se verifică validitatea formulei $U = (\exists x)p(x) \wedge (\exists x)q(x) \leftrightarrow (\exists x)(p(x) \wedge q(x))$. $U \equiv U_1 \wedge U_2$, unde:

$U_1 = (\exists x)p(x) \wedge (\exists x)q(x) \to (\exists x)(p(x) \wedge q(x))$ și $U_2 = (\exists x)(p(x) \wedge q(x)) \to (\exists x)p(x) \wedge (\exists x)q(x)$.

Vom demonstra că formula U_1 nu este tautologie, dar formula U_2 este tautologie.

Pentru construirea secvențelor inițiale corespunzătoare formulelor U_1 și U_2 trebuie redenumite variabilele legate, astfel încât acestea să fie distincte.

Arborele de reducere a secvenței inițiale $\Rightarrow (\exists x)p(x) \wedge (\exists y)q(y) \rightarrow (\exists z)(p(z) \wedge q(z))$ corespunzătoare formulei U_1 este:

$$\dfrac{\overline{p(x),q(y) \Rightarrow p(x),A} \qquad \dfrac{\overline{p(x),q(y) \Rightarrow q(x),q(y),A} \quad p(x),q(y) \Rightarrow q(x),p(y),A}{p(x),q(y) \Rightarrow q(x),p(y)\wedge q(y),(\exists z)(p(z)\wedge q(z))}(\wedge_d)}{p(x),q(y) \Rightarrow p(x)\wedge q(x),(\exists z)(p(z)\wedge q(z))}(\wedge_d)$$

$$\dfrac{p(x),q(y) \Rightarrow q(x),p(y)\wedge q(y),(\exists z)(p(z)\wedge q(z))}{p(x),q(y) \Rightarrow q(x),(\exists z)(p(z)\wedge q(z))}(\exists_d)[z \leftarrow y]$$

$$\dfrac{p(x),q(y) \Rightarrow p(x)\wedge q(x),(\exists z)(p(z)\wedge q(z))}{p(x),q(y) \Rightarrow (\exists z)(p(z)\wedge q(z))}(\exists_d)[z \leftarrow x]$$

$$\dfrac{\dfrac{\dfrac{p(x),q(y) \Rightarrow (\exists z)(p(z)\wedge q(z))}{(\exists x)p(x),(\exists y)q(y) \Rightarrow (\exists z)(p(z)\wedge q(z))}(\exists_s)}{(\exists x)p(x)\wedge(\exists y)q(y) \Rightarrow (\exists z)(p(z)\wedge q(z))}(\wedge_s)}{\Rightarrow (\exists x)p(x)\wedge(\exists y)q(y) \rightarrow (\exists z)(p(z)\wedge q(z))}(\rightarrow_d)$$

S-a notat cu A formula $(\exists z)(p(z) \wedge q(z))$.

La prima aplicare a regulii $(\exists_d)$ s-a utilizat substituția $[z \leftarrow x]$, iar la cea de-a doua aplicare substituția $[z \leftarrow y]$.

Dacă se inversează termenii cu care se instanțiază variabila z, se obține un arbore de reducere diferit, dar cu aceleaşi noduri frunză ca şi cel desenat.

Nodul ce conține formula $p(x),q(y) \Rightarrow q(x), p(y), A$ este nod frunză, deoarece secvența respectivă nu mai poate fi simplificată (s-au făcut cele două instanțieri adecvate).

Arborele obținut are un nod frunză care nu conține o secvență de bază, deci formula U_1 nu este validă.

Arborele de reducere a secvenței inițiale $\Rightarrow (\exists z)(p(z) \wedge q(z)) \rightarrow (\exists x)p(x) \wedge (\exists y)q(y)$ corespunzătoare formulei U_2 este simbolizat grafic mai jos.

Secvența inițială $\Rightarrow U_2$ a fost redusă la secvențe de bază, deci formula U_2 este validă.

Concluzionăm că formula U nu este validă, deci nu are loc proprietatea de *distributivitate*, *ci doar* semidistributivitatea cuantificatorului existențial față de conjuncție: $\models (\exists x)(p(x) \wedge q(x)) \rightarrow (\exists x)p(x) \wedge (\exists x)q(x)$.

$$\dfrac{\dfrac{\overline{p(z),q(z) \Rightarrow p(z),(\exists x)p(x)}}{p(z),q(z) \Rightarrow (\exists x)p(x)}(\exists_d)[x \leftarrow z] \qquad \dfrac{\overline{p(z),q(z) \Rightarrow q(z),(\exists y)q(y)}}{p(z),q(z) \Rightarrow (\exists y)q(y)}(\exists_d)[y \leftarrow z]}{p(z),q(z) \Rightarrow (\exists x)p(x)\wedge(\exists y)q(y)}(\wedge_d)$$

$$\dfrac{\dfrac{\dfrac{p(z),q(z) \Rightarrow (\exists x)p(x)\wedge(\exists y)q(y)}{p(z)\wedge q(z) \Rightarrow (\exists x)p(x)\wedge(\exists y)q(y)}(\wedge_s)}{(\exists z)(p(z)\wedge q(z)) \Rightarrow (\exists x)p(x)\wedge(\exists y)q(y)}(\exists_s)}{\Rightarrow (\exists z)(p(z)\wedge q(z)) \rightarrow (\exists x)p(x)\wedge(\exists y)q(y)}(\rightarrow_d)$$

Exemplul 4.8.

Verificați validitatea formulei: $(\exists x)(\forall y)(p(x,y) \rightarrow r(x)) \rightarrow ((\forall x)(\forall y)p(x,y) \rightarrow (\exists s)r(s))$

Se reduce secvența $\Rightarrow (\exists x)(\forall y)(p(x,y) \rightarrow r(x)) \rightarrow ((\forall x)(\forall y)p(x,y) \rightarrow (\exists s)r(s))$ conform arborelui:

$$\dfrac{\overline{(x,x),B(x),C,D(x) \Rightarrow p(x,x),r(x),A} \qquad \overline{p(x,x),r(x),B(x),C,D(x) \Rightarrow r(x),A}}{p(x,x) \rightarrow r(x),p(x,x),B(x),C,D(x) \Rightarrow r(x),A}(\rightarrow_s)$$

$$\dfrac{p(x,x) \rightarrow r(x),p(x,x),B(x),C,D(x) \Rightarrow r(x),A}{p(x,x) \rightarrow r(x),B(x),(\forall t)p(x,t),C \Rightarrow r(x),A}(\forall_s)[t \leftarrow x]$$

$$\dfrac{p(x,x) \rightarrow r(x),B(x),(\forall t)p(x,t),C \Rightarrow r(x),A}{(\forall y)(p(x,y) \rightarrow r(x)),(\forall z)(\forall t)p(z,t) \Rightarrow r(x),A}(\forall_s)[y \leftarrow x, z \leftarrow x]$$

$$\dfrac{(\forall y)(p(x,y) \rightarrow r(x)),(\forall z)(\forall t)p(z,t) \Rightarrow r(x),A}{(\forall y)(p(x,y) \rightarrow r(x)),(\forall z)(\forall t)p(z,t) \Rightarrow (\exists s)r(s)}(\exists_d)[s \leftarrow x]$$

$$\dfrac{\dfrac{\dfrac{(\forall y)(p(x,y) \rightarrow r(x)),(\forall z)(\forall t)p(z,t) \Rightarrow (\exists s)r(s)}{(\exists x)(\forall y)(p(x,y) \rightarrow r(x)),(\forall z)(\forall t)p(z,t) \Rightarrow (\exists s)r(s)}(\exists_s)}{(\exists x)(\forall y)(p(x,y) \rightarrow r(x)) \Rightarrow (\forall z)(\forall t)p(z,t) \rightarrow (\exists s)r(s)}(\rightarrow_d)}{\Rightarrow (\exists x)(\forall y)(p(x,y) \rightarrow r(x)) \rightarrow ((\forall z)(\forall t)p(z,t) \rightarrow (\exists s)r(s))}(\rightarrow_d)$$

Am utilizat notațiile: $A = (\exists s)r(s)$, $B(x) = (\forall y)(p(x,y) \rightarrow r(x))$, $C = (\forall z)(\forall t)p(z,t)$, $D(x) = (\forall t)p(x,t)$ pentru copiile formulelor existențiale şi universale care au fost instanțiate.

Arborele de reducere corespunzător secvenței inițiale obținute după redenumirea variabilelor legate are două noduri frunză etichetate cu secvențe de bază, deci formula inițială este tautologie.

4.2. Metoda anti-secvențelor

Calculul anti-secvențelor [5] a fost introdus ca un sistem axiomatic complementar calculului secvențelor, utilizat pentru verificarea nevalidității nederivabilității în logica propozițiilor.

Definiția 4.3.

O *anti-secvență* are forma $X \not\Rightarrow Y$, unde X (*antecedent*) și Y (*consecvent*) sunt mulțimi finite de formule propoziționale. Metasimbolul $\not\Rightarrow$ are semnificația "*nu este demonstrabil*". O anti-secvență $X \not\Rightarrow Y$ este *anti-secvență de bază* dacă toate formulele din X și Y sunt literali și $X \cap Y = \varnothing$.

Definiția 4.4.

Anti-secvența $X \not\Rightarrow Y$ *este adevărată* dacă există un model M al mulțimii de formule X în care toate formulele din mulțimea Y sunt evaluate ca false. M se numește *anti-model* al anti-secvenței $X \not\Rightarrow Y$.

Toate anti-secvențele de bază sunt adevărate și vor reprezenta axiomele acestui sistem axiomatic.

Regulile calculului anti-secvențelor:

Introducerea în antecedent (reguli de stânga)	Introducerea în consecvent (reguli de dreapta)
$(\neg_s^c)\ \dfrac{X \not\Rightarrow A, Y}{X, \neg A \not\Rightarrow Y}$	$(\neg_d^c)\ \dfrac{X, A \not\Rightarrow Y}{X \not\Rightarrow \neg A, Y}$
$(\wedge_s^c)\ \dfrac{X, A, B \not\Rightarrow Y}{X, A \wedge B \not\Rightarrow Y}$	$(\wedge_{d1}^c)\ \dfrac{X \not\Rightarrow A, Y}{X \not\Rightarrow A \wedge B, Y}$ $\quad (\wedge_{d2}^c)\ \dfrac{X \not\Rightarrow B, Y}{X \not\Rightarrow A \wedge B, Y}$
$(\vee_{s1}^c)\ \dfrac{X, A \not\Rightarrow Y}{X, A \vee B \not\Rightarrow Y}$ $\quad (\vee_{s2}^c)\ \dfrac{X, B \not\Rightarrow Y}{X, A \vee B \not\Rightarrow Y}$	$(\vee_d^c)\ \dfrac{X \not\Rightarrow A, B, Y}{X \not\Rightarrow A \vee B, Y}$
$(\rightarrow_{s1}^c)\ \dfrac{X \not\Rightarrow A, Y}{X, A \rightarrow B \not\Rightarrow Y}$ $\quad (\rightarrow_{s2}^c)\ \dfrac{X, B \not\Rightarrow Y}{X, A \rightarrow B \not\Rightarrow Y}$	$(\rightarrow_d^c)\ \dfrac{X, A \not\Rightarrow B, Y}{X \not\Rightarrow A \rightarrow B, Y}$

Diferența dintre tabela de mai sus și cea care conține regulile calculului secvențelor constă în împărțirea regulilor cu două premise din calculul secvențelor, în perechi de reguli în calculul anti-secvențelor. Astfel.

căutarea exhaustivă din calculul secvențelor devine nedeterminism în calculul anti-secvențelor și procesul de reducere devine liniar.

Teorema 4.3. (corectitudinea și completitudinea calculului anti-secvențelor)

Anti-secvența $X \not\Rightarrow Y$ este adevărată dacă și numai dacă aceasta poate fi redusă la o anti-secvență de bază.

Teorema 4.4. (complementaritatea calcul secvențe – calcul anti-secvențe)

Anti-secvența $X \not\Rightarrow Y$ este adevărată dacă și numai dacă secvența $X \Rightarrow Y$ nu este adevărată.

Teorema 4.5.

- *O formulă U nu este teoremă (nu este validă) dacă și numai dacă anti-secvența $\not\Rightarrow U$ este adevărată.*
- *Nederivabilitatea este exprimată in calculul anti-secvențelor astfel: $U_1 \wedge U_2 \wedge \ldots \wedge U_n \nvdash V_1 \vee V_2 \vee \ldots \vee V_m$ dacă și numai dacă $U_1, U_2, \ldots, U_n \not\Rightarrow V_1, V_2, \ldots, V_m$ este adevărată, adică din conjuncția ipotezelor niciuna dintre formulele din consecvent nu poate fi demonstrată.*

Metoda anti-secvențelor:

Se reduce anti-secvența inițială prin aplicarea succesivă a regulilor de reducere, cu scopul obținerii unei anti-secvențe de bază. Există următoarele cazuri:

1. Dacă anti-secvența $\not\Rightarrow U$ nu este adevărată, deoarece nu a putut fi redusă la o anti-secvență de bază în toate modurile posibile de aplicare a regulilor de reducere, atunci formula U este validă.
2. Dacă anti-secvența $\not\Rightarrow U$ este adevărată, aceasta reducându-se la anti-secvența de bază $p_1, \ldots, p_k \not\Rightarrow q_1, \ldots, q_j$, atunci formula U nu este validă. Există o interpretare i, care evaluează formula U ca falsă, $i(U)$=F. Un anti-model parțial pentru U se obține astfel:
   $i(p_1) = \ldots = i(p_k)$=T și $i(q_1) = \ldots = i(q_j)$=F
3. Dacă anti-secvența $U_1, U_2, \ldots, U_n \not\Rightarrow V_1, V_2, \ldots, V_m$ este adevărată, aceasta reducându-se la anti-secvența de bază $p_1, \ldots, p_k \not\Rightarrow q_1, \ldots, q_j$, atunci niciuna dintre formulele $V_1, V_2, \ldots, V_m$ nu este derivabilă (deductibilă) din mulțimea ipotezelor: $\{U_1, U_2, \ldots, U_n\}$. Există o interpretare i, care este model pentru ipoteze, dar care falsifică toate formulele din consecvent, adică are loc:
   $i(U_1) = \ldots = i(U_n)$=T și $i(V_1) = \ldots = i(V_m)$=F.

MATEMATICA

Un anti-model parțial al deducției:

$U_1 \wedge U_2 \wedge \ldots \wedge U_n \vdash V_1 \vee V_2 \vee \ldots \vee V_m$ este obținut astfel:

$i(p_1) = \ldots = i(p_k) = T$ și $i(q_1) = \ldots = i(q_j) = F$

Exemplul 4.9.

Demonstrați că formula $U = \neg p \vee \neg q \to \neg(p \vee q)$ nu este validă.

Vom utiliza comparativ cele două sisteme complementare.

Calculul secvențelor:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\overline{p \Rightarrow p}}{p, \neg p \Rightarrow}(\neg_s) \quad \dfrac{p \Rightarrow q}{p, \neg q \Rightarrow}(\neg_s)}{p, \neg p \vee \neg q \Rightarrow}(\vee_s) \qquad \dfrac{\dfrac{\overline{q \Rightarrow q}}{q, \neg q \Rightarrow}(\neg_s) \quad \dfrac{q \Rightarrow p}{q, \neg p \Rightarrow}(\neg_s)}{q, \neg p \vee \neg q \Rightarrow}}{p \vee q, \neg p \vee \neg q \Rightarrow}(\vee_s)}{\neg p \vee \neg q \Rightarrow \neg(p \vee q)}(\neg_d)}{\Rightarrow \neg p \vee \neg q \to \neg(p \vee q)}(\to_d)
$$

Arborele de reducere a secvenței inițiale: $\Rightarrow \neg p \vee \neg q \to \neg(p \vee q)$ are două noduri frunză care conțin secvențe de bază și două noduri frunză etichetate cu secvențe ce nu mai pot fi reduse, dar nu sunt de bază. Astfel secvența inițială nu este adevărată, iar formula U nu este validă.

Calculul anti-secvențelor:

Anti-secvența inițială $\not\Rightarrow \neg p \vee \neg q \to \neg(p \vee q)$ este adevărată deoarece se reduce la anti-secvența de bază $q \not\Rightarrow p$, conform arborelui de reducere următor:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{q \not\Rightarrow p}{q, \neg p \not\Rightarrow}(\neg_s^c)}{q, \neg p \vee \neg q \not\Rightarrow}(\vee_{s1}^c)}{p \vee q, \neg p \vee \neg q \not\Rightarrow}(\vee_{s2}^c)}{\neg p \vee \neg q \not\Rightarrow \neg(p \vee q)}(\neg_d^c)}{\not\Rightarrow \neg p \vee \neg q \to \neg(p \vee q)}(\to_d^c)
$$

U nu este o formulă validă. Un anti-model al formulei U este furnizat de anti-secvența de bază: $q \not\Rightarrow p$ astfel:

$i: \{p, q\} \to \{T, F\}$, $i(q) = T$, $i(p) = F$.

Are loc $i(U) = i(\neg p) \vee i(\neg q) \to \neg\, i(p \vee q) = \neg F \vee \neg T \to \neg (F \vee T) = T \vee F \to \neg T = T \to F = F$, deci i este un anti-model al formulei U.

Se observă că utilizarea calculului anti-secvențelor a fost mai eficientă pentru a demonstra nevaliditatea formulei U, furnizând totodată și un anti-model al formulei.

Exemplul 4.10.

Utilizând calculul anti-secvențelor verificați dacă are loc relația de derivabilitate (consecință sintactică): $\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \vdash p$.

Arborele binar corespunzător reducerii anti-secvenței inițiale: $\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \not\Rightarrow p$ este:

$$
\dfrac{\dfrac{\dfrac{\dfrac{q, r \not\Rightarrow p}{q, r \vee \neg q \not\Rightarrow p}(\vee_{s1}^c)}{\neg p \vee q, r \vee \neg q \not\Rightarrow p}(\vee_{s2}^c)\ pt.\ \neg p \vee q}{\neg p \vee q, r \vee \neg q \not\Rightarrow p, p \wedge r}(\wedge_{d1}^c)}{\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \not\Rightarrow p}(\neg_s^c)
$$

Anti-secvența inițială s-a redus la anti-secvența de bază: $q, r \not\Rightarrow p$, care furnizează anti-modelul:

$i: \{p, q, r\} \to \{T, F\}$, $i(q) = T$, $i(r) = T$, $i(p) = F$,

ce evaluează ca adevărate formulele ipoteză și ca falsă concluzia.

Verificare: $i(\neg p \vee q) = T$, $i(r \vee \neg q) = T$, $i(\neg(p \wedge r)) = T$ și $i(p) = F$, deci nu are loc relația de derivabilitate din enunț.

O altă variantă a procesului nedeterminist, de reducere a aceleiași anti-secvențe inițiale, nu se încheie cu obținerea unei anti-secvențe de bază, conform arborelui de reducere următor:

$$
\dfrac{\dfrac{\dfrac{\dfrac{q, r \not\Rightarrow p, r}{q, r \vee \neg q \not\Rightarrow p, r}(\vee_{s1}^c)}{\neg p \vee q, r \vee \neg q \not\Rightarrow p, r}(\vee_{s2}^c)\ pt.\ \neg p \vee q}{\neg p \vee q, r \vee \neg q \not\Rightarrow p, p \wedge r}(\wedge_{d2}^c)}{\neg p \vee q, r \vee \neg q, \neg(p \wedge r) \not\Rightarrow p}(\neg_s^c)
$$

Exemplul 4.11.

Utilizând calculul anti-secvențelor verificați validitatea formulei:

$U = p \vee (q \wedge r) \to (p \vee q) \wedge (p \vee r)$.

Pentru verificarea validității unei formule trebuie verificate toate variantele de aplicare a regulilor de reducere pentru anti-secvența $\not\Rightarrow U$, cu scopul obținerii unei anti-secvențe de bază.

Există patru posibile procese liniare de reducere a anti-secvenței inițiale $\not\Rightarrow U$, prin reducerea formulei conjunctive din consecvent (utilizând $\wedge_{d1}^c$ sau $\wedge_{d2}^c$), respectiv a formulei disjunctive din antecedent (utilizând $\vee_{s1}^c$ sau $\vee_{s2}^c$), la una dintre cele două componente.

Varianta 1:

$$\cfrac{\cfrac{\cfrac{\cfrac{p \not\Rightarrow p,q}{p \not\Rightarrow p \vee q}(\vee^c_d)}{p \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\wedge^c_{d1})}{p \vee (q \wedge r) \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\vee^c_{s1})}{\not\Rightarrow p \vee (q \wedge r) \to (p \vee q) \wedge (p \vee r)}(\to^c_d)$$

Varianta 2:

$$\cfrac{\cfrac{\cfrac{\cfrac{p \not\Rightarrow p,r}{p \not\Rightarrow p \vee r}(\vee^c_d)}{p \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\wedge^c_{d2})}{p \vee (q \wedge r) \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\vee^c_{s1})}{\not\Rightarrow p \vee (q \wedge r) \to (p \vee q) \wedge (p \vee r)}(\to^c_d)$$

Varianta 3:

$$\cfrac{\cfrac{\cfrac{\cfrac{q,r \not\Rightarrow p,q}{q \wedge r \not\Rightarrow p \vee q}(\vee^c_d, \wedge^c_s)}{q \wedge r \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\wedge^c_{d1})}{p \vee (q \wedge r) \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\vee^c_{s2})}{\not\Rightarrow p \vee (q \wedge r) \to (p \vee q) \wedge (p \vee r)}(\to^c_d)$$

Varianta 4:

$$\cfrac{\cfrac{\cfrac{\cfrac{q,r \not\Rightarrow p,r}{q \wedge r \not\Rightarrow p \vee r}(\vee^c_d, \wedge^c_s)}{q \wedge r \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\wedge^c_{d2})}{p \vee (q \wedge r) \not\Rightarrow (p \vee q) \wedge (p \vee r)}(\vee^c_{s2})}{\not\Rightarrow p \vee (q \wedge r) \to (p \vee q) \wedge (p \vee r)}(\to^c_d)$$

Anti-secvența inițială nu este adevărată deoarece în niciuna dintre cele patru variante, $\not\Rightarrow U$ nu s-a redus la o anti-secvență de bază, deci formula U este validă.

Concluzii:

- Calculul secvențelor și calculul anti-secvențelor sunt două sisteme axiomatice complementare utilizate pentru a verifica validitatea/derivabilitatea, respectiv nevaliditatea/nederivabilitatea în calculul propozițional.
- Calculul anti-secvențelor permite totodată obținerea anti-modelelor unei formule sau deducții.

5. METODA REZOLUȚIEI ÎN LOGICILE CLASICE

În anul 1965 J.A. Robinson a propus *rezoluția*, o metodă de demonstrare puternică, eficientă și ușor de automatizat. Scopul primar al metodei rezoluției este să verifice consistența/inconsistența unei mulțimi de clauze. Rezoluția este utilizată ca metodă de demonstrare prin respingere pentru a rezolva cele două probleme decizionale ale logicilor clasice. Metoda rezoluției a fost adaptată și pentru logicile neclasice (logicile multivalente, logicile modale, logicile temporale și logicile nemonotone). S-au utilizat lucrările [1, 2, 8, 10, 12, 14, 15, 20, 28, 30, 32, 35, 41, 42, 43, 45] ca și referințe bibliografice.

5.1. Rezoluția în logica propozițiilor

Sistemul formal (axiomatic) asociat rezoluției propoziționale este:

$\text{Res} = (\Sigma_{\text{Res}}, F_{\text{Res}}, A_{\text{Res}}, R_{\text{Res}})$, unde:

- $\Sigma_{\text{Res}} = \Sigma_P - \{\to, \leftrightarrow, \wedge\}$ - ***alfabetul***
- $F_{\text{Res}} \cup \{\square\}$ - ***mulțimea formulelor bine-formate***
  - F_{Res} - mulțimea clauzelor ce se pot forma folosind alfabetul Σ_{Res}
  - $\square$ - ***clauza vidă*** este clauza fără literali și simbolizează inconsistența
- $A_{\text{Res}} = \varnothing$ - ***mulțimea axiomelor***
- $R_{\text{Res}} = \{res\}$ - ***mulțimea regulilor de inferență (deducție)*** care conține ***regula rezoluției***: $A \vee l, B \vee \neg l \vdash_{res} A \vee B$, unde l este un literal, iar $A, B \in F_{\text{Res}}$.

Dacă C1 $= A \vee l$ și C2$= B \vee \neg l$, spunem că acestea ***rezolvă*** deoarece conțin doi literali opuși (complementari).

Se folosește notația: C=Res$_l$ (C1,C2)$= A \vee B$, unde C se numește ***rezolventul*** clauzelor C1 și C2. C1 și C2 se numesc ***clauze părinți***. În cazul particular C1$=l$ și C2$=\neg l$, Res$_l$(C1,C2)=$\square$ care este inconsistentă.

Reprezentarea grafică a aplicării rezoluției ca regulă de inferență este:

$C1 = A \vee l$ $\quad$ $C2 = B \vee \neg l$ $\quad\longrightarrow\quad$ $C = Res_l(C1, C2) = A \vee B$

$C1 = l$ $\quad$ $C2 = \neg l$ $\quad\longrightarrow\quad$ $Res_l(C1, C2) = \square$

Rezoluția ca și regulă de inferență este o generalizare a regulilor *modus ponens, modus tollens* și a *silogismului.*

- Regula ***modus ponens***: $p, p \to q \vdash_{mp} q$ se poate scrie în forma echivalentă: $p, \neg p \vee q \vdash_{mp} q$, care este o particularizare a regulii rezoluției.
- Regula ***modus tollens*** (axioma A3 a sistemului axiomatic):
  $\vdash (p \to q) \to (\neg q \to \neg p)$ (aplicăm inversa teoremei de deducție) ==>
  $p \to q \vdash \neg q \to \neg p$ (aplicăm inversa teoremei de deducție) ==>
  $p \to q, \neg q \vdash \neg p$ ==> $\neg p \vee q, \neg q \vdash \neg p$ --- o particularizare a rezoluției.
- Regula ***silogismului***:
  $\vdash (p \to q) \to ((q \to r) \to (p \to r))$ (aplicăm inversa teoremei de deducție) ==>
  $p \to q \vdash (q \to r) \to (p \to r)$ (aplicăm inversa teoremei de deducție) ==>
  $p \to q, q \to r \vdash p \to r$ ==>
  $\neg p \vee q, \neg q \vee r \vdash \neg p \vee r$ ceea ce este o particularizare a rezoluției.

Metoda (procedura) rezoluției constă în aplicarea succesivă a regulii de rezoluție asupra unei mulțimi inițiale de clauze, îmbogățită în acest proces cu noi clauze (rezolvenți), în scopul derivării clauzei vide (inconsistenței). Dacă nu se poate deriva clauza vidă, mulțimea inițială de clauze este consistentă.

Algoritmul rezoluție-propozițională:

Date de intrare: S – o mulțime de clauze
Date de ieșire: mesaj: "*S – consistentă*" sau "*S – inconsistentă*"

```
    S0 = S
    i = 0
    execută
        Se aleg clauzele C1,C2 ∈ Si care rezolvă
        C=Res (C1,C2)
        Si+1 = Si ∪ {C}
        dacă(C = □)
            atunci scrie "S – inconsistentă"; STOP
            altfel i = i+1
        sf_dacă
    până_când ( Si = Si-1 ) // nu se mai pot deriva clauze noi
    scrie "S – consistentă"
Sf_algoritm
```


Notația: $S \vdash_{Res} \square$ are semnificația: "Din mulțimea S de clauze s-a derivat clauza vidă prin aplicarea algoritmului rezoluție-propozițională".

Teorema 5.1.[8] (corectitudinea rezoluției)
Dacă din mulțimea S de clauze este derivată clauza vidă prin aplicarea algoritmului rezoluție-propozițională, atunci S este o mulțime inconsistentă:
dacă $S \vdash_{Res} \square$ atunci S este mulțime inconsistentă.

Teorema 5.2.[8] (completitudinea rezoluției)
Dacă o mulțime S de clauze este inconsistentă, atunci clauza vidă poate fi derivată din S prin algoritmul rezoluție-propozițională:
dacă S este mulțime inconsistentă atunci $S \vdash_{Res} \square$.

Teorema 5.3. [8](corectitudinea și completitudinea rezoluției)
Mulțimea S de clauze este inconsistentă dacă și numai dacă $S \vdash_{Res} \square$.

Rezultatele teoretice furnizate de teoremele următoare sunt utilizate pentru rezolvarea prin respingere a celor două probleme decizionale ale calculului propozițional.

Teorema 5.4.
Formula propozițională U este o tautologie (teoremă) dacă și numai dacă prin procedura de rezoluție poate fi derivată clauza vidă din mulțimea de clauze corespunzătoare formei normale conjunctive a formulei $\neg U$:
U este tautologie (teoremă) dacă și numai dacă $FNC(\neg U) \vdash_{Res} \square$.

Teorema 5.5.
$U_1, U_2, ..., U_n \models V$ dacă și numai dacă $U_1, U_2, ..., U_n \vdash V$ dacă și numai dacă $FNC(U_1 \wedge U_2 \wedge ... \wedge U_n \wedge \neg V) \vdash_{Res} \square$.

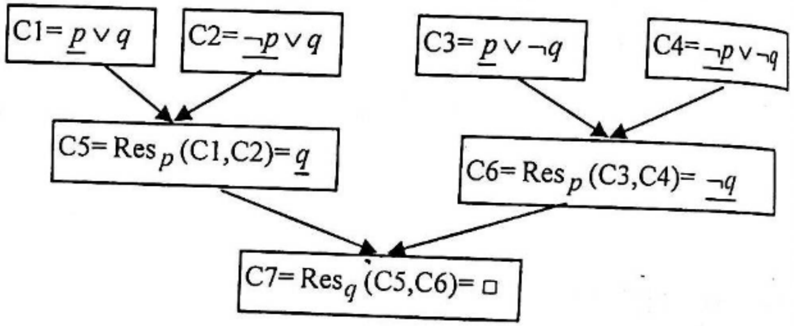
Exemplul 5.1.

Folosind metoda rezoluției să se demonstreze că mulțimea de clauze $S=\{ p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q \}$ este inconsistentă.

Denumim clauzele inițiale astfel:

$C1= p \vee q$, $C2=\neg p \vee q$, $C3= p \vee \neg q$ și $C4=\neg p \vee \neg q$.

Procesul de derivare a clauzei vide prin metoda rezoluției poate fi simbolizat grafic prin intermediul arborelui de mai jos.



Are loc $S \vdash_{Res} \square$ și conform teoremei de corectitudine a rezoluției rezultă că S este o mulțime inconsistentă.

Exemplul 5.2.

Folosind metoda rezoluției să se arate că are loc deducția:

$p \rightarrow q, p \rightarrow r \vdash p \rightarrow q \wedge r$.

Utilizăm Teorema 5.5. unde formulele ipoteză și concluzia sunt: $U_1 = p \rightarrow q$, $U_2 = p \rightarrow r$, respectiv $V= p \rightarrow q \wedge r$.

$FNC(U_1) = FNC(p \rightarrow q) = \neg p \vee q = C1$

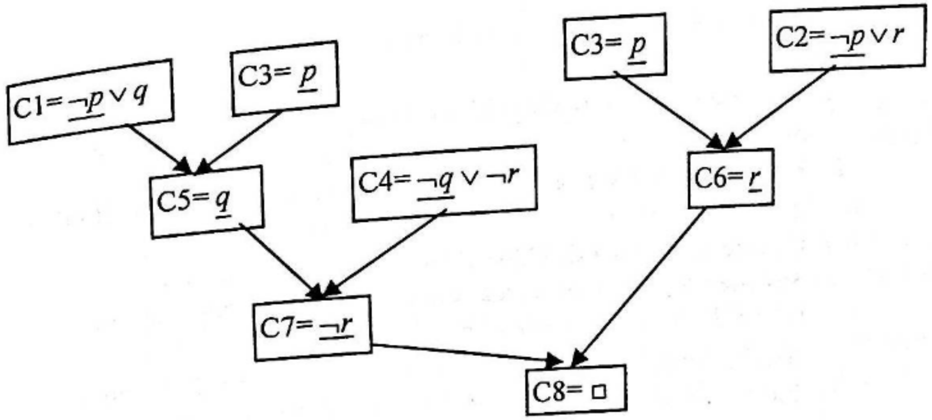
$FNC(U_2) = FNC(p \rightarrow r) = \neg p \vee r = C2$

$FNC(\neg V) = FNC(\neg(p \rightarrow q \wedge r)) = p \wedge (\neg q \vee \neg r)$

și clauzele $C3= p$, $C4=\neg q \vee \neg r$

Fie $S=\{C1,C2,C3,C4\}$ mulțimea de clauze obținută din formele normale conjunctive ale ipotezelor și negației concluziei deducției din enunțul problemei.

Arborele de derivare a clauzei vide din mulțimea S este următorul:



$FNC(U_1 \wedge U_2 \wedge \neg V) \vdash_{Res} \square$ și conform Teoremei 5.5. rezultă $U_1, U_2 \vdash V$, deci are loc deducția: $p \rightarrow q, p \rightarrow r \vdash p \rightarrow q \wedge r$

Teorema următoare, inspirată din procedura Davis-Putman, introduce un set de transformări pentru simplificarea mulțimii inițiale de clauze, cu păstrarea consistenței/inconsistenței acesteia.

Teorema 5.6.

O mulțime S de clauze poate fi simplificată, păstrând consistența/inconsistența acesteia, prin aplicarea următoarelor transformări:

1. **Eliminarea clauzelor tautologice.**
   Clauzele tautologice, deoarece sunt adevărate în toate interpretările, nu pot contribui la derivarea clauzei vide (inconsistența).
2. **Eliminarea clauzelor subsumate de alte clauze din S.**
   Clauza C1 *este subsumată de* C2 sau C2 *subsumează pe* C1, dacă există o clauză C3 astfel încât $C1=C2 \vee C3$ (toți literalii clauzei C2 sunt și literali ai clauzei C1).
   Această transformare are la bază proprietatea de absorbție: $C2 \wedge C1=C2 \wedge (C2 \vee C3) \equiv C2$.
3. **Ștergerea clauzelor care conțin un literal pur în S.**
   Literalul pur este acel literal care apare într-o clauză a mulțimii S, dar negația sa nu apare în celelalte clauze ale lui S.
4. **Fie $C=l$ o clauză unitate a lui S. Se șterg toate clauzele care conțin l și de asemenea se șterge literalul $\neg l$ din clauzele rămase.**
   Această transformare combină transformarea 2 (eliminarea clauzelor subsumate de $C=l$) și aplicarea regulii de rezoluție cu $C=l$ clauză părinte.

Exemplul 5.3.

1. $\neg p \vee \neg q \vee r \vee p$ este clauză tautologică deoarece $\neg p$, p sunt literali opuşi.
2. Fie $C1 = \neg q \vee r \vee p$ şi $C2 = r \vee p$ aparţinând aceleiaşi mulţimi S de clauze.
   C1 este subsumată de C2 şi C2 subsumează clauza C1.
   Se observă că prin aplicarea proprietăţii de absorbţie are loc:
   $C1 \wedge C2 = (\neg q \vee r \vee p) \wedge (r \vee p) \equiv r \vee p = C2$.
   Astfel, clauza C1 poate fi eliminată din mulţimea S de clauze
3. Fie $S = \{ C1 = \neg p \vee \neg q \vee r,\ C2 = r \vee p,\ C3 = \neg p \vee \neg r \}$.
   $\neg q$ este literal pur în S, iar C1 va fi eliminată din S.
   C1 nu poate contribui la derivarea $\square$ deoarece C1 nu va rezolva în raport cu literalul $\neg q$.
4. Fie $S = \{ C1 = \neg p \vee \neg q \vee r,\ C2 = r \vee p,\ C3 = \neg p \vee \neg r,\ C4 = q \}$
   C4 este clauză unitate, aplicăm transformarea 4 şi S devine S'
   $S' = \{ C1' = \neg p \vee r,\ C2 = r \vee p,\ C3 = \neg p \vee \neg r \}$.

Exemplul 5.4.

Fie $S = \{C1 = \neg q,\ C2 = q,\ C3 = p \vee q \vee r\}$

Dacă se aplică transformarea 4 cu C2 ca şi clauză unitate, C2 şi C3 sunt eliminate, iar C1 devine $C1' = \square$, după ştergerea literalului $\neg q$.

După simplificare S devine $S' = \{\square\}$ şi astfel mulţimea S este inconsistentă.

Exemplul 5.5.

Folosind transformările anterioare, să se simplifice mulţimea de clauze $S = \{C1 = \neg p \vee p \vee r,\ C2 = r \vee \neg r,\ C3 = p \vee q,\ C4 = p \vee q \vee r\}$.

- Clauza C4 este subsumată de clauza C3, deci este eliminată.
- C1 şi C2 sunt tautologii (conţin fiecare doi literali opuşi) deci sunt eliminate.
- Clauza C3 rămasă conţine doi literali puri: p şi q, deci este eliminată.

În urma aplicării transformărilor, mulţimea S simplificată devine $S' = \varnothing$.
Din S nu s-a putut deriva $\square$, deci S este o mulţime consistentă.

5.2. Strategiile şi rafinările rezoluţiei

Automatizarea procesului rezolutiv în cadrul metodelor de demonstrare presupune stabilirea unor *strategii* de aplicare a regulii de rezoluţie pentru a asigura explorarea tuturor modurilor posibile de derivare a clauzei vide

şi totodată evitarea deducerii unor clauze redundante sau irelevante pentru obţinerea clauzei vide.

1) **Strategia saturării pe nivele** asigură explorarea tuturor posibilităţilor de derivare a clauzei vide construind iterativ nivele de rezolvenţi.
   Nivelul iniţial (0) este format din mulţimea iniţială de clauze.
   Un nivel oarecare conţine toţi rezolvenţii obţinuţi dintr-o clauză părinte aparţinând nivelului precedent şi o clauză părinte aparţinând oricărui nivel anterior celui curent.
   Se continuă generarea nivelelor de rezolvenţi până la obţinerea clauzei vide sau până când nu se mai pot deriva noi rezolvenţi (ultimul nivel este vid).

Algoritmul strategia_saturării_pe_nivele

Date de intrare: S – mulţimea iniţială de clauze;
Date de ieşire: mesaj "*S – inconsistentă*" sau "*S – consistentă*"
// Se generează mulţimile de clauze: $S^0, S^1, \ldots, S^k$ ce reprezintă nivelele

$S^0 = S$; $k = 0$;
execută
  $k = k+1$
  $S^k = \{\mathrm{Res}(C1, C2) \mid C1 \in S^0 \cup S^1 \cup \ldots \cup S^{k-1}, C2 \in S^{k-1}\}$
  $S^k = S^k - (S^0 \cup S^1 \cup \ldots \cup S^{k-1})$ // se elimină rezolvenţii obţinuţi pe nivelul curent, care apar deja pe nivelele anterioare
până_când ($\square \in S^k$ sau $S^k = \varnothing$).
dacă ($\square \in S^k$)
  atunci scrie "*S – inconsistentă*"
  altfel scrie "*S – consistentă*" // nu se mai pot deriva rezolvenţi noi
sf_dacă
Sf_algoritm

2) **Strategia eliminării (ştergerii)**
   Rezolvenţii tautologii şi rezolvenţii subsumaţi de alte clauze sunt eliminaţi, adică nu vor fi utilizaţi mai departe în procesul deductiv deoarece produc clauze redundante.

Exemplul 5.6.

$\mathrm{Res}_p(p \vee q \vee \neg r,\ p \vee \neg p) = p \vee q \vee \neg r$

Se observă că rezolventul obţinut este clauza părinte, care nu este tautologie, deci prin utilizarea unei tautologii ca şi clauză părinte se obţine o clauză redundantă.

Exemplul 5.7.

Utilizând stategia saturării pe nivele combinată cu eliminarea rezolvenților tautologii de pe fiecare nivel, să se decidă consistența/inconsistența mulțimii de clauze:

$S = \{ p \vee q , \neg p \vee q \vee \neg r , \neg q \vee r \}$.

Construim secvența $S^0, S^1, S^2, \ldots$ care reprezintă nivelele de rezolvenți

$S^0 = S = \{C1 = p \vee q, C2 = \neg p \vee q \vee \neg r, C3 = \neg q \vee r\}$ – mulțimea inițială

Se generează primul nivel de rezolvenți:

C4= Res_p (C1,C2)= $q \vee \neg r$

C5= Res_q (C1,C3)= $p \vee r$

C6= Res_q (C2,C3)= $\neg p \vee \neg r \vee r$ --- tautologie

C7= Res_r (C2,C3)= $\neg p \vee q \vee \neg q$ --- tautologie

C6 și C7 nu vor fi introduse în primul nivel.

Primul nivel de rezolvenți este: $S^1 = \{C4 = q \vee \neg r, C5 = p \vee r\}$

Se generează cel de-al doilea nivel de rezolvenți:

C8= Res_r (C4,C5)= $q \vee p$ = C1

C9= Res_q (C4,C3)= $r \vee \neg r$ --- tautologie

C10= Res_r (C4,C3)= $q \vee \neg q$ --- tautologie

C11= Res_p (C5,C2)= $q \vee r \vee \neg r$ --- tautologie

C12= Res_r (C5,C2)= $\neg p \vee p \vee q$ --- tautologie

Clauzele C9, C10, C11 și C12 sunt tautologii, iar clauza C8 apare în mulțimea inițială ca și clauza C1, deci acestea nu vor fi introduse în cel de-al doilea nivel.

Astfel nivelul $S^2 = \varnothing$, semnificând faptul că nu mai pot fi generați noi rezolvenți.

Clauza vidă nu a putut fi derivată din mulțimea inițială de clauze, deci S este o mulțime consistentă.

3) Strategia mulțimii suport

Se evită aplicarea regulii de rezoluție asupra unor clauze dintr-o submulțime consistentă a mulțimii inițiale de clauze, deoarece rezolvenții obținuți sunt irelevanți în procesul de derivare a clauzei vide. Această strategie a fost inspirată din faptul următor: în general mulțimea premiselor (ipotezelor) unei deducții este consistentă, deci rezolvarea unor clauze provenite din această mulțime consistentă nu poate duce la derivarea clauzei vide (inconsistența).

Definiția 5.1.

Fie S o mulțime de clauze. O submulțime Y a lui S se numește ***mulțime suport*** a lui S, dacă S–Y este consistentă. ***Rezoluția mulțimii suport*** este rezoluția a două clauze care nu aparțin ambele mulțimii S–Y.

Exemplul 5.8.

Folosind strategia mulțimii suport să se arate că are loc deducția:

$p \to (q \to r), r \wedge s \to t, u \to s \wedge \neg t \vdash p \wedge q \to \neg u$.

Fie $U_1 = p \to (q \to r)$, $U_2 = r \wedge s \to t$, $U_3 = u \to s \wedge \neg t$, $V = p \wedge q \to \neg u$.

FNC(U_1)= $\neg p \vee \neg q \vee r$, C1= $\neg p \vee \neg q \vee r$

FNC(U_2)= $\neg r \vee \neg s \vee t$, C2= $\neg r \vee \neg s \vee t$

FNC(U_3)= $(\neg u \vee s) \wedge (\neg u \vee \neg t)$ compusă din clauzele C3= $\neg u \vee s$ și C4= $\neg u \vee \neg t$

FNC($\neg V$)= $p \wedge q \wedge u$, compusă din: C5= p, C6= q, C7= u

S = {C1,C2,C3,C4,C5,C6,C7}

Conform strategiei mulțimii suport alegem Y={C5,C6,C7} ca mulțime suport, corespunzătoare concluziei deducției.S–Y={C1,C2,C3,C4} este o mulțime consistentă de clauze corespunzătoare premiselor deducției.

Această strategie evită rezolvarea a două clauze din mulțimea consistentă S–Y. Deși clauzele C1 și C2 rezolvă în raport cu literalul r, evităm rezolvarea lor.

O derivare a clauzei vide din mulțimea S, conform strategiei mulțimii suport este:

C8= Res_p (C1,C5)= $\neg q \vee r$

C9= Res_q (C8,C6)= r

C10= Res_r (C9,C2)= $\neg s \vee t$

C11= Res_u (C7,C3)= s

C12= Res_u (C7,C4)= $\neg t$

C13= Res_s (C10,C11)= t.

C14= Res_t (C13,C12)= $\square$.

Astfel mulțimea S este inconsistentă și conform Teoremei 5.5. are loc deducția din enunț.

Rafinările rezoluției impun restricții asupra clauzelor care rezolvă, pentru a eficientiza procesul rezolutiv.

Printre rafinările rezoluției amintim: *rezoluția blocării*, *rezoluția liniară* (furnizează și o strategie de aplicare a rezoluției), *rezoluția semantică*. Strategiile rezolutive și rafinările rezoluției sunt complete și corecte. Problema care se pune este combinarea acestora.

Corectitudinea este păstrată și în cazul combinării unor strategii și rafinări ale rezoluției, dar completitudinea nu se realizează totdeauna.

Nerealizarea completitudinii: deși mulțimea inițială de clauze este inconsistentă, totuși nu se poate deriva clauza vidă, datorită prea multor restricții impuse de rafinările sau strategiile respective.

rezoluția generală + strategia eliminării --- completă și corectă
rezoluția generală + strategia mulțimii suport --- completă și corectă
rezoluția generală + strategia mulțimii suport + strategia eliminării --- completă și corectă

În continuare vom utiliza notația: $\vdash^{st}_{\mathrm{Res}}$ pentru a simboliza aplicarea procedurii de rezoluție propozițională combinată cu strategia/rafinarea *st*.

5.3. Rezoluția blocării (lock resolution)

- introdusă de Boyer în 1971, este foarte eficientă și ușor de implementat;
- fiecare apariție de literal din mulțimea de clauze este indexată arbitrar cu un întreg astfel încât indicii din mulțimea de clauze să fie distincți;
- ***restricția:*** literalii care rezolvă din clauzele părinți trebuie să aibă indicii cei mai mici din aceste clauze;
- literalii din rezolvenți moștenesc indicii de la clauzele părinți, iar în cazul moștenirii a doi literali identici, se păstrează cel cu indicele mai mic;
- la implementare trebuie combinată cu strategia saturării pe nivele pentru a asigura explorarea tuturor modurilor posibile de derivare a clauzei vide.

Teorema 5.7. [7] (completitudinea rezoluției blocării)
Fie S o mulțime de clauze în care fiecare literal este indexat în mod arbitrar cu un întreg. Dacă mulțimea S este inconsistentă, atunci există o deducție din mulțimea S a clauzei vide, prin rezoluția blocării: $S \vdash^{lock}_{\mathrm{Res}} \square$.

Teorema 5.8. [7] (corectitudinea rezoluției blocării)
Fie S o mulțime de clauze în care fiecare literal este indexat în mod arbitrar cu un întreg. Dacă $S \vdash^{lock}_{\mathrm{Res}} \square$ *atunci mulțimea S este inconsistentă.*

Observație:
Conform completitudinii rezoluției blocării, dacă dintr-o mulțime S de clauze indexate, folosind o indexare particulară, nu se poate deriva clauza vidă, atunci mulțimea S este consistentă.

Exemplul 5.9.

Rezoluția blocării + strategia eliminării este corectă dar nu completă.

Mulțimea $S = \{ p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q \}$ de clauze este inconsistentă conform Exemplului 5.1.

Alegem următoarea indexare a literalilor din cele patru clauze:

$C1 = {}_{(2)}p \vee {}_{(1)}q$, $C2 = {}_{(3)}\neg p \vee {}_{(4)}q$,

$C3 = {}_{(5)}p \vee {}_{(6)}\neg q$, $C4 = {}_{(8)}\neg p \vee {}_{(7)}\neg q$

Folosind rezoluția blocării se pot obține doar următorii rezolvenți care sunt tautologii:

$C5 = \mathrm{Res}_q(C1,C4) = {}_{(2)}p \vee {}_{(8)}\neg p$, $C6 = \mathrm{Res}_p(C2,C3) = {}_{(4)}q \vee {}_{(6)}\neg q$

Dacă aplicăm și strategia eliminării conform căreia rezolvenții tautologii sunt eliminați, adică nu vor contribui mai departe în procesul rezolutiv, atunci nu putem deriva $\square$, deși mulțimea S de clauze este inconsistentă. Astfel putem concluziona că rezoluția blocării + strategia eliminării nu este completă.

Continuând procesul rezolutiv folosind rezoluția blocării, dar păstrând rezolvenții tautologii, obținem mai departe:

$C7 = \mathrm{Res}_p(C2,C5) = {}_{(4)}q \vee {}_{(8)}\neg p$

Se observă că C7 și C2 sunt clauze similare, dar nu identice, ordinea literalilor (furnizată de indici) în cele două clauze este diferită. Din clauza C2 literalul $\neg p$ poate rezolva, iar din clauza C7 literalul q poate rezolva.

Astfel, rolul unei tautologii ca și clauză părinte poate fi de a modifica ordinea literalilor din cealaltă clauză părinte.

$C8 = \mathrm{Res}_q(C7,C4) = {}_{(8)}\neg p$, $C9 = \mathrm{Res}_p(C3,C8) = {}_{(6)}\neg q$

$C10 = \mathrm{Res}_q(C1,C9) = {}_{(2)}p$, $C11 = \mathrm{Res}_p(C10,C8) = \square$

Are loc $S \vdash^{lock}_{\mathrm{Res}} \square$, deci S este o mulțime inconsistentă.

Exemplul 5.10.

Rezoluția blocării + strategia mulțimii suport este corectă dar nu completă.

Folosind rezoluția blocării să se arate că are loc deducția:

$$p \to (q \to r), r \wedge s \to t, u \to s \wedge \neg t \vdash p \wedge q \to \neg u .$$

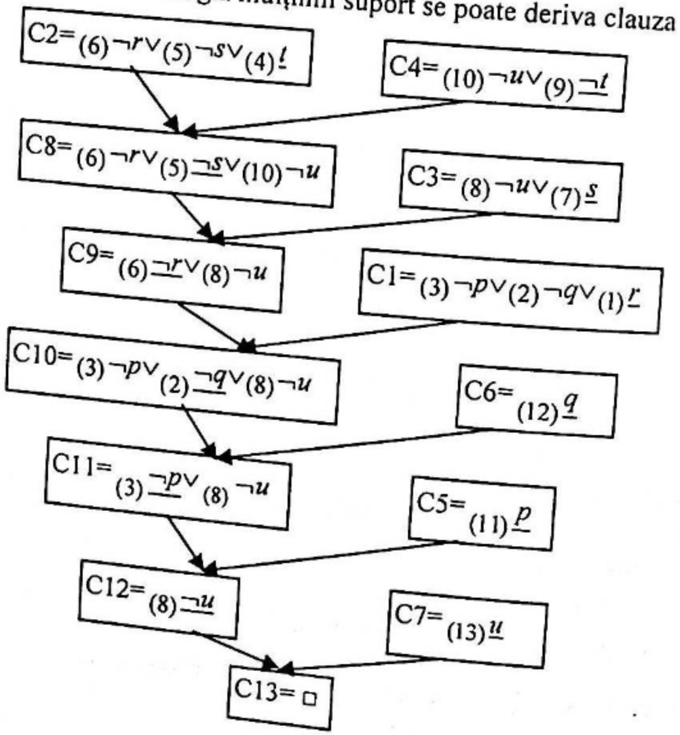
Conform Exemplului 5.8. se consideră mulțimea de clauze:
$S = \{C1, C2, C3, C4, C5, C6, C7\}$.
$Y = \{C5, C6, C7\}$ --- mulțimea suport a lui S.
$S - Y = \{C1, C2, C3, C4\}$ --- mulțime consistentă de clauze corespunzătoare premiselor.

Această strategie evită rezolvarea a două clauze din mulțimea consistentă $S-Y$. Indexăm literalii din clauze astfel:

$C1 = {}_{(3)}\neg p \vee {}_{(2)}\neg q \vee {}_{(1)} r$, $C2 = {}_{(6)}\neg r \vee {}_{(5)}\neg s \vee {}_{(4)} t$, $C3 = {}_{(8)}\neg u \vee {}_{(7)} s$.
$C4 = {}_{(10)}\neg u \vee {}_{(9)}\neg t$, $C5 = {}_{(11)} p$, $C6 = {}_{(12)} q$, $C7 = {}_{(13)} u$

Se observă că restricția impusă de rezoluția blocării (literalii cu indicii cei mai mici din fiecare clauză vor rezolva) + condiția cerută de strategia mulțimii suport, vor bloca procesul rezolutiv deoarece nu există nicio pereche de clauze care să rezolve, deci nu se poate deriva clauza vidă.

Eliminând strategia mulțimii suport se poate deriva clauza vidă astfel:

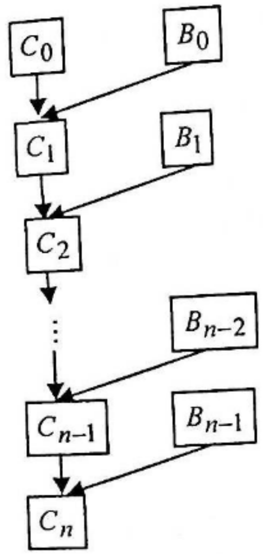


Aplicând rezoluția blocării, din mulțimea S s-a derivat clauza vidă: $S \vdash^{lock}_{Res} \square$, deci are loc deducția din enunț conform Teoremei 5.5.

5.4. Rezoluția liniară - Loveland 1970

Procesul rezolutiv este liniar: la fiecare pas una dintre clauzele părinte este rezolventul obținut la pasul anterior.

Arborele de derivare corespunzător procesului rezolutiv liniar are forma:



C_0 --- clauză vârf
$C_1, C_2, \dots, C_{n-1}, C_n$ --- clauze centrale
$B_0, B_1, \dots, B_{n-2}, B_{n-1}$ --- clauze laterale

$\forall i = \overline{1,n}$ are loc $C_i = \mathrm{Res}(C_{i-1}, B_{i-1})$

Teorema 5.9. (corectitudinea și completitudinea rezoluției liniare) *Mulțimea S de clauze este inconsistentă dacă și numai dacă* $S \vdash^{lin}_{Res} \square$.

Rezoluția liniară poate fi combinată cu strategia eliminării păstrându-se completitudinea.

Această rafinare a rezoluției furnizează și o strategie la nivel de implementare: *căutare cu revenire*. La fiecare iterație, pentru clauza centrală curentă pot exista mai multe posibile clauze laterale. După ce au fost utilizate toate posibilele clauze laterale, dar nu a fost derivată clauza vidă, se revine la iterația precedentă. Consistența mulțimii inițiale de clauze este demonstrată după o căutare completă fără derivarea clauzei vide.

Cazuri particulare ale rezoluției liniare:

- ***rezoluția unitate (unit)***: clauzele centrale au cel puțin o *clauză* părinte *unitate* (conține un singur literal)

- ***rezolutia de intrare (input)***: clauzele laterale sunt *clauze inițiale (de intrare)*

Teorema 5.10. (echivalența dintre rezoluția *unit* și rezoluția *input*)

Fie mulțimea S de clauze. $S \vdash^{input}_{Res} \square$ *dacă și numai dacă* $S \vdash^{unit}_{Res} \square$.

Aceste două rafinări ale rezoluției sunt corecte, eficiente, dar *nu sunt complete*:

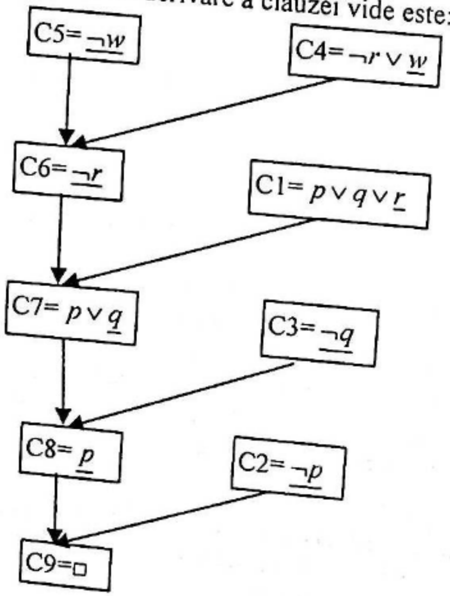
- **corectitudinea**: Dacă $S \vdash^{input/unit}_{Res} \square$ atunci S inconsistentă.
- **incompletitudinea:** există mulțimi inconsistente de clauze din care nu se poate deriva clauza vidă folosind rezoluția input sau rezoluția unit.

Exemplul 5.11.

Mulțimea de clauze $S=\{ p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q \}$ este inconsistentă conform Exemplului 5.1, dar nu există o derivare *unit* (unitate) a clauzei vide, deoarece S nu conține nicio clauză unitate. Datorită echivalenței dintre cele două subrafinări ale rezoluției liniare (Teorema 5.10) nu există nicio derivare *input* (de intrare) a clauzei vide din mulțimea S.

Exemplul 5.12.

Folosind rezoluția liniară să se demonstreze inconsistența mulțimii de clauze $S=\{C1,C2,C3,C4,C5\}$, unde:
C1= $p \vee q \vee r$, C2= $\neg p$, C3= $\neg q$, C4= $\neg r \vee w$ și C5= $\neg w$.
Arborele de derivare a clauzei vide este:



C5 --- clauză vârf

C4,C1,C3,C2 --- clauze laterale

C6,C7,C8,C9 --- clauze centrale

Are loc $S \vdash^{lin}_{Res} \square$, deci S este o mulțime inconsistentă. Această respingere liniară este și respingere *input* (clauzele laterale sunt clauze inițiale) și respingere *unitate* (clauzele centrale au cel puțin o clauză părinte unitate).

5.5. Rezoluția în logica predicatelor

Avantajele rezoluției predicative rezultă din scăderea computațională a formelor normale clauzale față de formulele inițiale.

Dezavantaje:

- prin procesul de Skolemizare se pierde semnificația formulelor inițiale și astfel nu se poate construi deducția, ci doar verifica inconsistența.
- este mai dificil de adaptat metoda rezoluției ca metodă de demonstrare pentru logicile neclasice.

Rezoluția predicativă ca sistem formal (axiomatic):

$Res^{Pr} = (\Sigma^{Pr}_{Res}, F^{Pr}_{Res}, A^{Pr}_{Res}, R^{Pr}_{Res})$. unde:

- $\Sigma^{Pr}_{Res} = \Sigma_{Pr} - \{\rightarrow, \leftrightarrow, \wedge, \exists\ \forall\}$ - ***alfabetul***
- $F^{Pr}_{Res} \cup \{\square\}$ - ***mulțimea formulelor bine formate***
  - F^{Pr}_{Res} - mulțimea clauzelor construite folosind alfabetul Σ^{Pr}_{Res}
  - $\square$ - clauza vidă
- $A^{Pr}_{Res} = \varnothing$ - ***mulțimea axiomelor***
- $R^{Pr}_{Res} = \{res^{Pr}, fact\}$ - ***mulțimea regulilor de inferență*** ce conține ***regula rezoluției*** (res^{Pr}) și ***regula de factorizare** (fact)*.
  - $A \vee l1, B \vee \neg l2 \vdash_{res}^{Pr} \theta(A) \vee \theta(B)$,unde $\theta = mgu(l1, l2)$ și $A, B \in F^{Pr}_{Res}$.
    C1 $= A \vee l1$ și C2= $B \vee \neg l2$. C1,C2 sunt clauze care *rezolvă* dacă literalii $l1$ și $l2$ sunt unificabili.
    Rezolventul binar este C= Res^{Pr}_{θ}(C1,C2)= $\theta(A) \vee \theta(B)$.
  - C $\vdash_{fact}$ C', C' se numește ***factor*** al lui C, dacă: $C = l_1 \vee l_2 \vee ... \vee l_k \vee l_{k+1} \vee ... \vee l_n$, $\lambda = mgu(l_1, l_2, ..., l_k)$ și
    $C' = \lambda(l_1) \vee \lambda(l_{k+1}) \vee ... \vee \lambda(l_n)$

Se pot combina cele două reguli de inferență și se obține definiția rezolventului.

Rezolventul clauzelor C1 și C2 poate fi:

1) rezolventul binar al clauzelor C1 și C2;
2) rezolventul binar al clauzei C1 și al unui factor al clauzei C2;
3) rezolventul binar al unui factor al clauzei C1 și al clauzei C2;
4) rezolventul binar al unui factor al clauzei C1 și al unui factor al clauzei C2.

Exemplul 5.13.

1. $C1 = \underline{p(f(x), g(y))} \vee q(x, y)$

   $C2 = \underline{\neg p(f(f(a)), g(z))} \vee q(f(a), g(z))$

   $\lambda = [x \leftarrow f(a), y \leftarrow z] = mgu(p(f(x), g(y)), p(f(f(a)), g(z)))$

   $Res_{\lambda}^{Pr}(C1, C2) = q(f(a), g(z)) \vee q(f(a), z)$

2. $C = p(x, g(y), z) \vee p(a, g(b), a) \vee q(x, z)$

   $B1 = p(x, g(y), z)$, $B2 = p(a, g(b), a)$

   $C' = \lambda(C)$, unde $\lambda = [x \leftarrow a, y \leftarrow b, z \leftarrow a] = mgu(B1, B2)$.

   $C' = p(a, g(b), a) \vee q(a, a)$ este factor al lui C.

Teorema 5.11.

Fie $U_1, U_2, \dots, U_n$ și V formule predicative.

- $\vdash V$ *dacă și numai dacă* $(\neg V)^c \vdash_{Res}^{Pr} \square$.
- $U_1, U_2, \dots, U_n \vdash V$ *dacă și numai dacă* $\{U_1^c, U_2^c, \dots, U_n^c, (\neg V)^c\} \vdash_{Res}^{Pr} \square$.

Observație:

Este recomandabilă redenumirea variabilelor libere aparținând clauzelor provenite din formele normale clauzale ale formulelor, astfel încât acestea să fie distincte.

Algoritmul rezoluție-predicativă:

Date de intrare: $U_1, U_2, \dots, U_n, V$ – formule predicative.

Date de ieșire: mesaj:"$U_1, U_2, \dots, U_n \vdash V$" sau "*nu are loc* $U_1, U_2, \dots, U_n \vdash V$"

Se construiește mulțimea de clauze: $S = \{U_1^c, U_2^c, \dots, U_n^c, (\neg V)^c\}$;

execută

Se selectează literalii l1,l2 și clauzele C1,C2 astfel încât:
C1, C2 sunt clauze sau factori ai unor clauze din *S*;

Fie $l1 \in C1$ și $\neg l2 \in C2$;
dacă (l1 și l2 sunt unificabili cu $\theta = mgu(l1, l2)$)
atunci
$C = Res_{\theta}^{Pr}(C1, C2)$;
dacă ($C = \square$) **atunci** scrie "are loc $U_1, U_2, \dots, U_n \vdash V$ ": STOP.
altfel $S = S \cup \{C\}$;
sf_dacă
sf_dacă
până_când (nu se mai pot deriva noi rezolvenți **sau** un număr fixat de iterații au fost executate)
dacă (nu se mai pot deriva noi rezolvenți)
atunci scrie "*nu are loc* $U_1, U_2, \dots, U_n \vdash V$ ".
altfel scrie "*nu se poate decide dacă are loc sau nu* $U_1, U_2, \dots, U_n \vdash V$ "
sf_dacă
Sf_algoritm

Se observă că algoritmul *rezoluție-predicativă* este semi-decizional, deoarece există situații în care nu se poate decide dacă are loc sau nu deducția (dacă mulțimea corespunzătoare de clauze este consistentă sau inconsistentă). După ce un număr fixat de iterații a fost executat și nu s-a derivat clauza vidă algoritmul se oprește fără o decizie.

În logica predicatelor se pot utiliza strategiile și rafinările rezoluției introduse în subcapitolele precedente pentru cazul propozițional.

Exemplul 5.14.

Să se verifice dacă are loc $A \vdash B$, unde:

$A = (\forall x)(p(x) \to q(x))$ și $B = (\forall x)p(x) \to (\forall x)q(x)$.

Aducem formulele A și $\neg B$ la forme normale clauzale:

- forme normale prenexe:

  $A^p = (\forall x)(\neg p(x) \vee q(x))$

  $(\neg B)^p = \neg((\forall x)p(x) \to (\forall y)q(y)) \equiv (\forall x)p(x) \wedge (\exists y)\neg q(y) \equiv$
  $\equiv (\forall x)(\exists y)(p(x) \wedge \neg q(y))$

  S-au extras în fața formulei întâi cuantificatorul universal și apoi cel existențial.

- forme normale Skolem (eliminarea cuantificatorilor existențiali):

  $A^S = (\forall x)(\neg p(x) \vee q(x))$,

  $(\neg B)^S = (\forall x)(p(x) \wedge \neg q(f(x)))$, *f* este funcție Skolem unară

- forme normale clauzale – forme Skolem fără cuantificatori (eliminarea cuantificatorilor universali) şi matricea în FNC

$A^{Sq} = \neg p(x) \vee q(x) = A^c$

$(\neg B)^{Sq} = p(x) \wedge \neg q(f(x)) = (\neg B)^c$

Se formează mulţimea de clauze $S = \{C1, C2, C3\}$

$C1 = \neg p(x) \vee q(x)$

$C2 = p(x)$

$C3 = \neg q(f(x))$

În clauzele mulţimii S s-a păstrat variabila liberă x provenită din formulele iniţiale.

Din mulţimea S nu se poate deriva □ deoarece atomii $\neg q(f(x))$ şi $q(x)$ nu sunt unificabili (x este subtermen al lui $f(x)$).

După redenumirea variabilei libere x cu y, respectiv cu z în clauzele C2 şi C3 ($C2 = p(y)$ şi $C3 = \neg q(f(z))$) se derivează clauza vidă astfel:

$C1, C2 \vdash_{res}^{[y \leftarrow x]} C4 = q(x)$

$C4, C3 \vdash_{res}^{[x \leftarrow f(z)]} \square$

Conform Teoremei 5.11. are loc $A \vdash B$.

În acest exemplu s-a observat necesitatea redenumirii variabilelor libere din clauze astfel încât acestea să fie distincte, pentru derivarea □.

Exemplul 5.15.

Demonstraţi că formula $U = (\forall y)(\exists x)\neg(p(x,y) \leftrightarrow \neg p(x,x))$ este validă utilizând rezoluţia.

$\neg U = \neg((\forall y)(\exists x)\neg(p(x,y) \leftrightarrow \neg p(x,x))) \equiv (\exists y)(\forall x)(p(x,y) \leftrightarrow \neg p(x,x))$

$\equiv (\exists y)(\forall x)((\neg p(x,y) \vee \neg p(x,x)) \wedge (p(x,x) \vee p(x,y)))$ --- formă normală prenexă

$(\neg U)^c \equiv (\neg p(x,a) \vee \neg p(x,x)) \wedge (p(x,x) \vee p(x,a))$, a -constantă Skolem

Forma clauzală obţinută furnizează clauzele:

$C1 = \neg p(x,a) \vee \neg p(x,x)$

$C2 = p(x,x) \vee p(x,a)$

Rezolvând cele două clauze nu se pot deriva decât tautologii. Trebuie aplicată regula de factorizare clauzelor C1, C2 şi apoi încerca rezolvarea factorilor obţinuţi.

$C1 \vdash_{fact}^{[x \leftarrow a]} C3 = \neg p(a,a)$, C3 este factor al clauzei C1.

$C2 \vdash_{fact}^{[x \leftarrow a]} C4 = p(a,a)$, C4 este factor al clauzei C2.

$C3, C4 \vdash_{res} \square$, deci U este o formulă validă conform Teoremei 5.11.

Exemplul 5.16.

Să se arate că mulţimea $S = \{C1, C2, C3, C4, C5, C6, C7\}$ de clauze este inconsistentă, folosind rezoluţia generală.

$C1 = \neg p(x) \vee q(x) \vee r(x, f(x))$

$C2 = \neg p(x) \vee q(x) \vee s(f(x))$

$C3 = t(a)$, $C4 = p(a)$

$C5 = \neg r(a,y) \vee t(y)$

$C6 = \neg t(x) \vee \neg q(x)$

$C7 = \neg t(x) \vee \neg s(x)$

Constante: a, simboluri de funcţii: f, simboluri de predicate: p, q, r, s, t.

În procesul rezolutiv se obţin următoarele clauze:

$\theta 1 = [x \leftarrow a]$;

$C8 = \mathrm{Res}_{\theta 1}^{Pr}(C3, C6) = \neg q(a)$;

$C9 = \mathrm{Res}_{\theta 1}^{Pr}(C2, C4) = q(a) \vee s(f(a))$

$C10 = \mathrm{Res}^{Pr}(C8, C9) = s(f(a))$

$C11 = \mathrm{Res}_{\theta 1}^{Pr}(C1, C4) = q(a) \vee r(a, f(a))$

$C12 = \mathrm{Res}^{Pr}(C8, C11) = r(a, f(a))$

$C13 = \mathrm{Res}_{\theta 2}^{Pr}(C5, C12) = t(f(a))$, $\theta 2 = [y \leftarrow f(a)]$

$C14 = \mathrm{Res}_{\theta 3}^{Pr}(C7, C13) = \neg s(f(a))$, $\theta 3 = [x \leftarrow f(a)]$

$C15 = \mathrm{Res}^{Pr}(C10, C14) = \square$.

Are loc $S \vdash_{Res}^{Pr} \square$, deci mulţimea S este inconsistentă conform teoremei de corectitudine a metodei rezoluţiei.

Exemplul 5.17.

Folosind rezoluția liniară să se arate că mulțimea de clauze predicative $S = \{C1, C2, C3, C4\}$ este inconsistentă, unde clauzele sunt:

$C1 = P(x, I(x), e)$

$C2 = \neg R(x) \vee \neg R(y) \vee \neg P(x, I(y), z) \vee R(z)$

$C3 = R(a)$, $C4 = \neg R(e)$

Constante: a, e, simbol de funcție: I, simboluri de predicate: P, R.
Pentru rezoluția liniară alegem *clauza vârf*: C4

$C2 \vdash_{fact}^{[y \leftarrow x]} C5 = \neg R(x) \vee \neg P(x, I(x), z) \vee R(z)$

C5 este factor al clauzei C2

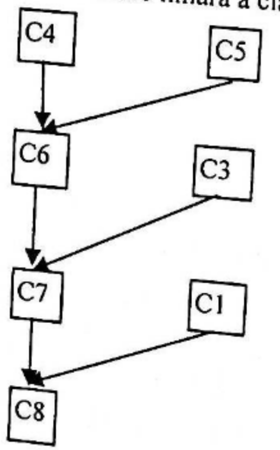
$C4, C5 \vdash_{res}^{[z \leftarrow e]} C6 = \neg R(x) \vee \neg P(x, I(x), e)$

$C6, C3 \vdash_{res}^{[x \leftarrow a]} C7 = \neg P(a, I(a), e)$

$C7, C1 \vdash_{res}^{[x \leftarrow a]} C8 = \square$

Din S s-a derivat clauza vidă folosind strategia liniară, deci S este o mulțime inconsistentă de clauze.

Arborele de derivare liniară a clauzei vide este:



C4 --- clauză vârf

C5, C3 și C1 --- clauze laterale

C6, C7 și C8 --- clauze centrale

Respingerea liniară obținută din S este totodată o respingere de intrare, deoarece toate clauzele laterale sunt clauze inițiale (sau factori ai clauzelor inițiale) și o respingere unitate, deoarece toate clauzele centrale au cel puțin o clauză părinte unitate.

Exemplul 5.18.

Să se verifice inconsistența mulțimii S folosind rezoluția blocării:

$S = \{ \neg p(x) \vee q(x) \vee r(x), \neg q(y) \vee r(y), p(a), \neg r(a) \}$

1) Se indexează literalii din clauze astfel:

$C1 = {}_{(3)}\neg p(x) \vee {}_{(2)}q(x) \vee {}_{(1)}r(x)$

$C2 = {}_{(5)}\neg q(y) \vee {}_{(4)}r(y)$

$C3 = {}_{(6)}\, p(a)$, $C4 = {}_{(7)}\neg r(a)$

Se obțin următorii rezolvenți:

$C5 = Res_{\theta 1}^{Pr}(C1, C4) = {}_{(3)}\neg p(a) \vee {}_{(2)}q(a)$, $\theta 1 = [x \leftarrow a]$

$C6 = Res_{\theta 2}^{Pr}(C2, C4) = {}_{(5)}\neg q(a)$, $\theta 2 = [y \leftarrow a]$

$C7 = Res^{Pr}(C5, C6) = {}_{(3)}\neg p(a)$

$C8 = Res^{Pr}(C3, C7) = \square$

S-a obținut $S \vdash_{Res}^{Pr, lock} \square$, deci conform teoremei de corectitudine și completitudine, mulțimea S de clauze predicative este inconsistentă.

2) Se indexează literalii din clauze astfel:

$C1 = {}_{(2)}\neg p(x) \vee {}_{(1)}q(x) \vee {}_{(3)}r(x)$

$C2 = {}_{(4)}\neg q(y) \vee {}_{(5)}r(y)$

$C3 = {}_{(6)}\, p(a)$, $C4 = {}_{(7)}\neg r(a)$

Se obțin următorii rezolvenți

$C5 = Res_{\lambda 1}^{Pr}(C1, C2) = {}_{(2)}\neg p(x) \vee {}_{(3)}r(x)$, $\lambda 1 = [y \leftarrow x]$

$C6 = Res_{\lambda 2}^{Pr}(C3, C5) = {}_{(3)}r(a)$, $\lambda 2 = [x \leftarrow a]$

$C7 = Res^{Pr}(C4, C6) = \square$

Și în această nouă indexare a literalilor din clauze, s-a obținut o respingere din S folosind rezoluția blocării.

Definiția 5.2.

1. O *clauză* se numește ***pozitivă*** dacă aceasta conține doar literali pozitivi.

2. O *clauză* se numeşte ***negativă*** dacă aceasta conţine doar literali negativi.
3. O clauză se numeşte ***clauză Horn*** dacă aceasta conţine un singur literal pozitiv, toţi ceilalţi fiind literali negativi.

Teorema 5.12.

Rezoluţia de intrare este completă pe o mulţime de clauze Horn, cu o clauză negativă ca şi clauză de vârf.(PROLOG)

Problemă specifică manipulării bazelor de cunoştinţe:

Din mulţimea ipotezelor:

H1: $U_1 \wedge U_2 \wedge ... \wedge U_n \rightarrow V$

H2: $X_1 \wedge X_2 \wedge ... \wedge X_l \rightarrow Y$

...

Hj: $W_1 \wedge W_2 \wedge ... \wedge W_m \rightarrow T$

este deductibilă concluzia C= $Z_1 \wedge Z_2 \wedge ... \wedge Z_m$?

Demonstrarea prin respingere folosind rezoluţia presupune transformarea în forme normale clauzale a ipotezelor şi a negaţiei concluziei.

O formulă de tipul: $U_1 \wedge U_2 \wedge ... \wedge U_n \rightarrow V$ furnizează clauza Horn: $\neg U_1 \vee \neg U_2 \vee ... \vee \neg U_n \vee V$

O formulă de tipul $\neg(Z_1 \wedge Z_2 \wedge ... \wedge Z_m)$ furnizează clauza negativă: $\neg Z_1 \vee \neg Z_2 \vee ... \vee \neg Z_m$

Astfel, pentru a rezolva problema enunţată, vom utiliza rezoluţia de intrare pe mulţimea clauzelor Horn provenite din ipoteze, clauza vârf fiind clauza negativă $\neg Z_1 \vee \neg Z_2 \vee ... \vee \neg Z_m$ furnizată de negaţia concluziei.

5.6. Exemple de modelare a raţionamentului

Exemplul 5.19.

Fie baza de cunoştinţe formată din ipotezele H1,H2,H3 şi H4 şi afirmaţiile D1 şi D2.

H1: Toate păsările colibri sunt bogat colorate.
H2: Păsările mari nu se hrănesc cu miere.
H3: Păsările care nu se hrănesc cu miere nu sunt bogat colorate.
H4: *Piky* este o pasăre colibri.
D1: *Piky* este o pasăre mică şi se hrăneşte cu miere.
D2: Toate păsările colibri sunt păsări mici.

Să se verifice dacă au loc deducţiile:

H1,H2,H3,H4 $\vdash$ D1 şi H1,H2,H3 $\vdash$ D2

Se transformă afirmaţiile din limbaj natural în formule predicative:

H1: $(\forall x)(colibri(x) \rightarrow bogat\_colorat(x))$ – regulă

H2: $(\forall x)(\neg pasare\_mica(x) \rightarrow \neg hrana\_miere(x))$ – regulă

H3: $(\forall x)(\neg hrana\_miere(x) \rightarrow \neg bogat\_colorat(x))$ – regulă

H4: $colibri(Piky)$

D1: $pasare\_mica(Piky) \wedge hrana\_miere(Piky)$ – concluzie

D2: $(\forall x)(colibri(x) \rightarrow pasare\_mica(x))$ – concluzie

Se transformă în forme normale clauzale formulele: H1, H2, H3, H4, $\neg$ D1, $\neg$ D2.

H1^c = $\neg colibri(x) \vee bogat\_colorat(x)$: **C1** – clauză Horn

H2^c = $pasare\_mica(x) \vee \neg hrana\_miere(x)$: **C2** – clauză Horn

H3^c = $hrana\_miere(x) \vee \neg bogat\_colorat(x)$: **C3** – clauză Horn

H4^c = $colibri(Piky)$: **C4** – clauză pozitivă, caz particular al clauzei Horn

$(\neg D1)^c = \neg pasare\_mica(Piky) \vee \neg hrana\_miere(Piky)$: **C5** clauză negativă

$\neg$ D2= $\neg((\forall x)(colibri(x) \rightarrow pasare\_mica(x))) \equiv$

$\equiv (\exists x)(colibri(x) \wedge \neg pasare\_mica(x))$

$(\neg D2)^c = colibri(a) \wedge \neg pasare\_mica(a)$ = **C6** $\wedge$ **C7**, *a* - constantă Skolem

Verificăm dacă are loc deducţia H1, H2, H3, H4 $\vdash$ D1.

Fie $S1$ = {C1,C2,C3,C4} mulţimea de clauze Horn provenite din ipoteze, iar C5 clauza negativă corespunzătoare negaţiei concluziei.

Respingerea de intrare (input) din $S1 \cup \{C5\}$ cu clauza vârf $C5$ este:

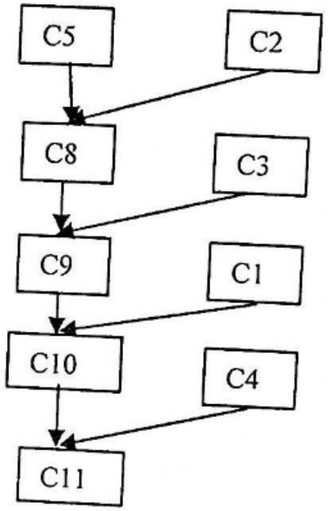
C8 = Res_{λ}^{Pr} (C5,C2) = $\neg hrana\_miere(Piky)$, $\lambda = [x \leftarrow Piky]$

C9 = Res_{λ}^{Pr} (C8,C3)= $\neg bogat\_colorat(Piky)$

C10 = Res_{λ}^{Pr} (C9,C1)= $\neg colibri(Piky)$

C11 = Res^{Pr} (C10,C4)= $\square$

Procesul rezolutiv este simbolizat grafic astfel:



C5 --- clauză vârf, clauză negativă

C2,C3,C1,C4 --- clauze laterale, clauze de intrare, clauze Horn

Are loc $S1 \cup \{C5\} \vdash^{Pr}_{Res} \square$, deci $S1 \cup \{C5\}$ este o mulţime inconsistentă de clauze (conform Teoremei 5.12.) şi astfel are loc deducţia: H1, H2, H3, H4 $\vdash$ D1.

Mulţimea clauzelor corespunzătoare formulei $H1 \wedge H2 \wedge H3 \wedge \neg D2$ este: $S2 = \{C1,C2,C3,C6,C7\}$.

Utilizăm strategia mulţimii suport, $S2$ având mulţimea suport Y={C6,C7}. Nu vor rezolva clauzele din mulţimea consistentă corespunzătoare ipotezelor: $S2 - Y = \{C1,C2,C3\}$

$\lambda = [x \leftarrow a]$

$C12 = Res^{Pr}_{\lambda}$ (C1,C6) = *bogat_colorat(a)*

$C13 = Res^{Pr}_{\lambda}$ (C12,C3) = *hrana_miere(a)*

$C14 = Res^{Pr}_{\lambda}$ (C13,C2) = *pasare_mica(a)*

$C15 = Res^{Pr}$ (C14,C7) = $\square$

Are loc $S2 \vdash^{Pr}_{Res} \square$, deci $S2$ este o mulţime inconsistentă şi conform Teoremei 5.11. are loc deducţia: H1,H2,H3 $\vdash$ D2

Exemplul 5.20.

Să se demonstreze teorema: "*Dacă fiecare element dintr-un grup este propriul său invers, atunci grupul este abelian*".

Axiomele care definesc grupul, ipoteza: "fiecare element este propriul său invers" şi concluzia: "grupul este abelian" sunt exprimate în continuare.

Limbaj matematic:

H1: $(\forall x)(\forall y)(\forall z)[(x * y) * z = x * (y * z)]$ – asociativitatea

H2: $(\exists e)(\forall x)[x * e = e * x = x]$ – existenţă element neutru: e

H3: $(\forall x)[x * x = e]$ – fiecare element este propriul său invers

C: $(\forall x)(\forall y)[x * y = y * x]$ – concluzia: "grupul este abelian"

Logica predicatelor:

Se utilizează simbolul de predicat ternar P, cu semnificaţia: P(x,y,z): "x*y=z" şi constanta e.

Axiomei H1 îi corespund formulele U1 şi U2, unde:

U1: $(\forall x)(\forall y)(\forall z)(\forall u)(\forall v)(\forall w)[P(x,y,u) \wedge P(u,z,w) \wedge P(y,z,v) \to P(x,v,w)]$

U2: $(\forall x)(\forall y)(\forall z)(\forall u)(\forall v)(\forall w)[P(y,z,v) \wedge P(x,v,w) \wedge P(x,y,u) \to P(u,z,w)]$

$U1^c = \neg P(x,y,u) \vee \neg P(u,z,w) \vee \neg P(y,z,v) \vee P(x,v,w)$: **C1** – forma cauzală

$U2^c = \neg P(y,z,v) \vee \neg P(x,v,w) \vee \neg P(x,y,u) \vee P(u,z,w)$: **C2** – forma clauzală

Axiomei H2 îi corespund formulele U3 şi U4, unde:

U3: $(\forall x)P(x,e,x)$, $U3^c = P(s,e,s)$: **C3** – forma clauzală

U4: $(\forall x)P(e,x,x)$, $U4^c = P(e,r,r)$: **C4** – forma clauzală

Ipotezei H3 îi corespunde formula U5:

U5: $(\forall x)P(x,x,e)$, $U5^c = P(l,l,e)$: **C5** – forma clauzală

Concluziei C îi corespunde formula U6:

U6: $(\forall x)(\forall y)(\exists t)[P(x,y,t) \to P(y,x,t)]$

$\neg$ U6: $\neg((\forall x)(\forall y)(\exists t)[P(x,y,t) \to P(y,x,t)]) \equiv$

$\equiv (\exists x)(\exists y)(\forall t)[P(x,y,t) \wedge \neg P(y,x,t)]$

$(\neg U6)^c = P(a,b,t) \wedge \neg P(b,a,t) =$ **C6** $\wedge$ **C7** , a,b – constante Skolem

Problema verificării dacă H1,H2,H3 $\vdash$ C s-a redus la verificarea inconsistenţei mulţimii de clauze $S = \{C1,C2,C3,C4,C5,C6,C7\}$.

Se observă că în clauzele C3, C4 şi C5 s-a redenumit variabila liberă x astfel încât să avem variabile distincte în mulţimea S de clauze.

Derivarea clauzei vide din mulţimea S se obţine astfel:

$C1 = \underline{\neg P(x,y,u)} \vee \neg P(u,z,w) \vee \neg P(y,z,v) \vee P(x,v,w)$, $C5 = \underline{P(l,l,e)}$

$\theta 1 = [x \leftarrow l, y \leftarrow l, u \leftarrow e] = \mathrm{mgu}(P(x,y,u), P(l,l,e))$

C8 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 1}(C1,C5) = \neg P(e,z,w) \vee \neg P(l,z,v) \vee P(l,v,w)$

$C2 = \neg P(y,z,v) \vee \neg P(x,v,w) \vee \underline{\neg P(x,y,u)} \vee P(u,z,w)$, $C6 = \underline{P(a,b,t)}$

$\theta 2 = [x \leftarrow a, y \leftarrow b, u \leftarrow t] = \mathrm{mgu}(P(a,b,t), P(x,y,u))$

C9 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 2}(C2,C6) = \neg P(b,z,v) \vee \neg P(a,v,w) \vee P(t,z,w)$

$C4 = \underline{P(e,r,r)}$, $C8 = \underline{\neg P(e,z,w)} \vee \neg P(l,z,v) \vee P(l,v,w)$

$\theta 3 = [z \leftarrow r, w \leftarrow r] = \mathrm{mgu}(P(e,z,w), P(e,r,r))$

C10 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 3}(C4,C8) = \neg P(l,r,v) \vee P(l,v,r)$

$C5 = \underline{P(l,l,e)}$, $C9 = \underline{\neg P(b,z,v)} \vee \neg P(a,v,w) \vee P(t,z,w)$

$\theta 4 = [l \leftarrow b, z \leftarrow b, v \leftarrow e] = \mathrm{mgu}(P(l,l,e), P(b,z,v))$

C11 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 4}(C5,C9) = \neg P(a,e,w) \vee P(t,b,w)$

$C3 = \underline{P(s,e,s)}$, $C11 = \underline{\neg P(a,e,w)} \vee P(t,b,w)$

$\theta 5 = [s \leftarrow a, w \leftarrow a] = \mathrm{mgu}(P(s,e,s), P(a,e,w))$

C12 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 5}(C3,C11) = P(t,b,a)$,

$C10 = \underline{\neg P(l,r,v)} \vee P(l,v,r)$, $C12 = \underline{P(t,b,a)}$

$\theta 6 = [l \leftarrow t, r \leftarrow b, v \leftarrow a] = \mathrm{mgu}(P(t,b,a), P(l,r,v))$

C13 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 6}(C10,C12) = P(t,a,b)$

$C2 = \underline{\neg P(y,z,v)} \vee \neg P(x,v,w) \vee \neg P(x,y,u) \vee P(u,z,w)$, $C5 = \underline{P(l,l,e)}$

$\theta 7 = [y \leftarrow l, z \leftarrow l, v \leftarrow e] = \mathrm{mgu}(P(l,l,e), P(y,z,v))$

C14 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 7}(C2,C5) = \underline{\neg P(x,e,w)} \vee \neg P(x,l,u) \vee P(u,l,w)$

$C3 = \underline{P(s,e,s)}$

$\theta 8 = [x \leftarrow s, w \leftarrow s] = \mathrm{mgu}(P(s,e,s), P(x,e,w))$

C15 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 8}(C3,C14) = \underline{\neg P(s,l,u)} \vee P(u,l,s)$

$C13 = \underline{P(t,a,b)}$

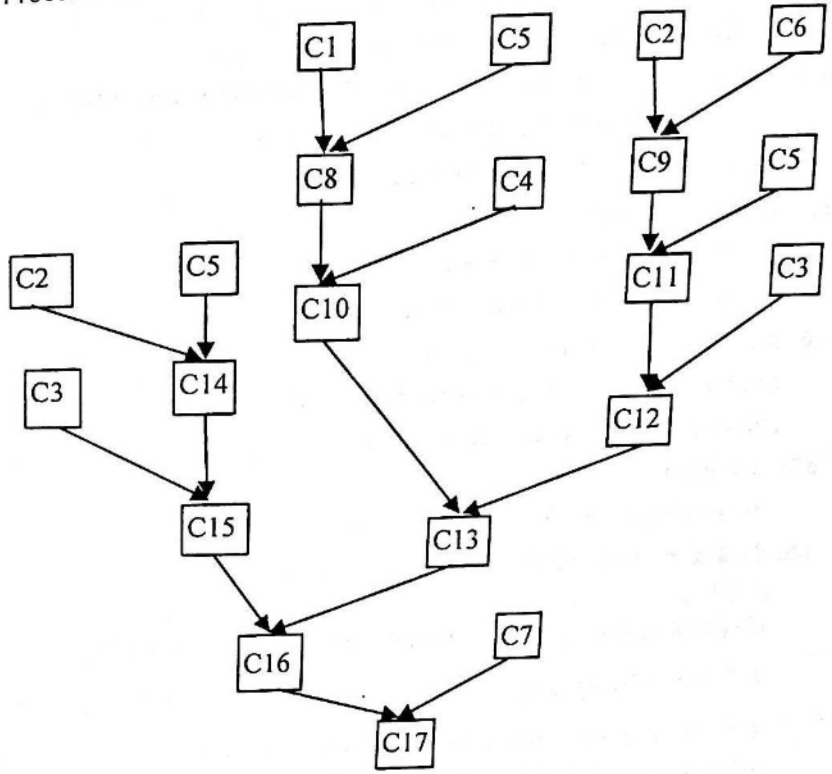
$\theta 9 = [s \leftarrow t, l \leftarrow a, u \leftarrow b] = \mathrm{mgu}(P(t,a,b), P(s,l,u))$

C16 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 9}(C13,C15) = \underline{P(b,a,t)}$,

$C7 = \underline{\neg P(b,a,t)}$

C17 $= \mathrm{Res}^{\mathrm{Pr}}(C16,C7) = \square$

Procesul de derivare a clauzei vide este simbolizat grafic prin arborele:



S-a derivat clauza vidă din S deci mulțimea S este inconsistentă. Are loc deducția H1, H2, H3 $\vdash$ C și astfel teorema "*Dacă fiecare element dintr-un grup este propriul său invers, atunci grupul este abelian*" este demonstrată.

Exemplul 5.21.
Să se demonstreze teorema: "*Există o infinitate de numere prime*".

Axiomele utilizate în demonstrarea teoremei din enunț sunt exprimate în limbaj matematic și în limbajul logicii predicatelor. Sunt obținute clauzele provenite din formele lor normale clauzale:

a1: $x \not< x$;
$(\forall x)\neg maimic(x,x)$, **C1** $= \neg maimic(x,x)$

a2: dacă $x < y$ atunci $y \not< x$

$(\forall x)(\forall y)(maimic(x,y) \rightarrow \neg maimic(y,x))$

C2 $= \neg maimic(x,y) \vee \neg maimic(y,x)$

a3: $x \mid x$ --- reflexivitatea divizibilității

$(\forall x) divide(x,x)$, **C3** $= divide(x,x)$

a4: dacă $x \mid y$ și $y \mid z$, atunci $x \mid z$ --- tranzitivitatea divizibilității

$(\forall x)(\forall y)(\forall z)(divide(x,y) \wedge divide(y,z) \rightarrow divide(x,z))$

C4 $= \neg divide(x,y) \vee \neg divide(y,z) \vee divide(x,z)$

a5: dacă $x < y$ atunci $y \nmid x$

$(\forall x)(\forall y)(maimic(x,y) \rightarrow \neg divide(y,x))$

C5 $= \neg maimic(x,y) \vee \neg divide(y,x)$

a6: dacă $y \mid F(x)$ atunci $x < y$, unde $F(x) = x!+1$

$(\forall x)(\forall y)(divide(y,F(x)) \rightarrow maimic(x,y))$

C6 $= \neg divide(y,F(x)) \vee maimic(x,y)$

a7: $x < F(x)$

$(\forall x) maimic(x,F(x))$; **C7** $= maimic(x,F(x))$

a8: dacă x nu este număr prim, atunci există y astfel încât $y \mid x$, y este prim și $y < x$.

$U = (\forall x)(\neg prim(x) \rightarrow (\exists y)(divide(y,x) \wedge prim(y) \wedge maimic(y,x)))$

$U^P = (\forall x)(\exists y)(prim(x) \vee (divide(y,x) \wedge prim(y) \wedge maimic(y,x)))$

$U^{Sq} = prim(x) \vee (divide(H(x),x) \wedge prim(H(x)) \wedge maimic(H(x),x))$, unde $H(x)$ este o funcție Skolem introdusă în procesul de Skolemizare.

Aplicăm distributivitatea pentru a obține forma normală clauzală.

$U^c = (prim(x) \vee divide(H(x),x)) \wedge (prim(x) \vee prim(H(x))) \wedge$
$\wedge (prim(x) \vee maimic(H(x),x))$

Clauzele obținute din U^c sunt:

C8 $= prim(x) \vee divide(H(x),x)$

C9 $= prim(x) \vee prim(H(x))$

C10 $= prim(x) \vee maimic(H(x),x)$

Concluzia:

"Dacă x este un număr prim atunci există y astfel încât y este prim. $x < y$ și $F(x) \nless y$"

este exprimată prin formula predicativă C.

$C = (\forall x)(prim(x) \rightarrow (\exists y)(prim(y) \wedge maimic(x,y) \wedge \neg maimic(F(x),y)))$

$C = (\forall x)(prim(x) \rightarrow (\exists y)(prim(y) \wedge maimic(x,y) \wedge \neg maimic(F(x),y))) \equiv$

$\neg C = \neg(\forall x)(prim(x) \rightarrow (\exists y)(prim(y) \wedge maimic(x,y) \wedge maimic(F(x),y))) \equiv$

$\equiv (\exists x)(prim(x) \wedge (\forall y)(\neg prim(y) \vee \neg maimic(x,y) \vee maimic(F(x),y)))$

$\equiv (\exists x)(prim(x) \wedge (\neg prim(y) \vee \neg maimic(x,y) \vee maimic(F(x),y))) : (\neg C)^P$

$\equiv (\exists x)(\forall y)(prim(x) \wedge (\neg prim(y) \vee \neg maimic(x,y) \vee maimic(F(x),y)))$ formă normală prenexă

Forma normală clauzală a formulei $\neg C$ este:

$prim(a) \wedge (\neg prim(y) \vee \neg maimic(a,y) \vee maimic(F(a),y))$,

formată din clauzele C11 și C12, unde a - constantă Skolem.

C11 $= prim(a)$

C12 $= \neg prim(y) \vee \neg maimic(a,y) \vee maimic(F(a),y)$

Utilizând rezoluția demonstrăm inconsistența mulțimii de clauze: $S = \{C1, C2, \ldots, C12\}$.

C9$= \underline{prim(x)} \vee prim(H(x))$,

C12$= \underline{\neg prim(y)} \vee \neg maimic(a,y) \vee maimic(F(a),y))$

$\theta 1 = [y \leftarrow x]$

C13 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 1}$ (C9,C12)$= prim(H(x)) \vee \neg maimic(a,x) \vee maimic(F(a),x)$

C8$= \underline{prim(x)} \vee divide(H(x),x)$

C12$= \underline{\neg prim(y)} \vee \neg maimic(a,y) \vee maimic(F(a),y)$

C14 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 1}$ (C8,C12)$= divide(H(x),x) \vee \neg maimic(a,x) \vee maimic(F(a),x)$

C13$= \underline{prim(H(x))} \vee \neg maimic(a,x) \vee maimic(F(a),x)$

C12$= \underline{\neg prim(y)} \vee \neg maimic(a,y) \vee maimic(F(a),y)$

$\theta 2 = [y \leftarrow H(x)]$

C15 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 2}$ (C13,C12) $=$

$= \neg maimic(a,x) \vee \underline{\neg maimic(a,H(x))} \vee maimic(F(a),x) \vee maimic(F(a),H(x))$

C6$= \neg divide(u,F(v)) \vee \underline{maimic(v,u)}$

$\theta 3 = [v \leftarrow a, u \leftarrow H(x)]$

C16 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 3}$ (C6,C15) $=$

$\neg maimic(a,x) \vee maimic(F(a),x) \vee \underline{maimic(F(a),H(x))} \vee \neg divide(H(x),F(a))$

C5$= \underline{\neg maimic(u,v)} \vee \neg divide(v,u)$

$\theta 4 = [u \leftarrow F(a), v \leftarrow H(x)]$

C17 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 4}$ (C5,C16) =

$= \neg maimic(a,x) \vee maimic(F(a),x) \vee \underline{\neg divide(H(x),F(a))}$

C14= $\underline{divide(H(x),x)} \vee \neg maimic(a,x) \vee maimic(F(a),x)$

$\theta 5 = [x \leftarrow F(a)]$

C18 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 5}$ (C17,C14)= $\underline{\neg maimic(a, F(a))} \vee maimic(F(a), F(a))$

C7= $\underline{maimic(x, F(x))}$

$\theta 6 = [x \leftarrow a]$

C19 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 6}$ (C7,C18)= $maimic(F(a), F(a))$.

C1= $\neg maimic(x,x)$

C20 $= \mathrm{Res}^{\mathrm{Pr}}_{\theta 5}$ (C1,C19)=$\square$,

$S \vdash^{\mathrm{Pr}}_{\mathrm{Res}} \square$, deci S este o mulțime inconsistentă, are loc deducția:

a1, a2, a3, a4, a5, a6, a7, a8 $\vdash C$ și astfel este demonstrată teorema din enunț.

6. ALGEBRE BOOLEENE. FUNCȚII BOOLEENE

Algebrele booleene, introduse de George Boole (1815-1864), stau la baza definirii funcțiilor booleene, care sunt utilizate în realizarea circuitelor logice. Acest capitol a avut ca surse bibliografice următoarele lucrări [6, 9, 17, 18, 36, 44].

Deoarece în literatura de specialitate se folosesc pentru cele două operații binare ale algebrelor booleene diferite simboluri: $\wedge$, $\cap$ sau $*$, respectiv $\vee$, $\cup$ sau $+$, precizăm că, în acest capitol se vor folosi seturile de simboluri: $\{\wedge,\vee\}$ și $\{*,+\}$.

6.1. Algebre booleene

Definiția 6.1. (definirea axiomatică)

O ***algebră booleană*** este o structură $(A, \wedge, \vee, \bar{\ }, 0, 1)$, unde:

$|A| \geq 2$, A conținând cel puțin 2 elemente diferite, 0 și 1, $0 \neq 1$

- $\wedge$, $\vee$ sunt operații binare
- $\bar{\ }$ este operator unar
- există elementul unic 0 – elementul zero, cu proprietățile:
  $x \wedge 0 = 0 \wedge x = 0$ și $x \vee 0 = 0 \vee x = x$, $\forall x \in A$
- există elementul unic 1 – elementul unitate, cu proprietățile:
  $x \wedge 1 = 1 \wedge x = x$ și $x \vee 1 = 1 \vee x = 1$, $\forall x \in A$
- elementul zero, 0, și cel unitate, 1, sunt primul, respectiv ultimul element, iar $\bar{x}$ este complementul lui x, cu proprietățile:
  $x \wedge \bar{x} = 0$ și $x \vee \bar{x} = 1$, $\forall x \in A$
- dubla negație:
  $\bar{\bar{x}} = x$, $\forall x \in A$
- operațiile $\wedge$ și $\vee$ sunt comutative:
  $x \wedge y = y \wedge x$ și $x \vee y = y \vee x$, $\forall x, y \in A$
- operațiile $\wedge$ și $\vee$ sunt asociative:
  $x \wedge (y \wedge z) = (x \wedge y) \wedge z$ $(= x \wedge y \wedge z)$ și
  $x \vee (y \vee z) = (x \vee y) \vee z$ $(= x \vee y \vee z)$, $\forall x, y, z \in A$
- au loc proprietățile de distributivitate a operatorilor $\wedge$ și $\vee$:

$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ şi

$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$, $\forall x, y, z \in A$

- are loc proprietatea de idempotenţă pentru operatorii $\wedge$ şi $\vee$:
  $x \wedge x = x$ şi $x \vee x = x$, $\forall x \in A$
- au loc legile lui De Morgan:
  $\overline{x \wedge y} = \bar{x} \vee \bar{y}$ şi $\overline{x \vee y} = \bar{x} \wedge \bar{y}$, $\forall x, y \in A$
- au loc proprietăţile de absorbţie:
  $x \wedge (x \vee y) = x$ şi $x \vee (x \wedge y) = x$, $\forall x, y \in A$

Într-o algebră booleană are loc principiul dualităţii: „Pentru orice egalitate între două expresii booleene $U=V$, există o nouă egalitate $U'=V'$, obţinută prin interschimbarea operaţiilor $\wedge$, $\vee$ şi a elementelor: 0, 1".

Afirmaţiile de mai sus reprezintă axiomele algebrei booleene, dintre care majoritatea sunt perechi de axiome duale.

Exemplul 6.1.

Algebra booleană binară este: $B = (B_2=\{0,1\}, *, +, \bar{}, 0, 1)$, unde tabelele operaţiilor $+$, $*$, $\bar{}$ sunt:

+	0	1
0	0	1
1	1	1

*	0	1
0	0	0
1	0	1

x	$\bar{x}$
0	1
1	0

- $|B_2| = 2$
- $+$, $*$ sunt operaţii binare, iar $\bar{}$ este operaţie unară;
- 0 este elementul zero – cel mai mic element şi 1 este elementul unitate – cel mai mare element; conform tabelelor operaţiilor, din prima linie, respectiv prima coloană:

$\forall x, y, z \in B_2$ au loc proprietăţile:

- 0 şi 1 sunt primul, respectiv ultimul element, iar $\bar{x}$ este complementul lui x:
  $x * \bar{x} = 0$ şi $x + \bar{x} = 1$
- $\bar{\bar{x}} = x$ (dubla negaţie)
- operaţiile $+$, $*$ sunt comutative deoarece primele două tabele sunt simetrice faţă de diagonala principală;
- operaţiile $+$, $*$ sunt asociative:
  $x + (y + z) = (x + y) + z \; (= x + y + z)$ şi
  $x * (y * z) = (x * y) * z \; (= x * y * z)$

- au loc proprietăţile de distributivitate a operatorilor $+$ şi $*$:
  $x + (y * z) = (x + y) * (x + z)$ şi
  $x * (y + z) = (x * y) + (x * z)$
- are loc proprietatea de idempotenţă pentru $+$ şi $*$:
  $x * x = x$ şi $x + x = x$
- au loc legile lui De Morgan:
  $\overline{x * y} = \bar{x} + \bar{y}$ şi $\overline{x + y} = \bar{x} * \bar{y}$
- are loc proprietatea de absorbţie pentru $+$ şi $*$:
  $x * (x + y) = x$ şi $x + (x * y) = x$

Exemplul 6.2.

Structura $(F_P, \wedge, \vee, \neg, \mathrm{F}, \mathrm{T})$ este o algebră booleană:

- $|F_P| \geq 2$, F_P – mulţimea formulelor propoziţionale corect construite.
- F, T $\in F_P$, unde F, T sunt valorile de adevăr: *false* şi *true*;
- $\wedge$ (conectiva conjuncţie), $\vee$ (conectiva disjuncţie) sunt operaţii binare;
- $\neg$ (conectiva negaţie) este operaţie unară;

Oricare ar fi formulele propoziţionale U, V, Z au loc.

- F – valoarea de adevăr "fals" este cel mai mic element, cu proprietăţile:
  $U \wedge \mathrm{F} = \mathrm{F} \wedge U = \mathrm{F}$ şi $U \vee \mathrm{F} = \mathrm{F} \vee U = U$
- T – valoarea de adevăr "adevărat, true" este cel mai mare element, cu proprietăţile:
  $U \wedge \mathrm{T} = \mathrm{T} \wedge U = U$ şi $U \vee \mathrm{T} = \mathrm{T} \vee U = \mathrm{T}$
- $\neg U$ este complementul lui U:
  $U \wedge \neg U = \mathrm{F}$ şi $U \vee \neg U = \mathrm{T}$
- $\neg\neg U = U$
- operaţiile $\wedge$, $\vee$ sunt comutative:
  $U \wedge V = V \wedge U$ şi $U \vee V = V \vee U$
- operaţiile $\wedge$, $\vee$ sunt asociative:
  $U \wedge (V \wedge Z) = (U \wedge V) \wedge Z \; (= U \wedge V \wedge Z)$ şi
  $U \vee (V \vee Z) = (U \vee V) \vee Z \; (= U \vee V \vee Z)$
- au loc proprietăţile de distributivitate a operatorilor $\wedge$ şi $\vee$:
  $U \wedge (V \vee Z) = (U \wedge V) \vee (U \wedge Z)$ şi
  $U \vee (V \wedge Z) = (U \vee V) \wedge (U \vee Z)$
- are loc idempotenţa pentru conectivele $\wedge$ şi $\vee$:
  $U \wedge U = U$ şi $U \vee U = U$

- au loc legile lui De Morgan:
  $\neg(U \vee V) = \neg U \wedge \neg V$ şi $\neg(U \wedge V) = \neg U \vee \neg V$
- are loc absorbţia:
  $U \wedge (V \vee U) = U$ şi $U \vee (U \wedge V) = U$

Exemplul 6.3.

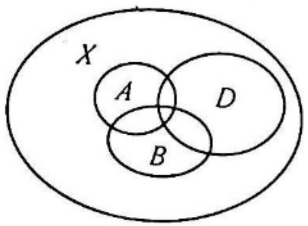
Un alt exemplu de algebră booleană este cel al mulţimii părţilor unei mulţimi date $X \neq \varnothing$, împreună cu operaţiile de intersecţie, reuniune şi respectiv complementara în raport cu X, $C_X(A) = X - A$, notată în mod simplificat doar cu $C(A)$:

Verificăm axiomele algebrei booleene pentru *structura*:

$(\mathcal{P}(X), \cap, \cup, C, \varnothing, X)$:

- $|\mathcal{P}(X)| \geq 2$ (cel puţin $\varnothing$ şi mulţimea totală X aparţin mulţimii $\mathcal{P}(X)$)
- $\cap$, $\cup$ sunt operaţii binare, C este un operator unar,

$\forall A, B, D \in \mathcal{P}(X)$, adică $A, B, D \subseteq X$, au loc proprietăţile de mai jos, care se observă uşor din figură:



- $\varnothing$ este cel mai mic element:
  $A \cap \varnothing = \varnothing \cap A = \varnothing$ şi $A \cup \varnothing = \varnothing \cup A = A$
- X este cel mai mare element:
  $A \cap X = X \cap A = A$ şi $A \cup X = X \cup A = X$
- $C(A)$ este complementara lui A faţă de mulţimea X.
  $A \cap C(A) = \varnothing$ şi $A \cup C(A) = X$
- $C(C(A)) = X - (X - A) = A$
- operaţiile $\cap$, $\cup$ sunt comutative:
  $A \cap B = B \cap A$ şi $A \cup B = B \cup A$
- operaţiile $\cap$, $\cup$ sunt asociative:
  $A \cap (B \cap D) = (A \cap B) \cap D = A \cap B \cap D$ şi
  $A \cup (B \cup D) = (A \cup B) \cup D = A \cup B \cup D$
- au loc proprietăţile de distributivitate a operatorilor $\cap$ şi $\cup$:
  $A \cap (B \cup D) = (A \cap B) \cup (A \cap D)$ şi

  $A \cup (B \cap D) = (A \cup B) \cap (A \cup D)$
- are loc idempotenţa pentru $\cap$ şi $\cup$:
  $A \cap A = A$ şi $A \cup A = A$
- au loc legile lui De Morgan:
  $C(A \cup B) = C(A) \cap C(B)$ şi
  $C(A \cap B) = C(A) \cup C(B)$
- proprietăţile de absorbţie:
  $A \cap (A \cup B) = A$ şi $A \cup (A \cap B) = A$

Ultimele două exemple sunt identice ca şi structură, diferă doar alegerea simbolurilor pentru operaţii.

Algebrele booleene şi funcţiile booleene sunt inseparabile. În cele ce urmează se vor studia funcţiile booleene, prin cele mai importante elemente ale lor: definiţia, formele canonice şi simplificarea acestora.

6.2. Funcţii booleene

În acest subcapitol şi capitolele următoare se vor nota operaţiile binare ale algebrei booleene binare cu simbolurile $\wedge$ şi $\vee$.

Definiţia 6.2.

Fie $\boldsymbol{B} = (B_2, \wedge, \vee, \bar{\ }, 0, 1)$ algebra booleană binară, $B_2 = \{0,1\}$ şi $n \in N^*$. O ***funcţie booleană de n variabile*** este o funcţie $f : B_2^n \to B_2$ definită recursiv astfel:

1. Funcţia *proiecţie*: $P_i : B_2^n \to B_2$, $P_i(x_1, ..., x_i, ..., x_n) = x_i$, care păstrează doar variabila x_i, este o funcţie booleană.
2. Dacă avem două funcţii booleene $f, g : B_2^n \to B_2$, atunci $f \wedge g, f \vee g, \overline{f}$ sunt funcţii booleene, unde:
   $(f \wedge g)(x_1, ..., x_n) = f(x_1, ..., x_n) \wedge g(x_1, ..., x_n)$
   $(f \vee g)(x_1, ..., x_n) = f(x_1, ..., x_n) \vee g(x_1, ..., x_n)$
   $\overline{f}(x_1, ..., x_n) = \overline{f(x_1, ..., x_n)}$
3. Orice funcţie booleană este obţinută prin aplicarea de un număr finit de ori a regulilor 1 şi 2 de mai sus.

Teorema 6.1.

$\forall n \in N^*$, există 2^{2^n} funcţii booleene de n variabile.

Teorema 6.2.

Structura $(\boldsymbol{FB(n)}, \wedge, \vee, \bar{\ }, f_0, f_{2^{2^n}-1})$ *este o algebră booleană, unde* $\boldsymbol{FB(n)}$ *este mulțimea tuturor funcțiilor booleene de* $n \in N^*$ *variabile, operațiile* $\wedge$, $\vee$ *și* $\bar{\ }$ *sunt cele din Definiția 6.2. și* $f_0(x_1,\ldots,x_n)=0$, $f_{2^{2^n}-1}(x_1,\ldots,x_n)=1$ *sunt funcțiile constante* 0, *respectiv* 1

Exemplul 6.4.

Pentru n=1, $B_2=\{0,1\}$ există următoarele funcții booleene de o variabilă

x	$f_0(x)$	$f_1(x)$	$f_2(x)$	$f_3(x)$
0	0	0	1	1
1	0	1	0	1

Se obțin deci $4 = 2^2 = 2^{2^1} = 2^{2^n}$ funcții booleene de o variabilă.

Exemplul 6.5.

Tabelul următor conține toate cele $2^{2^2} = 2^4 = 16$ funcții booleene de două variabile:

x	y	$f_0(x,y)$	$f_1(x,y)$	$f_2(x,y)$	$f_3(x,y)$	$f_4(x,y)$	$f_5(x,y)$	$f_6(x,y)$	$f_7(x,y)$
		contradicția	$x \wedge y$	$x \wedge \bar{y}$	x	$\bar{x} \wedge y$	y	$x \oplus y$	$x \vee y$
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

x	y	$f_8(x,y)$	$f_9(x,y)$	$f_{10}(x,y)$	$f_{11}(x,y)$	$f_{12}(x,y)$	$f_{13}(x,y)$	$f_{14}(x,y)$	$f_{15}(x,y)$
		$x \downarrow y$	$x \leftrightarrow y$	$\bar{y}$	$y \to x$	$\bar{x}$	$x \to y$	$x \uparrow y$	*tautologia*
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

Se observă că fiecare funcție are o funcție inversă, obținută ca negația sa. Perechile de funcții inverse sunt următoarele: (f_0, f_{15}), (f_1, f_{14}), (f_2, f_{13}), (f_3, f_{12}), (f_4, f_{11}), (f_5, f_{10}), (f_6, f_9), (f_7, f_8).

În tabelul anterior s-au folosit următoarele simboluri de operații:

$\downarrow$ (nor) – funcția lui Pierce, $x \downarrow y = \overline{(x \vee y)}$

$\uparrow$ (nand) – funcția lui Schaffer, $x \uparrow y = \overline{(x \wedge y)}$

$\to$ (implicația logică), $x \to y = \bar{x} \vee y$

$\leftrightarrow$ (echivalența logică), $x \leftrightarrow y = (x \to y) \wedge (y \to x)$

$\oplus$ (sau exclusiv), $x \oplus y = \overline{(x \leftrightarrow y)}$

Pentru o reprezentare uniformă a variabilelor și negațiile acestora, folosind puteri, se utilizează notația următoare:

$$x^\alpha = \begin{cases} x, \text{dacă } \alpha = 1 \\ \bar{x}, \text{dacă } \alpha = 0 \end{cases}, x \in \{0, 1\}$$

Pentru x, $\alpha \in \{0, 1\}$ au loc relațiile:

$x^0 = \bar{x}$, $x^1 = x$ și $0^0 = \bar{0} = 1$; $0^1 = 0$; $1^0 = \bar{1} = 0$; $1^1 = 1$

Astfel, se obține: $x^\alpha = \begin{cases} 1, \text{dacă } x = \alpha \\ 0, \text{dacă } x \neq \alpha \end{cases}$, $x, \alpha \in \{0, 1\}$

Formele canonice ale funcțiilor booleene

Teorema 6.3.

O funcție booleană $f: B_2^n \to B_2$, $n \in N^*$ *poate fi transformată în două forme echivalente:*

- ***forma canonică disjunctivă*** *(FCD):*

(1) $f(x_1,\ldots,x_n) = \bigvee_{(\alpha_1,\ldots,\alpha_n) \in B_2^n} (f(\alpha_1,\ldots,\alpha_n) \wedge x_1^{\alpha_1} \wedge \ldots \wedge x_n^{\alpha_n})$

- ***forma canonică conjunctivă*** *(FCC):*

(2) $f(x_1,\ldots,x_n) = \bigwedge_{(\alpha_1,\ldots,\alpha_n) \in B_2^n} (f(\alpha_1,\ldots,\alpha_n) \vee x_1^{\overline{\alpha_1}} \vee \ldots \vee x_n^{\overline{\alpha_n}})$

Teorema 6.4.

O funcție booleană $f: B_2^n \to B_2$, $n \in N^*$ *este unic determinată de valorile sale* $f(\alpha_1,\ldots,\alpha_n)$, *unde* $(\alpha_1,\ldots,\alpha_n) \in B_2^n$:

- forma canonică disjunctivă:

(1) $\Leftrightarrow$ (1') $f(x_1,\ldots,x_n) = \bigvee_{(\alpha_1,\ldots,\alpha_n) \in B_2^n \text{ și } f(\alpha_1,\ldots,\alpha_n)=1} (x_1^{\alpha_1} \wedge \ldots \wedge x_n^{\alpha_n})$

- forma canonică conjunctivă:

$$(2) \Leftrightarrow (2') \quad f(x_1,\dots,x_n) = \bigwedge_{(\alpha_1,\dots,\alpha_n)\in B_2^n \text{ și } f(\alpha_1,\dots,\alpha_n)=0} (x_1^{\overline{\alpha_1}} \vee \dots \vee x_n^{\overline{\alpha_n}})$$

Exemplul 6.6.

Funcțiile booleene de o singură variabilă, formele lor canonice disjunctive și cele simplificate sunt scrise în tabelul următor.

funcția booleană	forma canonică disjunctivă	forma simplificată
$f_0(x) = \begin{cases} 0, x=0 \\ 0, x=1 \end{cases}$	$f_0(x) = (0 \wedge x^0) \vee (0 \wedge x^1) = (0 \wedge \overline{x}) \vee (0 \wedge x)$	0
$f_1(x) = \begin{cases} 0, x=0 \\ 1, x=1 \end{cases}$	$f_1(x) = (0 \wedge x^0) \vee (1 \wedge x^1) = 0 \vee x$	x
$f_2(x) = \begin{cases} 1, x=0 \\ 0, x=1 \end{cases}$	$f_2(x) = (1 \wedge x^0) \vee (0 \wedge x^1) = (1 \wedge \overline{x}) \vee 0$	$\overline{x}$
$f_3(x) = \begin{cases} 1, x=0 \\ 1, x=1 \end{cases}$	$f_3(x) = (1 \wedge x^0) \vee (1 \wedge x^1) = \overline{x} \vee x$	1

Formele canonice disjunctivă și conjunctivă se scriu ușor, pornind de la expresia funcției, folosind formulele (1), respectiv (2).

Exemplul 6.7.

Pentru n=2, formele canonice disjunctive ale funcțiilor f_8, f_{13}, f_6, f_{12} din Exemplul 6.5. sunt obținute folosind formula (1) astfel:

$$f_8(x,y) = (1 \wedge x^0 \wedge y^0) \vee (0 \wedge x^0 \wedge y^1) \vee (0 \wedge x^1 \wedge y^0) \vee (0 \wedge x^1 \wedge y^1)$$
$$= \overline{x} \wedge \overline{y}$$

$$f_{13}(x,y) = (1 \wedge x^0 \wedge y^0) \vee (1 \wedge x^0 \wedge y^1) \vee (0 \wedge x^1 \wedge y^0) \vee (1 \wedge x^1 \wedge y^1)$$
$$= (\overline{x} \wedge \overline{y}) \vee (\overline{x} \wedge y) \vee (x \wedge y)$$

$$f_6(x,y) = (0 \wedge x^0 \wedge y^0) \vee (1 \wedge x^0 \wedge y^1) \vee (1 \wedge x^1 \wedge y^0) \vee (0 \wedge x^1 \wedge y^1)$$
$$= (\overline{x} \wedge y) \vee (x \wedge \overline{y})$$

$$f_{12}(x,y) = (1 \wedge x^0 \wedge y^0) \vee (1 \wedge x^0 \wedge y^1) \vee (0 \wedge x^1 \wedge y^0) \vee (0 \wedge x^1 \wedge y^1)$$
$$= (\overline{x} \wedge \overline{y}) \vee (\overline{x} \wedge y) = \overline{x} \wedge (\overline{y} \vee y) = \overline{x}$$

Se observă că indicii funcțiilor, reprezentați în baza 2 cu 4 cifre binare, coincid cu numerele binare de 4 cifre de pe coloanele corespunzătoare.

Pentru $f_{12}(x,y)$, $12=1100_{(2)}$ și coloana valorilor funcției este $\begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$.

Observație:

Forma canonică disjunctivă este utilă când există un număr mare de zerouri ale funcției și un număr mic de valori 1 (realizări), iar forma canonică conjunctivă este recomandată în caz contrar, când există un număr mic de zerouri și un număr mare de valori 1 ale funcției.

Exemplul 6.8.

Să se scrie cele două forme canonice ale funcției $f_9(x,y)$.

$$f_9(x,y) = (1 \wedge x^0 \wedge y^0) \vee (0 \wedge x^0 \wedge y^1) \vee (0 \wedge x^1 \wedge y^0) \vee (1 \wedge x^1 \wedge y^1)$$
$$= (\overline{x} \wedge \overline{y}) \vee (x \wedge y) \text{ --- FCD, se aplică distributivitatea}$$
$$= ((\overline{x} \wedge \overline{y}) \vee x) \wedge ((\overline{x} \wedge \overline{y}) \vee y) = (\overline{x} \vee x) \wedge (\overline{y} \vee x) \wedge (\overline{x} \vee y) \wedge (\overline{y} \vee y)$$
$$= 1 \wedge (x \vee \overline{y}) \wedge (\overline{x} \wedge y) \wedge 1$$
$$= (x \vee \overline{y}) \wedge (\overline{x} \vee y) \text{ --- FCC}$$

Exemplul 6.9.

Să se scrie formele canonice adecvate (conform observației precedente) pentru funcțiile booleene de două variabile:

$f_{15}, f_7, f_2, f_5, f_3, f_0$.

$$f_{15}(x,y) = (1 \wedge x^0 \wedge y^0) \vee (1 \wedge x^0 \wedge y^1) \vee (1 \wedge x^1 \wedge y^0) \vee (1 \wedge x^1 \wedge y^1)$$
$$= (\overline{x} \wedge \overline{y}) \vee (\overline{x} \wedge y) \vee (x \wedge \overline{y}) \vee (x \wedge y)$$
$$= ((\overline{x} \wedge \overline{y}) \vee (\overline{x} \wedge y)) \vee ((x \wedge \overline{y}) \vee (x \wedge y))$$
$$= (\overline{x} \wedge (\overline{y} \vee y)) \vee (x \wedge (\overline{y} \vee y)) = (\overline{x} \wedge 1) \vee (x \wedge 1) = \overline{x} \vee x$$
$$= 1 \text{ --- FCD}$$

$$f_7(x,y) = (0 \vee x^{\overline{0}} \vee y^{\overline{0}}) \wedge (1 \vee x^{\overline{0}} \vee y^{\overline{1}}) \wedge (1 \vee x^{\overline{1}} \vee y^{\overline{0}}) \wedge (1 \vee x^{\overline{1}} \vee y^{\overline{1}})$$
$$= (x^{\overline{0}} \vee y^{\overline{0}}) \wedge 1 \wedge 1 \wedge 1 = x^{\overline{0}} \vee y^{\overline{0}} = x^1 \vee y^1 = x \vee y \text{ --- FCC}$$

Pentru formulele (1'), respectiv (2'), vor apare doar termenii în care funcţia ia valoarea 1, respectiv 0. Astfel, funcţiile de mai sus au forma:

$$f_{15}(x,y)=(x^0 \wedge y^0)\vee(x^0 \wedge y^1)\vee(x^1 \wedge y^0)\vee(x^1 \wedge y^1)=1$$

$$f_7(x,y)=\overline{x^0}\vee\overline{y^0}=x\vee y$$

Se va utiliza formula (1') pentru funcţiile: $f_2(x,y)$, $f_5(x,y)$ şi formula (2') pentru funcţiile $f_3(x,y)$, $f_0(x,y)$.

$$f_2(x,y)=(x^1 \wedge y^0)=x\wedge\overline{y}$$

$$f_5(x,y)=(x^0 \wedge y^1)\vee(x^1 \wedge y^1)=(\overline{x}\wedge y)\vee(x\wedge y)=(\overline{x}\vee x)\wedge y=1\wedge y=y$$

$$f_3(x,y)=(\overline{x^0}\vee\overline{y^0})\wedge(\overline{x^0}\vee\overline{y^1})=(x\vee y)\wedge(x\vee\overline{y})=x\wedge(y\vee\overline{y})=x\wedge 1=x$$

$$\begin{aligned}f_0(x,y)&=(\overline{x^0}\vee\overline{y^0})\wedge(\overline{x^0}\vee\overline{y^1})\wedge(\overline{x^1}\vee\overline{y^0})\wedge(\overline{x^1}\vee\overline{y^1})\\&=(x\vee y)\wedge(x\vee\overline{y})\wedge(\overline{x}\vee y)\wedge(\overline{x}\vee\overline{y})\\&=((x\vee y)\wedge(x\vee\overline{y}))\wedge((\overline{x}\vee y)\wedge(\overline{x}\vee\overline{y}))\\&=(x\vee(y\wedge\overline{y}))\vee(\overline{x}\vee(y\wedge\overline{y}))=(x\vee 0)\wedge(\overline{x}\vee 0)=x\wedge\overline{x}=0\end{aligned}$$

Observaţii:

- Funcţia booleană f_0 nu poate fi scrisă în forma canonică disjunctivă, deoarece nu ia valoarea 1 pentru niciun argument.
- Funcţia booleană f_{2^n-1} nu poate fi scrisă în forma canonică conjunctivă, deoarece nu ia valoarea 0 pentru niciun argument

Definiţia 6.3.

Fie $f: B_2^n \to B_2$, $n\in N^*$ o funcţie booleană cu n variabile.

1. O conjuncţie de variabile se numeşte ***monom***.
2. Un monom care conţine toate cele n variabile se numeşte ***monom canonic*** sau ***minterm*** de n variabile. Are forma:

   $x_1^{\alpha_1}\wedge\ldots\wedge x_n^{\alpha_n}, \alpha_i\in B_2$.
3. Disjuncţia care conţine toate cele n variabile, având forma:

   $x_1^{\alpha_1}\vee\ldots\vee x_n^{\alpha_n}, \alpha_i\in B_2$ se numeşte ***maxterm*** de n variabile

Exemplul 6.10. Pentru n=3 au loc:

1. $x_1\wedge x_2$ este un monom, dar nu este un monom canonic;
2. $\overline{x_1}\wedge x_2\wedge x_3$ este un monom canonic (minterm) de trei variabile.
3. $x_1\vee\overline{x_2}\vee x_3$ este un maxterm de trei variabile.

Proprietăţi:

- $\forall n\in\mathbf{N}^*$, există exact 2^n maxtermi, notaţi cu $M_0, M_1,\ldots, M_{2^n-1}$ şi exact 2^n mintermi, notaţi cu $m_0, m_1,\ldots, m_{2^n-1}$.
- Un maxterm este o funcţie booleană care ia valoarea 0 doar pentru un argument.
- Un minterm este o funcţie booleană care ia valoarea 1 doar pentru un argument.

Observaţii:

1. Indicele unui minterm de n variabile este obţinut prin conversia în zecimal a numărului binar format cu cifrele ce reprezintă puterile celor n variabile din expresia mintermului.
2. Indicele unui maxterm de n variabile este obţinut prin conversia în zecimal a numărului binar format cu dualele cifrelor ce reprezintă puterile celor n variabile din expresia maxtermului.

Exemplul 6.11.

Pentru n=1, există $2^n=2^1=2$ mintermi şi maxtermi:

x	$m_0=M_1=\overline{x}$	$m_1=M_0=x$
0	1	0
1	0	1

Exemplul 6.12.

Pentru n =2, există $2^n=2^2=4$ mintermi şi maxtermi.

Tabelele valorilor şi expresiile mintermilor şi maxtermilor de două variabile sunt următoarele:

x	y	$m_0=\overline{x}\wedge\overline{y}$	$m_1=\overline{x}\wedge y$	$m_2=x\wedge\overline{y}$	$m_3=x\wedge y$	$M_0=x\vee y$	$M_1=x\vee\overline{y}$	$M_2=\overline{x}\vee y$	$M_3=\overline{x}\vee\overline{y}$
0	0	1	0	0	0	0	1	1	1
0	1	0	1	0	0	1	0	1	1
1	0	0	0	1	0	1	1	0	1
1	1	0	0	0	1	1	1	1	0

Se observă că mintermii au o unică valoare 1 pe coloanele lor, iar maxtermii au o unică valoare 0 pe coloanele corespunzătoare.

$$m_0 = m_{00_{(2)}} = x^0 \wedge y^0 = \bar{x} \wedge \bar{y}$$

$$m_3 = m_{11_{(2)}} = x^1 \wedge y^1 = x \wedge y$$

$$M_1 = M_{01_{(2)}} = x^{\bar{0}} \vee y^{\bar{1}} = x \vee \bar{y}$$

$$M_2 = M_{10_{(2)}} = x^{\bar{1}} \vee y^{\bar{0}} = \bar{x} \vee y$$

Exemplul 6.13.

Pentru n=3, există $2^n = 2^3 = 8$ mintermi şi maxtermi, dintre care amintim:

$$m_0 = m_{000_{(2)}} = x^0 \wedge y^0 \wedge z^0 = \bar{x} \wedge \bar{y} \wedge \bar{z}$$

$$M_0 = M_{000_{(2)}} = x^{\bar{0}} \vee y^{\bar{0}} \vee z^{\bar{0}} = x \vee y \vee z$$

$$m_3 = m_{011_{(2)}} = x^0 \wedge y^1 \wedge z^1 = \bar{x} \wedge y \wedge z$$

$$M_3 = M_{011_{(2)}} = x^{\bar{0}} \vee y^{\bar{1}} \vee z^{\bar{1}} = x \vee \bar{y} \vee \bar{z}$$

$$m_7 = m_{111_{(2)}} = x^1 \wedge y^1 \wedge z^1 = x \wedge y \wedge z$$

$$M_7 = M_{111_{(2)}} = x^{\bar{1}} \vee y^{\bar{1}} \vee z^{\bar{1}} = \bar{x} \vee \bar{y} \vee \bar{z}$$

Teorema 6.5.

1. *Conjuncţia a doi mintermi distincţi este* 0:

   $m_i \wedge m_j = 0, \quad \forall i \neq j, \ i, j = 0, \ldots, 2^{n-1}$

2. *Disjuncţia a doi maxtermi distincţi este* 1:

   $M_i \vee M_j = 1, \quad \forall i \neq j, \ i, j = 0, \ldots, 2^{n-1}$

3. *Un minterm şi un maxterm cu acelaşi indice sunt funcţii inverse:*

   $M_i = \overline{m}_i, \ \overline{M}_i = m_i, \quad \forall i = 0, \ldots, 2^{n-1}$

Exemplul 6.14.

1. $m_2 \wedge m_6 = (\bar{x} \wedge y \wedge \bar{z}) \wedge (x \wedge y \wedge \bar{z}) = (\bar{x} \wedge x) \wedge y \wedge \bar{z} = 0 \wedge y \wedge \bar{z} = 0$
2. $M_1 \vee M_3 = (x \vee y \vee \bar{z}) \vee (x \vee \bar{y} \vee \bar{z}) = (y \vee \bar{y}) \vee x \vee \bar{z} = 1 \vee x \vee \bar{z} = 1$
3. $\overline{m_3} = \overline{x^0 \wedge y^1 \wedge z^1} = \overline{\bar{x} \wedge y \wedge z} = x \vee \bar{y} \vee \bar{z} = x^1 \vee y^0 \vee z^0 =$
   $= x^{\bar{0}} \vee y^{\bar{1}} \vee z^{\bar{1}} = M_3$

Observaţii:

- Forma canonică conjunctivă, FCC, este conjuncţia maxtermilor corespunzători argumentelor pentru care funcţia ia valoarea 0.
- Forma canonică disjunctivă, FCD, este disjuncţia mintermilor corespunzători argumentelor pentru care funcţia ia valoarea 1

Exemplul 6.15.

Fie n=3 şi funcţia booleană $f(x_1, x_2, x_3)$ dată prin tabelul de mai jos. Să se scrie cele două forme canonice ale funcţiei.

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	M_2	M_4	M_5	m_0	m_1	m_3	m_6	m_7
0	0	0	1	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	0	1	0	0	0
0	1	0	0	0	1	1	0	0	0	0	0
0	1	1	1	1	1	1	0	0	1	0	0
1	0	0	0	1	0	1	0	0	0	0	0
1	0	1	0	1	1	0	0	0	0	0	0
1	1	0	1	1	1	1	0	0	0	1	0
1	1	1	1	1	1	1	0	0	0	0	1

S-au construit tabelele valorilor pentru mintermii şi maxtermii utilizaţi în construcţia celor două forme canonice ale funcţiei.

Forma canonică conjunctivă este dată de maxtermii corespunzători zerourilor funcţiei.

$$\mathrm{FCC}(f) = M_2 \wedge M_4 \wedge M_5 =$$
$$= (x_1 \vee \bar{x}_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2 \vee \bar{x}_3)$$

Forma canonică disjunctivă este dată de mintermii corespunzători valorilor 1 ale funcţiei.

$$\mathrm{FCD}(f) = m_0 \vee m_1 \vee m_3 \vee m_6 \vee m_7 =$$
$$= (\bar{x}_1 \wedge \bar{x}_2 \wedge \bar{x}_3) \vee (\bar{x}_1 \wedge \bar{x}_2 \wedge x_3) \vee (\bar{x}_1 \wedge x_2 \wedge x_3) \vee$$
$$\vee (x_1 \wedge x_2 \wedge \bar{x}_3) \vee (x_1 \wedge x_2 \wedge x_3)$$

Se observă că elementele din cele două forme canonice ale unei funcţii booleene sunt „complementare": indicii mintermilor/maxtermilor care lipsesc din forma canonică disjunctivă/conjunctivă vor fi indicii maxtermilor/mintermilor care apar în forma canonică conjunctivă/disjunctivă.

7. SIMPLIFICAREA FUNCȚIILOR BOOLEENE

Funcțiile booleene sunt folosite în practică mai ales pentru construirea circuitelor logice. După cum s-a putut observa în capitolul anterior, o funcție booleană are mai multe forme. În momentul construirii unui circuit, se va utiliza cea mai simplă dintre formele echivalente ale funcției, adică forma cu cel mai mic număr de apariții ale variabilelor.

Lucrările [6, 9, 17, 36, 44] descriu mai multe metode de simplificare a funcțiilor booleene, dintre care abordăm în acest capitol pe cele mai cunoscute: metoda diagramelor Veitch-Karnaugh, metoda lui Quine-Mc'Clusky și metoda lui Moisil.

7.1. Noțiuni de bază

Fie $B = (B_2 = \{0, 1\}, \wedge, \vee, \bar{}, 0, 1)$ algebra booleană binară definită în capitolul precedent.

Definiția 7.1.

Fie funcția booleană cu $n \in \mathbf{N}^*$ variabile, $f : B_2{}^n \to B_2$. Mulțimea $S_f = \{(x_1, x_2, ..., x_n) \in B_2{}^n \mid f(x_1, x_2, ..., x_n) = 1\}$, formată din grupurile $(x_1, x_2, ..., x_n) \in B_2{}^n$ pentru care funcția ia valoarea 1, se numește ***suportul funcției f***.

Exemplul 7.1.

- Pentru $n = 4$ variabile, suportul monomului $m = x_1 \wedge \bar{x}_3$, este:

  $S_m = \{(1,1,0,1), (1,0,0,1), (1,1,0,0), (1,0,0,0)\}$.
- Pentru $n = 3$ variabile și funcția booleană:

  $f(x_1, x_2, x_3) = (\bar{x}_1 \wedge x_2 \wedge \bar{x}_3) \vee (\bar{x}_1 \wedge \bar{x}_2 \wedge x_3)$, suportul funcției este:

  $S_f = \{(0,1,0), (0,0,1)\}$.

Se observă că pentru o funcție cu n variabile, aflată în forma canonică disjunctivă, suportul său este format din n-uplurile care reprezintă puterile variabilelor din mintermii ce formează expresia funcției.

Definiția 7.2.

Monomul m este ***mai mic sau egal*** cu monomul m' dacă și numai dacă suportul lui m este inclus sau egal cu suportul lui m':

$$m \le m' \Leftrightarrow S_m \subseteq S_{m'}.$$

Exemplul 7.2.

Fie monoamele $m = x_1 \wedge \bar{x}_2 \wedge x_4$ și $m' = x_1 \wedge \bar{x}_2$, n=4 variabile.

$S_m = \{(1,0,0,1), (1,0,1,1)\}$

$S_{m'} = \{(1,0,0,0), (1,0,0,1), (1,0,1,0), (1,0,1,1)\}$

Deoarece $S_m \subseteq S_{m'}$ are loc relația: $m \le m'$, adică $x_1 \wedge \bar{x}_2 \wedge x_4 \le x_1 \wedge \bar{x}_2$.

Se observă că toate variabilele din monomul mai mare, m', se regăsesc în scrierea monomului mai mic, m.

Definiția 7.3.

Pentru o funcție booleană cu n variabile, fie monoamele:

$m = x_{k_1}^{\alpha_{k_1}} \wedge .... \wedge x_{k_j}^{\alpha_{k_j}} \wedge x_{k_i} \wedge x_{k_l}^{\alpha_{k_l}} \wedge ... \wedge x_{k_s}^{\alpha_{k_s}}$ și

$m' = x_{k_1}^{\alpha_{k_1}} \wedge .... \wedge x_{k_j}^{\alpha_{k_j}} \wedge \overline{x_{k_i}} \wedge x_{k_l}^{\alpha_{k_l}} \wedge ... \wedge x_{k_s}^{\alpha_{k_s}}$, unde :

$1 \le k_1 < ... < k_j < k_i < k_l < ... < k_s \le n$,

1. m și m' se numesc ***monoame adiacente (vecine)*** deoarece acestea diferă doar prin puterea (semnul: negație sau nu) variabilei cu indicele „k_i".
2. ***factorizarea monoamelor*** m și m' este operația prin care se obține, eliminând variabila cu indicele „k_i", monomul $m \vee m' = x_{k_1}^{\alpha_{k_1}} \wedge .... \wedge x_{k_j}^{\alpha_{k_j}} \wedge x_{k_l}^{\alpha_{k_l}} \wedge ... \wedge x_{k_s}^{\alpha_{k_s}}$ mai mare decât m și m'.

Factorizarea se bazează pe aplicarea proprietății de comutativitate a operatorului „$\wedge$" și de distributivitate a operatorului „$\wedge$" față de operatorul „$\vee$". În cazul particular al definiției, notând cu y partea comună a celor două monoame se obține:

$$(y \wedge x_{k_i}) \vee (y \wedge \overline{x_{k_i}}) = y \wedge (x_{k_i} \vee \overline{x_{k_i}}) = y \wedge 1 = y.$$

Observație:

În continuare se va utiliza concatenarea variabilelor pentru a simboliza conjuncția acestora.

Astfel, monomul $m = x_1 \wedge \bar{x}_2 \wedge x_4$ se va scrie $m = x_1\bar{x}_2x_4$.

Definiția 7.4.

Simplificarea unei funcții booleene este operația de obținere a unei forme echivalente a funcției, având un număr cât mai mic de apariții ale variabilelor.

Astfel, dacă funcția f este în forma canonică disjunctivă, adică este o disjuncție de monoame canonice (mintermi), într-un număr finit de pași se poate obține o formă simplificată a funcției. Pentru aceasta, se va aplica operația de factorizare, obținând monoame cât mai mari care să acopere mintermii inițiali.

Definiția 7.5.

Mulțimea $M(f)$ se numește ***mulțimea monoamelor maximale*** ale funcției booleene $f : B_2^n \to B_2$ dacă au loc:

1. $\forall m \in M(f), m \in FB(n), m \le f$ și
2. $\forall m \in M(f), \nexists m' \in FB(n)$ astfel încât $m < m' \le f$.

Exemplul 7.3.

Fie funcția booleană: $f(x, y) = (x \wedge y) \vee (\bar{x} \wedge y) = m_3 \vee m_1$

Se observă următoarele:

- $m_3 < f$ și $m_1 < f$;
- $m_3 < y \le f$ și $m_1 < y \le f$.

Deci $m_1, m_3 \notin M(f)$ și $M(f) = \{y\}$

Observație:

Monoamele maximale sunt fie mintermi, fie se obțin prin operația de factorizare

Definiția 7.6.

Mulțimea $C(f)$ se numește ***mulțimea monoamelor centrale*** ale funcției booleene $f : B_2^n \to B_2$ dacă au loc:

1. $C(f) \subseteq M(f)$ și
2. $\forall m \in C(f): m \not\le \bigvee_{m' \in M(f)-\{m\}} m'$

Observație:

Un monom maximal este monom central pentru o funcție f, dacă acesta nu este mai mic decât disjuncția tuturor celorlalte monoame maximale ale funcției f

Proprietăți:

Pentru $f : B_2^n \to B_2$, o funcție booleană cu n variabile, au loc:

- $M(f)$ este totdeauna o mulțime nevidă, $M(f) \ne \varnothing$. dar $C(f)$ poate fi mulțimea vidă.
- $f(x_1,\dots,x_n) = \bigvee_{m \in M(f)} m$

Algoritmul de simplificare a funcțiilor booleene:

Date de intrare: f – o funcție booleană în formă canonică disjunctivă

Date de ieșire: $f'_1, f'_2, \dots, f'_k$ – variantele simplificate ale funcției f

Se determină $M(f)$ și $C(f)$.

dacă $M(f) = C(f)$

atunci $f' = \bigvee_{m \in M(f)} m$

STOP1 // caz1 --- soluție unică

altfel

dacă $C(f) \ne \varnothing$

atunci $g \overset{not.}{=} \bigvee_{m \in C(f)} m$

$f'_i = g \vee h_i,\ i = \overline{1,k}$, unde h_i este disjuncția unui număr cât mai mic de monoame maximale astfel încât $S_{h_i} = S_f - S_g$

STOP2 // caz2 --- k soluții

altfel // nu există monoame centrale

$f'_i = h_i,\ i = \overline{1,k}$, unde h_i este disjuncția unui număr cât mai mic de monoame maximale astfel încât $S_{h_i} = S_f$

STOP3 // caz3 --- k soluții

sf_dacă

sf_dacă

Sf_algoritm

A simplifica o funcție booleană înseamnă a o acoperi (disjuncție) cu un număr cât mai mic de monoame maximale, eventual centrale. Pot exista mai multe forme simplicate ale aceleiași funcții booleene.

Algoritmul prezentat anterior este un algoritm general de simplificare a funcțiilor booleene. Acest algoritm este aplicat pas cu pas în metoda grafică a diagramelor Veitch-Karnaugh și metoda analitică a lui Quine-Mc'Clusky. A treia metodă de simplificare, este metoda algebrică a lui Moisil care utilizează axiomele algebrei propoziționale.

Toate cele trei metode de simplificare au ca intrare funcţia booleană în forma canonică disjunctivă (FCD) şi constau din următorii paşi:

1. factorizarea, ce are ca rezultat mulţimea monoamelor maximale;
2. determinarea monoamelor centrale utilizate în scrierea formelor simplificate;
3. identificarea cazului din algoritmul de simplificare;
4. pentru cazurile al doilea şi al treilea se determină şi se scriu toate formele simplificate, iar pentru primul caz se va scrie unica formă simplificată.

7.2. Metoda diagramelor Veitch-Karnaugh

Metoda se bazează pe reprezentarea grafică, sub forma unei diagrame, a mintermilor care compun expresia formei canonice disjunctive a funcţiei.

Este foarte utilă pentru funcţii cu un număr mic de variabile: 2, 3 sau 4. Pentru un număr de variabile mai mare decât 4 este destul de greu de aplicat.

Pentru a se putea realiza factorizarea, se construieşte una dintre diagramele următoare, corespunzătoare numărului de variabile ale funcţiei ce se simplifică. În diagramă este notată poziţia fiecărui minterm. Relaţia de vecinătate grafică dintre mintermi corespunde relaţiei de adiacenţă (vecinătate) din Definiţia 7.3.

nr. variabile	Diagrama Veitch	Diagrama Karnaugh

2

	x_1	$\bar{x}_1$
x_2	m_3	m_1
$\bar{x}_2$	m_2	m_0

$x_1 \backslash x_2$	0	1
0	m_0	m_1
1	m_2	m_3

3

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$
x_2	m_7	m_6	m_2	m_3
$\bar{x}_2$	m_5	m_4	m_0	m_1
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3

$x_1 \backslash x_2x_3$	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

4

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$	
x_2	m_{15}	m_{13}	m_5	m_7	x_4
x_2	m_{14}	m_{12}	m_4	m_6	$\bar{x}_4$
$\bar{x}_2$	m_{10}	m_8	m_0	m_2	$\bar{x}_4$
$\bar{x}_2$	m_{11}	m_9	m_1	m_3	x_4
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3	

$x_1x_2 \backslash x_3x_4$	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

Suportul a doi mintermi ce sunt reprezentaţi în căsuţe vecine în diagramă diferă doar printr-o singură valoare de 1. Aceste diagrame trebuie privite circular. Astfel, prima linie/coloană este vecină (adiacentă) cu ultima linie/coloană. Diagramele nu sunt unice, variabilele putând fi interschimbate.

Din cele două diagrame pentru n=4 se observă perechile de mintermi adiacenţi:

- $m_{12}= x_1x_2\bar{x}_3\bar{x}_4$ şi $m_8= x_1\bar{x}_2\bar{x}_3\bar{x}_4$ diferă prin variabila x_2
- m_0 şi m_2 diferă prin variabila x_3
- m_1 şi m_9 diferă prin variabila x_1
- m_5 şi m_7 diferă prin variabila x_3
- m_7 şi m_6 diferă prin variabila x_4

Pornindu-se de la forma canonică disjunctivă a funcţiei, se vor reprezenta în diagramă doar mintermii expresiei funcţiei. Aceştia pot fi scrişi ca atare în diagramă, sau se poate nota valoarea 1 în căsuţa corespunzătoare.

Din diagramă se obţin monoamele maximale, aplicând factorizarea în toate modurile posibile.

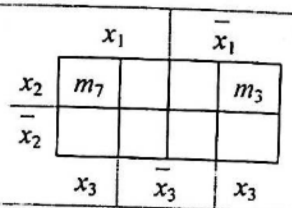
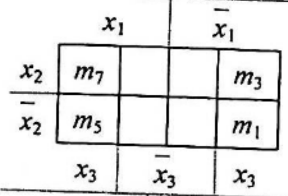
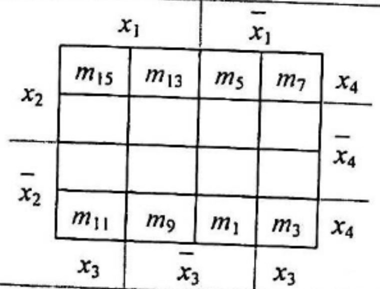
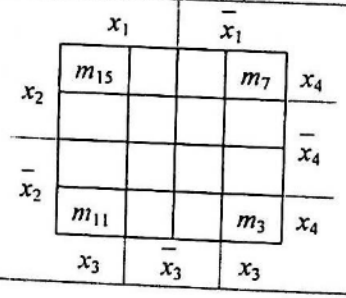
Se poate aplica factorizarea simplă între doi mintermi adiacenţi (aflaţi în două căsuţe vecine), sau se poate aplica o *k–factorizare* între 2^k mintermi aflaţi în căsuţe vecine.

Pentru o funcţie booleană cu n variabile se recomandă încercarea unei *n–factorizări*, apoi *n-1–factorizări* şi aşa mai departe, până la factorizări simple, astfel obţinându-se toate monoamele maximale.

În această reprezentare sub forma diagramelor, un monom maximal este central dacă, grupul de mintermi din care provine conţine cel puţin un minterm (căsuţă) încercuit o singură dată.

Observaţie:

În general este destul de uşor de observat grupul celor 2^k mintermi adiacenţi. Deoarece prima şi ultima linie/coloană sunt de asemenea vecine, există anumite grupuri mai greu de observat, cum sunt următoarele.

Nr. de variabile care se simplifică	Diagrama Veitch	Monomul maximal din diagramă
1 din 3 factorizare simplă		x_2x_3
2 din 3 factorizare de ordinul 2		x_3
3 din 4 factorizare de ordinul 3		x_4
2 din 4 factorizare de ordinul 2		x_3x_4

Exemplul 7.4.

Se doreşte simplificarea funcţiei:

$$f(x_1, x_2, x_3) = x_1x_2x_3 \vee x_1x_2\bar{x}_3 \vee \bar{x}_1x_2\bar{x}_3 \vee x_1\bar{x}_2x_3 \vee x_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2x_3$$
$$= m_7 \vee m_6 \vee m_2 \vee m_5 \vee m_4 \vee m_1$$

Se construiesc diagramele Veitch şi Karnaugh pentru f, dar se va utiliza în simplificare doar diagrama Veitch.

Diagrama Veitch corespunzătoare este:

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$
x_2	m_7	m_6	m_2	
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3

Diagrama Karnaugh corespunzătoare este:

$x_1 \backslash x_2x_3$	00	01	11	10
0		m_1		m_2
1	m_4	m_5	m_7	m_6

1. **Factorizarea**

- Se caută 2^k, $k \leq 3$ (numărul variabilelor) căsuţe ocupate, vecine pe orizontală sau verticală, cu k cât mai mare. Atenţie la grupurile de pe „frontieră"!
- Se încercuiesc toate grupurile (grupurile incluse în altele mai mari se neglijează). În cazul funcţiei de mai sus, există un grup de 4 mintermi vecini, un grup normal de 2 mintermi vecini şi unul de 2 mintermi vecini pe frontieră:

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$
x_2	m_7	m_6	m_2	
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3

- Monoamele maximale se obţin prin factorizare.
Dacă într-un grup de mintermi vecini, o variabilă apare atât negată, cât şi nenegată, atunci aceasta se elimină. Monomul maximal este format din variabilele comune tuturor mintermilor grupului.

Se vor evidenţia grup cu grup monoamele maximale.

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$
x_2	m_7	m_6	m_2	
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3

Doar variabila x_1 își păstrează forma, deci are loc o dublă factorizare, monomul maximal obținut este $x_1 = m_7 \vee m_6 \vee m_5 \vee m_4$.

	x_1		$\bar{x}_1$	
x_2	m_7	m_6	m_2	
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$		x_3

Monomul maximal: $x_2\bar{x}_3 = m_6 \vee m_2$ (simplă factorizare conform diagramei precedente).

	x_1		$\bar{x}_1$	
x_2	m_7	m_6	m_2	
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$		x_3

Monomul maximal: $\bar{x}_2x_3 = m_5 \vee m_1$ (simplă factorizare).

Mulțimea monoamelor maximale este: $M(f) = \{x_1, x_2\bar{x}_3, \bar{x}_2x_3\}$

2. **Monoamele centrale**: un monom maximal este central dacă, grupul corespunzător conține cel puțin o căsuță încercuită o singură dată. Aceste grupuri se hașurează.

	x_1		$\bar{x}_1$	
x_2	m_7	m_6	m_2	
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$		x_3

Se observă că, grupul corespunzător monomului maximal x_1 conține două căsuțe încercuite o singură dată: m_7 și m_4. Pentru monomul maximal $x_2\bar{x}_3$ căsuța cu m_2 este încercuită o singură dată, iar pentru $\bar{x}_2x_3$, mintermul m_1 este încercuit o singură dată.

Astfel, mulțimea monoamelor centrale este: $C(f) = \{x_1, x_2\bar{x}_3, \bar{x}_2x_3\}$

3. Identificarea cazului din algoritmul de simplificare: $M(f) = C(f)$. Este **primul caz** al algoritmului de simplificare.

4. Există o soluție unică pentru forma simplificată a funcției, obținută ca disjuncție a tuturor monoamelor maximale:

$f'(x_1, x_2, x_3) = x_1 \vee x_2\bar{x}_3 \vee \bar{x}_2x_3$.

Exemplul 7.5.

Se șterge primul minterm al funcției din exemplul precedent și se obține funcția de mai jos, care se va simplifica.

$$f(x_1, x_2, x_3) = x_1x_2\bar{x}_3 \vee \bar{x}_1x_2\bar{x}_3 \vee x_1\bar{x}_2x_3 \vee x_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2x_3 =$$
$$= m_6 \vee m_2 \vee m_5 \vee m_4 \vee m_1$$

Diagrama Karnaugh corespunzătoare este:

$x_1 \backslash x_2x_3$	00	01	11	10
0		m_1		m_2
1	m_4	m_5		m_6

1. $M(f) = \{x_1\bar{x}_2, x_1\bar{x}_3, x_2\bar{x}_3, \bar{x}_2x_3\}$

Pentru obținerea monoamelor maximale s-au aplicat doar factorizări simple, deoarece conform diagramei Karnaugh există doar grupuri de câte doi mintermi vecini.

$max_1 = x_1\bar{x}_2 = m_5 \vee m_4$

$max_2 = x_1\bar{x}_3 = m_4 \vee m_6$

$max_3 = x_2\bar{x}_3 = m_6 \vee m_2$

$max_4 = \bar{x}_2x_3 = m_5 \vee m_1$

2. Mulțimea monoamelor centrale este: $C(f) = \{x_2\bar{x}_3, \bar{x}_2x_3\}$

max_1 nu este monom central deoarece $max_1 \leq max_2 \vee max_4$;
max_2 nu este monom central deoarece $max_2 \leq max_1 \vee max_3$;
max_3 este monom central deoarece:
$max_3 \not\leq max_1 \vee max_2 \vee max_4$, mintermul m_2 este încercuit o singură dată fiind acoperit doar de max_3;
max_4 este monom central deoarece:
$max_4 \not\leq max_1 \vee max_2 \vee max_3$, mintermul m_1 este încercuit o singură dată fiind acoperit doar de max_4;

3. $M(f) \neq C(f)$ și $C(f) \neq \varnothing$. Este **cazul al doilea** de aplicare a algoritmului de simplificare.

Se notează cu $g(x_1, x_2, x_3) \overset{not.}{=} x_2\bar{x}_3 \vee \bar{x}_2x_3 = max_3 \vee max_4$
(disjuncția monoamelor centrale)

4. Alegerea celor mai simple soluții:

Trebuie să căutăm funcțiile h, cele mai simple, care să acopere mintermii încă neacoperiți de monoamele centrale astfel:

- Se caută grupul cel mai mare (cu cele mai multe căsuțe nehașurate), iar monomul maximal corespunzător se adaugă la soluție (în funcția ajutătoare h).
- Dacă mai multe grupuri îndeplinesc condiția de mai sus, se vor lua fiecare pe rând, în final, putându-se obține mai multe soluții.
- Se poate hașura grupul obținut și se repetă cei doi pași anteriori până când se hașurează toți mintermii din diagramă.

$x_1 \backslash x_2x_3$	00	01	11	10
0		m_1		m_2
1	m_4	m_5		m_6

Singurul minterm nehașurat, m_4, este acoperit de două monoame maximale cu același număr de variabile: max_1 și max_2.

Astfel există două funcții ajutătoare h:

$$h_1(x_1, x_2, x_3) \overset{not.}{=} x_1\bar{x}_2 = max_1 \text{ și } h_2(x_1, x_2, x_3) \overset{not.}{=} x_1\bar{x}_3 = max_2$$

Corespunzător există două forme simplificate ale funcției inițiale:

$$f'_1(x_1, x_2, x_3) = g \vee h_1 = x_2\bar{x}_3 \vee \bar{x}_2x_3 \vee x_1\bar{x}_2 = max_3 \vee max_4 \vee max_1$$

$$f'_2(x_1, x_2, x_3) = g \vee h_2 = x_2\bar{x}_3 \vee \bar{x}_2x_3 \vee x_1\bar{x}_3 = max_3 \vee max_4 \vee max_2$$

Exemplul 7.6.

Fie funcția obținută din exemplul anterior prin adăugarea mintermului m_3.

$$f(x_1, x_2, x_3) = x_1x_2\bar{x}_3 \vee \bar{x}_1x_2\bar{x}_3 \vee \bar{x}_1x_2x_3 \vee x_1\bar{x}_2x_3 \vee x_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2x_3 =$$
$$= m_6 \vee m_2 \vee m_3 \vee m_5 \vee m_4 \vee m_1$$

Diagrama Veitch corespunzătoare este:

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$
x_2		m_6	m_2	m_3
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3

1. Mulțimea monoamelor maximale este:

$M(f)= \{x_1\bar{x}_2, x_1\bar{x}_3, x_2\bar{x}_3, \bar{x}_2x_3, \bar{x}_1x_2, \bar{x}_1x_3\}$

$max_1= x_1\bar{x}_2 = m_5\vee m_4$ $\quad$ $max_2= x_1\bar{x}_3 = m_4\vee m_6$

$max_3= x_2\bar{x}_3 = m_6\vee m_2$ $\quad$ $max_4= \bar{x}_2x_3 = m_5\vee m_1$

$max_5= \bar{x}_1x_2 = m_2\vee m_3$ $\quad$ $max_6= \bar{x}_1x_3 = m_3\vee m_1$

2. $C(f)=\varnothing$ deoarece nu există niciun minterm încercuit o singură dată.

3. $M(f)\neq C(f)$ și $C(f) = \varnothing$, deci este **cazul al treilea** al algoritmului de simplificare.

4. Deoarece toate grupurile sunt la fel de mari și toți mintermii sunt exact de două ori încercuiți, soluția va fi dată de o încercuire a tuturor mintermilor cu un număr minim de grupuri (monoame maximale), adică, cu cât mai puține suprapuneri:

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$
x_2		m_6	m_2	m_3
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3

$$f'_1(x_1, x_2, x_3) = x_1\bar{x}_2 \vee x_2\bar{x}_3 \vee \bar{x}_1x_3 = max_1 \vee max_3 \vee max_6$$

	x_1	x_1	$\bar{x}_1$	$\bar{x}_1$
x_2		m_6	m_2	m_3
$\bar{x}_2$	m_5	m_4		m_1
	x_3	$\bar{x}_3$	$\bar{x}_3$	x_3

$$f'_2(x_1, x_2, x_3) = x_1\bar{x}_3 \vee \bar{x}_1x_2 \vee \bar{x}_2x_3 = max_2 \vee max_5 \vee max_4$$

S-au obținut astfel două forme simplificate f'_1 și f'_2 ale funcției inițiale.

Observație:

Din exemplele anterioare, se poate observa faptul că adăugarea sau ștergerea unui minterm poate schimba cazul de simplificare.

Exemplul 7.7.

Să se simplifice următoarea funcție booleană de 4 variabile:

$$f(x_1, x_2, x_3, x_4) = x_1\bar{x}_3x_4 \vee x_1\bar{x}_2x_3\bar{x}_4 \vee \bar{x}_2x_4 \vee x_1\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2x_3x_4.$$

Pentru a se construi uşor diagrama Veitch, identificăm mai întâi mintermii funcţiei. Se adaugă fiecărui monom toate variabilele lipsă, atât negate cât şi simple, obţinând forma canonică disjunctivă.

$f(x_1, x_2, x_3, x_4) =$

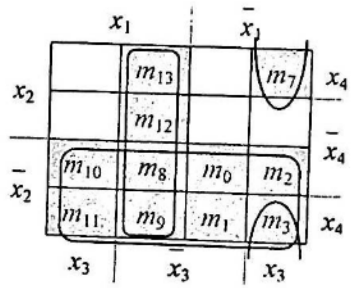
$$= x_1\bar{x}_2\bar{x}_3x_4 \vee x_1x_2\bar{x}_3x_4 \vee x_1\bar{x}_2x_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2x_3x_4 \vee x_1\bar{x}_2x_3x_4 \vee$$
$$\vee x_1\bar{x}_2x_3x_4 \vee x_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee x_1x_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2x_3\bar{x}_4 \vee$$
$$\vee \bar{x}_1\bar{x}_2x_3x_4 \vee \bar{x}_1x_2x_3x_4$$

Mintermii (abstracţie făcând de repetiţiile unora dintre ei) sunt:

$f(x_1, x_2, x_3, x_4) =$

$$= m_9 \vee m_{13} \vee m_{10} \vee m_1 \vee m_3 \vee m_{11} \vee m_8 \vee m_{12} \vee m_0 \vee m_2 \vee m_7$$

Diagrama Veitch corespunzătoare este:



1. Mulţimea monoamelor maximale este:

   $M(f) = \{ x_1\bar{x}_3, \bar{x}_2, \bar{x}_1x_3x_4 \}$

   $max_1 = x_1\bar{x}_3 = m_{13} \vee m_{12} \vee m_8 \vee m_9$ – factorizare de ordinul 2

   $max_2 = \bar{x}_2 = m_{10} \vee m_8 \vee m_0 \vee m_2 \vee m_{11} \vee m_9 \vee m_1 \vee m_3$

   – factorizare de ordinul 3

   $max_3 = \bar{x}_1x_3x_4 = m_7 \vee m_3$ – factorizare simplă

2. $C(f) = M(f)$
3. Este **primul caz** al algoritmului de simplificare.
4. Deci, există o soluţie unică pentru forma simplificată a funcţiei, obţinută ca disjuncţie a tuturor monoamelor maximale:

$f'(x_1, x_2, x_3, x_4) = x_1\bar{x}_3 \vee \bar{x}_2 \vee \bar{x}_1x_3x_4.$

7.3. Metoda analitică a lui Quine-Mc'Clusky

Această metodă de simplificare a funcţiilor booleene se bazează pe completarea a două tabele ajutătoare, unul pentru factorizare, utilizat la calcularea mulţimii monoamelor maximale şi unul pentru identificarea mulţimii monoamelor centrale. Funcţia booleană trebuie furnizată în forma canonică disjunctivă.

Spre deosebire de metoda diagramelor, care pentru mai mult de 4 variabile este dificil de aplicat, această metodă poate fi utilizată pentru oricâte variabile şi poate fi uşor implementată, datorită caracterului său analitic.

Primul pas este acela de a ordona mulţimea suport a funcţiei cu n variabile, crescător sau descrescător, după numărul de valori 1 conţinut de fiecare n-uplu.

Mintermii din expresia funcţiei se vor reprezenta prin intermediul puterilor variabilelor, într-un tabel, fiecare pe o linie, în ordinea crescătoare/descrescătoare a numărului valorilor de 1 din n-uplurile suportului funcţiei. Capul de tabel conţine numele variabilelor funcţiei. Astfel, se vor forma grupuri de mintermi, un grup conţinând toţi mintermii cu acelaşi număr de valori 1 ca puteri ale variabilelor. Aceste grupuri se vor delimita prin bare orizontale. După reprezentarea expresiei funcţiei în tabel, se trasează o dublă bară orizontală.

Doar două grupuri vecine pot să conţină monoame vecine (conform Definiţiei 7.3), deci factorizarea poate avea loc doar între monoamele a două grupuri vecine.

Operaţia de factorizare constă în introducerea unei linii noi în tabel, care conţine aceleaşi valori (0, 1 sau –) pe coloanele variabilelor comune celor două monoame care participă la factorizare şi simbolul „–” pe coloana corespunzătoare variabilei eliminate. Monoamele care participă la factorizare vor fi bifate (stânga) deoarece acestea nu sunt maximale. Rezultatele factorizării a două grupuri de monoame vecine vor forma un nou grup (delimitat printr-o bară orizontală) de monoame în tabel, care poate fi utilizat mai departe în alte factorizări de ordin superior.

Sfârşitul tuturor factorizărilor dintre două grupuri vecine de monoame cu acelaşi număr de simboluri „–” se marchează cu o dublă bară orizontală, pentru a sugera grafic faptul că s-a încheiat factorizarea de un anumit ordin şi se trece la factorizare de ordin mai mare. Acest proces continuă până când nu se mai pot face factorizări între două grupuri vecine delimitate printr-o singură bară orizontală. Tabelul se încheie cu o triplă bară orizontală.

Observaţie:

Simbolul „–” nu se combină cu nimic! Astfel, începând cu factorizări de ordinul 2, se vor compara doar linii din grupuri vecine ce au simbolul „–” pe aceeaşi poziţie.

Mulțimea **monoamelor maximale** este formată din toate monoamele corespunzătoare liniilor nebifate din tabel.

Identificarea mulțimii **monoamelor centrale** se face utilizând un tabel care arată corespondența dintre monoamele maximale (pe coloane) și mintermii (pe linii) care au contribuit prin factorizare la obținerea acestora. O căsuță din tabel se marchează cu o steluță dacă mintermul corespunzător liniei a fost utilizat pentru obținerea monomului maximal de pe coloană. Definiția unui monom central se traduce în această reprezentare în existența pe coloana monomului respectiv a unei steluțe care este unică pe linia sa.

Disjuncția tuturor monoamelor centrale va aparține tuturor formelor simplificate ale funcției.

Mintermii neacoperiți de monoamele centrale se vor acoperi în toate modurile posibile utilizând un număr cât mai mic de monoame maximale cât mai mari și cu cât mai puține suprapuneri, conform tabelului precedent, rezultând toate formele simplificate ale funcției inițiale.

Exemplul 7.8.

Fie funcția booleană cu trei variabile dată prin tabela valorilor sale.

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Forma canonică disjunctivă a funcției date este:

$$f(x_1,x_2,x_3) = \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2x_3 \vee \bar{x}_1x_2x_3 \vee x_1\bar{x}_2\bar{x}_3 \vee x_1x_2\bar{x}_3 \vee x_1x_2x_3$$
$$= m_0 \vee m_1 \vee m_3 \vee m_4 \vee m_6 \vee m_7$$

S_f={(0,0,0), (0,0,1), (0,1,1), (1,0,0), (1,1,0), (1,1,1)}

Suportul funcției inițiale, ordonat crescător după numărul de valori 1 din tripletele suportului funcției este:

S_f={(0,0,0), (1,0,0), (0,0,1), (1,1,0), (0,1,1), (1,1,1)}

Grupul		x_1	..2	x_3	
I	√	0	0	0	m_0
II	√	1	0	0	m_4
	√	0	0	1	m_1
III	√	1	1	0	m_6
	√	0	1	1	m_3

IV	√	1	1	1	m_7
V=I+II		–	0	0	$m_0 \vee m_4 = \bar{x}_2\bar{x}_3 = max_1$
		0	0	–	$m_0 \vee m_1 = \bar{x}_1\bar{x}_2 = max_2$
VI=II+III		1	–	0	$m_4 \vee m_6 = x_1\bar{x}_3 = max_3$
		0	–	1	$m_1 \vee m_3 = \bar{x}_1x_3 = max_4$
VII=III+IV		1	1	–	$m_6 \vee m_7 = x_1x_2 = max_5$
		–	1	1	$m_3 \vee m_7 = x_2x_3 = max_6$

$$M(f) = \{max_1, max_2, max_3, max_4, max_5, max_6\} =$$
$$= \{\bar{x}_2\bar{x}_3, \bar{x}_1\bar{x}_2, x_1\bar{x}_3, \bar{x}_1x_3, x_1x_2, x_2x_3\} =$$

Tabelul următor arată corespondența dintre mintermii funcției și monoamele maximale:

mon. max. / mintermi	max_1	max_2	max_3	max_4	max_5	max_6
m_0	*	*				
m_4	*		*			
m_1		*		*		
m_6			*		*	
m_3				*		*
m_7					*	*

Se observă că nu există nicio steluță unică pe linia sa, deci $C(f) = \varnothing$. Este **cazul al treilea** al algoritmului de simplificare. Se vor căuta cele mai simple soluții, adică se vor alege monoame maximale astfel încât intersecția liniilor având steluțe comune cu coloanele monoamelor maximale selectate să fie cât mai mică și numărul monoamelor maximale alese să fie, de asemenea, cât mai mic, iar reuniunea acestor linii să fie egală cu mulțimea tuturor liniilor tabelului.

Există două forme simplificate ale funcției date:

$f'_1(x_1, x_2, x_3) = \bar{x}_2\bar{x}_3 \vee \bar{x}_1x_3 \vee x_1x_2 = max_1 \vee max_4 \vee max_5$ (soluția hașurată)

$f'_2(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2 \vee x_1\bar{x}_3 \vee x_2x_3 = max_2 \vee max_3 \vee max_6$

Exemplul 7.9.

Să se simplifice următoarea funcție booleană de 4 variabile:

$$f(x_1, x_2, x_3, x_4) = x_1\bar{x}_2x_3\bar{x}_4 \vee \bar{x}_1x_2x_3 \vee \bar{x}_1x_2\bar{x}_3x_4 \vee x_1x_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_3x_4 \vee \bar{x}_1x_3 \vee \bar{x}_2\bar{x}_3x_4$$

Se poate obține forma canonică disjunctivă a funcției și deci mintermii funcției prin intermediul suportului funcției, care este rezultatul reuniunii suporturile tuturor monoamelor din expresia funcției:

$S_f=\{(1,0,1,0)\} \cup \{(0,1,1,0),(0,1,1,1)\} \cup \{(0,1,0,1)\} \cup \{(1,1,0,0)\} \cup$
$\cup \{(0,0,0,1),(0,1,0,1)\} \cup \{(0,0,1,0),(0,0,1,1),(0,1,1,0),(0,1,1,1)\} \cup$
$\cup \{(0,0,0,1),(1,0,0,1)\} =$
$=\{(0,1,1,1),(1,0,1,0),(0,1,1,0),(1,1,0,0),(0,1,0,1),(1,0,0,1),(0,0,1,1),$
$(0,0,0,1),(0,0,1,0)\}$

Forma canonică disjunctivă a funcției este următoarea:

$f(x_1, x_2, x_3, x_4)= m_7 \vee m_{10} \vee m_6 \vee m_{12} \vee m_5 \vee m_9 \vee m_3 \vee m_1 \vee m_2$

1. Factorizarea (cu suportul funcției ordonat descrescător după numărul de valori 1 din cvadruple):

Grupul		x_1	x_2	x_3	x_4	
I	√	0	1	1	1	m_7
	√	1	0	1	0	m_{10}
	√	0	1	1	0	m_6
II		1	1	0	0	$m_{12}= x_1x_2\bar{x}_3\bar{x}_4 = max_1$
	√	0	1	0	1	m_5
	√	1	0	0	1	m_9
	√	0	0	1	1	m_3
III	√	0	0	0	1	m_1
	√	0	0	1	0	m_2
	√	0	1	1	–	$m_7 \vee m_6$
IV=I+II	√	0	1	–	1	$m_7 \vee m_5$
	√	0	–	1	1	$m_7 \vee m_3$
		–	0	1	0	$m_{10} \vee m_2= \bar{x}_2x_3\bar{x}_4 = max_2$
	√	0	–	1	0	$m_6 \vee m_2$
V=II+III	√	0	–	0	1	$m_5 \vee m_1$
		–	0	0	1	$m_9 \vee m_1= \bar{x}_2\bar{x}_3x_4 = max_3$
	√	0	0	–	1	$m_3 \vee m_1$
	√	0	0	1	–	$m_3 \vee m_2$
VI=IV+V		0	–	1	–	$m_7\vee m_6\vee m_3\vee m_2= \bar{x}_1x_3 = max_4$
		0	–	–	1	$m_7\vee m_5\vee m_3\vee m_1= \bar{x}_1x_4 = max_5$

$M(f) = \{max_1, max_2, max_3, max_4, max_5\}=$
$= \{ x_1x_2\bar{x}_3\bar{x}_4, \bar{x}_2x_3\bar{x}_4, \bar{x}_2\bar{x}_3x_4, \bar{x}_1x_3, \bar{x}_1x_4\}$

Pentru cele două factorizări duble au loc:

$(m_7 \vee m_6) \vee (m_3 \vee m_2) = (m_7 \vee m_3) \vee (m_6 \vee m_2)$ și
$(m_7 \vee m_5) \vee (m_3 \vee m_1) = (m_7 \vee m_3) \vee (m_5 \vee m_1)$,

deci mai multe factorizări de ordin superior conduc la același rezultat.

2. Identificarea monoamelor centrale:

mon. max. / mintermi	max_1	max_2	max_3	max_4	max_5
m_7				*	*
m_{10}		⊛			
m_6				⊛	
m_{12}	⊛				
m_5					⊛
m_9			⊛		
m_3				*	*
m_1			*		*
m_2		*		*	

S-au încercuit steluțele unice pe liniile lor. Se observă că pe toate coloanele există steluțe încercuite, deci $M(f) = C(f)$.

3. Este **primul caz** al algoritmului de simplificare.
4. Soluția unică este:
$f'(x_1, x_2, x_3, x_4) = max_1 \vee max_2 \vee max_3 \vee max_4 \vee max_5 =$
$= x_1x_2\bar{x}_3\bar{x}_4 \vee \bar{x}_2x_3\bar{x}_4 \vee \bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1x_3 \vee \bar{x}_1x_4$

Exemplul 7.10.

Să se simplifice următoarea funcție de 4 variabile, dată prin forma canonică disjunctivă:

$f(x_1, x_2, x_3, x_4)= m_4 \vee m_6 \vee m_7 \vee m_8 \vee m_9 \vee m_{10} \vee m_{11} \vee m_{12}$

Suportul funcției, ordonat descrescător după numărul de valori 1, este:

$S_f=\{(0,1,1,1),(1,0,1,1),(0,1,1,0),(1,0,0,1),(1,0,1,0),(1,1,0,0),(0,1,0,0),(1,0,0,0)\}$

1. Factorizarea - obținerea monoamelor maximale:

Grupul		x_1	x_2	x_3	x_4	
	√	0	1	1	1	m_7
I	√	1	0	1	1	m_{11}
	√	0	1	1	0	m_6
	√	1	0	0	1	m_9
II	√	1	0	1	0	m_{10}
	√	1	1	0	0	m_{12}
III	√	0	1	0	0	m_4
	√	1	0	0	0	m_8
		0	1	1	–	$m_7\vee m_6= \bar{x}_1x_2x_3= max_1$
IV=I+II	√	1	0	–	1	$m_{11}\vee m_9$
	√	1	0	1	–	$m_{11}\vee m_{10}$

		0	1	–	0	$m_6 \vee m_4 = \bar{x}_1 x_2 \bar{x}_4 = max_2$
	√	1	0	0	–	$m_9 \vee m_8$
V=II+III	√	1	0	–	0	$m_{10} \vee m_8$
		–	1	0	0	$m_{12} \vee m_4 = x_2 \bar{x}_3 \bar{x}_4 = max_3$
		1	–	0	0	$m_{12} \vee m_8 = x_1 \bar{x}_3 \bar{x}_4 = max_4$
VI=IV+V		1	0	–	–	$m_{11} \vee m_9 \vee m_{10} \vee m_8 = x_1 \bar{x}_2 = max_5$

$M(f) = \{max_1, max_2, max_3, max_4, max_5\}$
$= \{\bar{x}_1 x_2 x_3, \bar{x}_1 x_2 \bar{x}_4, x_2 \bar{x}_3 \bar{x}_4, x_1 \bar{x}_3 \bar{x}_4, x_1 \bar{x}_2\}$

2. Identificarea monoamelor centrale:

mon. max. / mintermi	max_1	max_2	max_3	max_4	max_5
m_7	⊛				
m_{11}					⊛
m_6	*	*			
m_9					⊛
m_{10}					⊛
m_{12}			*	*	
m_4		*	*		
m_8				*	*

S-au încercuit steluțele unice pe linii, s-au hașurat coloanele monoamelor centrale și apoi liniile cu steluțe hașurate.

$C(f) = \{max_1, max_5\}$

3. $M(f) \neq C(f)$ și $C(f) \neq \varnothing$, este cazul al doilea al algoritmului de simplificare.

Se notează cu $g(x_1, x_2, x_3, x_4) \overset{not.}{=} max_1 \vee max_5 = \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2$.

4. Pentru a identifica soluția, trebuie să alegem funcții $h(x_1, x_2, x_3, x_4)$, cât mai simple, astfel încât să acopere mintermii m_{12} și m_4 aflați pe liniile nehașurate.
Se va alege monomul maximal pe a cărui coloană se află cele mai multe steluțe nehașurate, în cazul de față monomul $max_3 = x_2 \bar{x}_3 \bar{x}_4$.
Observăm că, pentru $h(x_1, x_2, x_3, x_4) = max_3$, funcția h acoperă $S_f - S_g$ și deci am identificat unica formă simplificată a funcției:

$f'(x_1, x_2, x_3, x_4) = g(x_1, x_2, x_3, x_4) \vee h(x_1, x_2, x_3, x_4) = \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 \vee x_2 \bar{x}_3 \bar{x}_4$

Observație:
Chiar dacă, în general, pentru cazurile al doilea și al treilea din algoritmul de simplificare se obțin mai multe forme simplificate ale funcției, există situații în care se obține o unică formă simplificată.

7.4. Metoda lui Moisil – metodă algebrică, bazată pe logica propozițiilor

Metoda de simplificare a lui Moisil utilizează logica propozițiilor pentru obținerea tuturor formelor simplificate ale funcției inițiale. Această metodă se bazează pe aplicarea uneia dintre metodele prezentate anterior pentru identificarea mulțimii monoamelor maximale, adică pentru pasul de factorizare.

Exemplul 7.11.
Se utilizează metoda lui Moisil pentru simplificarea funcției din Exemplul 7.8.

$f(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 x_3 \vee \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 \bar{x}_3 \vee x_1 x_2 x_3$
$= m_0 \vee m_1 \vee m_3 \vee m_4 \vee m_6 \vee m_7$

$M(f) = \{max_1 = \bar{x}_2 \bar{x}_3,\ max_2 = \bar{x}_1 \bar{x}_2,\ max_3 = x_1 \bar{x}_3,\ max_4 = \bar{x}_1 x_3,$
$max_5 = x_1 x_2,\ max_6 = x_2 x_3\}$

Tabelul care arată corespondența dintre mintermii funcției și monoamele maximale este:

mon. max. / mintermi	max_1	max_2	max_3	max_4	max_5	max_6
m_0	*	*				
m_4	*		*			
m_1		*		*		
m_6			*		*	
m_3				*		*
m_7					*	*

Se consideră propozițiile:

p_1: „max_1 face parte din forma simplificată a funcției"
p_2: „max_2 face parte din forma simplificată a funcției"
p_3: „max_3 face parte din forma simplificată a funcției"
p_4: „max_4 face parte din forma simplificată a funcției"
p_5: „max_5 face parte din forma simplificată a funcției"
p_6: „max_6 face parte din forma simplificată a funcției"

Din tabelul anterior, urmărind liniile, se obțin următoarele afirmații valide, ce vor fi traduse în formule propoziționale care sunt tautologii, astfel:

- Afirmația validă: „m_0 **este acoperit şi de** max_1 **şi de** max_2", deci „max_1, max_2 **sau ambele fac parte din forma simplificată a funcţiei**", se traduce în:

  $p_1 \vee p_2 \equiv T$

- Afirmația validă „m_4 **este acoperit şi de** max_1 **şi de** max_3", deci „max_1, max_3 **sau ambele fac parte din forma simplificată a funcţiei**", se traduce în:

  $p_1 \vee p_3 \equiv T$

- Afirmația validă „m_1 **este acoperit şi de** max_2 **şi de** max_4", deci „max_2, max_4 **sau ambele fac parte din forma simplificată a funcţiei**", se traduce în:

  $p_2 \vee p_4 \equiv T$

- Afirmația validă „m_6 **este acoperit şi de** max_3 **şi de** max_5", deci „max_3, max_5 **sau ambele fac parte din forma simplificată a funcţiei**", se traduce în:

  $p_3 \vee p_5 \equiv T$

- Afirmația validă „m_3 **este acoperit şi de** max_4 **şi de** max_6", deci „max_4, max_6 **sau ambele fac parte din forma simplificată a funcţiei**", se traduce în:

  $p_4 \vee p_6 \equiv T$

- Afirmația validă „m_7 **este acoperit şi de** max_5 **şi de** max_6", deci „max_5, max_6 **sau ambele fac parte din forma simplificată a funcţiei**", se traduce în:

  $p_5 \vee p_6 \equiv T$

Construim următoarea formulă propozițională care este tautologie:

$$T \equiv (p_1 \vee p_2) \wedge (p_1 \vee p_3) \wedge (p_2 \vee p_4) \wedge (p_3 \vee p_5) \wedge (p_4 \vee p_6) \wedge (p_5 \vee p_6)$$

şi reprezintă faptul că toţi mintermii din expresia inițială a funcţiei trebuie să fie acoperiţi de monoame maximale.

Această formulă se află în forma normală conjunctivă (FNC), trebuie să o aducem la forma normală disjunctivă (FND) şi să reținem cuburile cu număr minim de literali:

$$T \equiv (p_1 \vee p_2) \wedge (p_1 \vee p_3) \wedge (p_2 \vee p_4) \wedge (p_4 \vee p_6) \wedge (p_3 \vee p_5) \wedge (p_5 \vee p_6)$$

Se dă factor comun:

$$T \equiv (p_1 \vee (p_2 \wedge p_3)) \wedge (p_4 \vee (p_2 \wedge p_6)) \wedge (p_5 \vee (p_3 \wedge p_6))$$

Se aplică distributivitatea:

$$T \equiv (p_1 \wedge p_4 \wedge p_5) \vee (p_1 \wedge p_4 \wedge p_3 \wedge p_6) \vee (p_1 \wedge p_2 \wedge p_6 \wedge p_5) \vee$$
$$\vee (p_1 \wedge p_2 \wedge p_6 \wedge p_3 \wedge p_6) \vee (p_2 \wedge p_3 \wedge p_4 \wedge p_5) \vee (p_2 \wedge p_3 \wedge p_4 \wedge p_3 \wedge p_6) \vee$$
$$\vee (p_2 \wedge p_3 \wedge p_2 \wedge p_6 \wedge p_5) \vee (p_2 \wedge p_3 \wedge p_2 \wedge p_6 \wedge p_3 \wedge p_6)$$

Se aplică idempotența:

$$T \equiv (p_1 \wedge p_4 \wedge p_5) \vee (p_1 \wedge p_4 \wedge p_3 \wedge p_6) \vee (p_1 \wedge p_2 \wedge p_6 \wedge p_5) \vee$$
$$\vee (p_1 \wedge p_2 \wedge p_3 \wedge p_6) \vee (p_2 \wedge p_3 \wedge p_4 \wedge p_5) \vee (p_2 \wedge p_3 \wedge p_4 \wedge p_6) \vee$$
$$\vee (p_2 \wedge p_3 \wedge p_6 \wedge p_5) \vee (p_2 \wedge p_3 \wedge p_6)$$

O formulă propozițională în FND este tautologie dacă cel puțin un cub este tautologie. Considerând ca tautologii cele mai scurte cuburi, obținem:

$p_1 \wedge p_4 \wedge p_5 \equiv T$, soluția corespunzătoare este:

$$f'_1(x_1, x_2, x_3) = \overline{x_2}\,\overline{x_3} \vee \overline{x_1}x_3 \vee x_1x_2 = max_1 \vee max_4 \vee max_5$$

$p_2 \wedge p_3 \wedge p_6 \equiv T$, soluția corespunzătoare este:

$$f'_2(x_1, x_2, x_3) = \overline{x_1}\,\overline{x_2} \vee x_1\overline{x_3} \vee x_2x_3 = max_2 \vee max_3 \vee max_6$$

Astfel, s-au obținut cele două forme simplificate ale funcției date. soluții identice cu cele obținute prin metoda lui Quine.

7.5. Aplicație practică

O aplicație directă a simplificării funcțiilor booleene este simplificarea automată a unei expresii condiționale. Pentru început trebuie identificate expresiile condiționale simple, care sunt legate prin operatori logici. Acestea se notează cu variabile şi astfel se obține o expresie logică ce poate fi privită ca o funcție booleană. Procesul de simplificare este uşor de implementat conform paşilor descrişi în cele ce urmează.

Pas1: Expresia condițională se transformă din forma curentă în formă normală disjunctivă, utilizând algoritmul adecvat din primul capitol.

Pas2: Pentru a se obține forma canonică disjunctivă a funcției booleene ataşate, la fiecare cub se adaugă variabilele lipsă, atât simple, cât şi negate.

Pas3: Pentru factorizare se utilizează metoda lui Quine Mc'Clusky.

Pas4: Pentru identificarea formei simplificate, se utilizează metoda lui Moisil.

Exemplul 7.12.

Fie următoarea expresie condițională Pascal:

(($x < y$) *and* ($y <= z$) *and* ($z < 5$)) *or* (($x < y$) *and* ($y >= 5$) *and* ($z < 5$))

Expresia este echivalentă cu:

(($x < y$) *and* (($y <= z$) *and* ($z < 5$)) *and* ($z < 5$)) *or* (($x < y$) *and* ($y > = 5$) *and* ($z < 5$))

Observăm că din ($y <= z$) *and* ($z < 5$) se obține $y < 5$, adică $\neg$ ($y >= 5$). Deci, expresia devine:

(($x < y$) *and* (*not* ($y >= 5$)) *and* ($z < 5$)) *or* (($x < y$) *and* ($y >= 5$) *and* ($z < 5$))

Pentru a se putea urmări mai ușor procesul de simplificare, notăm expresiile condiționale cu variabile simple, după cum urmează:

- $x < y$ cu x_1,
- $y >= 5$ cu x_2
- $z < 5$ cu x_3.

Se obține următoarea funcție booleană:

$f(x_1, x_2, x_3) = (x_1 \wedge \bar{x}_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3)$

Forma simplificată a funcției f este: $f'(x_1, x_2, x_3) = x_1 \wedge x_3$

Astfel, expresia inițială s-a redus la expresia:

($x < y$) *and* ($z < 5$)

O aplicație care realizează în mod automat cele de mai sus a fost prezentată în cadrul conferinței ICMI2006, vezi [24].

7.6. Considerații privind simplificarea funcțiilor booleene

Toate cele trei metode de simplificare prezentate utilizează FCD a funcției ca formă inițială. Din punctul de vedere al simplificarii, scopul este acela de a se obține o formă echivalentă a funcției inițiale, cu cât mai puține apariții de variabile. Aceasta poate fi o formă conjunctivă sau una disjunctivă.

Prin negarea FCC a unei funcții se obține FCD a negației acestei funcției. Astfel, dacă se dorește simplificarea FCC a funcției f, se pot aplica metodele de mai sus pentru $\bar{f}$, care este în FCD.

Exemplul 7.13.

Fie funcția $f(x_1, x_2, x_3, x_4) = M_0M_4$ – formă canonică conjunctivă.

$$\bar{f}(x_1, x_2, x_3, x_4) = m_0 \vee m_4 = \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1x_2\bar{x}_3\bar{x}_4 = \bar{x}_1\bar{x}_3\bar{x}_4 (\bar{x}_2 \vee x_2) =$$
$$= \bar{x}_1\bar{x}_3\bar{x}_4 \text{ – formă simplificată}$$

Negând forma simplificată de mai sus, obținem forma simplificată a funcției inițiale f, adică: $f'(x_1, x_2, x_3, x_4) = x_1 \vee x_3 \vee x_4$

Uneori, forma simplificată a unei funcții booleene este o FCC și nu una FCD, așa cum se poate observa din exemplul următor. Se recomandă să se simplifice atât funcția dată, cât și negația sa, iar în final să se aleagă forma cea mai simplă.

Exemplul 7.14.

Fie funcția booleană de 4 variabile:

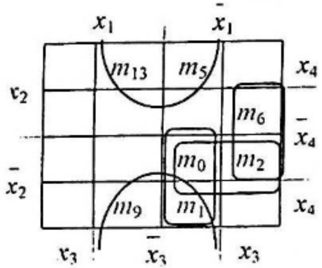
$$f(x_1, x_2, x_3, x_4) = x_1x_2\bar{x}_3x_4 \vee \bar{x}_1x_2\bar{x}_3x_4 \vee \bar{x}_1x_2x_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2x_3\bar{x}_4 \vee$$
$$\vee x_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3x_4 =$$
$$= m_{13} \vee m_5 \vee m_6 \vee m_0 \vee m_2 \vee m_9 \vee m_1 \text{ (FCD)}$$
$$= M_{15} \wedge M_7 \wedge M_{14} \wedge M_{12} \wedge M_4 \wedge M_{10} \wedge M_8 \wedge M_{11} \wedge M_3 \text{ (FCC)}$$

Negând FCC(f) se obține FCD($\bar{f}$).

$$\bar{f}(x_1, x_2, x_3, x_4) = m_{15} \vee m_7 \vee m_{14} \vee m_{12} \vee m_4 \vee m_{10} \vee m_8 \vee m_{11} \vee m_3$$

Vom simplifica prin metoda diagramei Veitch ambele funcții, f și $\bar{f}$

Simplificăm funcția $f(x_1, x_2, x_3, x_4)$:

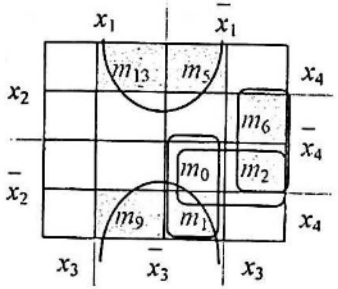


1 Conformul grupării mintermilor din diagramă, mulțimea monoamelor maximale este:

$M(f) = \{\bar{x}_3x_4, \bar{x}_1\bar{x}_2\bar{x}_3, \bar{x}_1\bar{x}_2\bar{x}_4, \bar{x}_1x_3\bar{x}_4\}$

$max_1 = \bar{x}_3 x_4 = m_{13} \vee m_5 \vee m_9 \vee m_1,$
$max_2 = \bar{x}_1\bar{x}_2\bar{x}_3 = m_1 \vee m_0$
$max_3 = \bar{x}_1\bar{x}_2\bar{x}_4 = m_0 \vee m_2$
$max_4 = \bar{x}_1 x_3 \bar{x}_4 = m_6 \vee m_2$

2. $C(f) = \{\ max_1 = \bar{x}_3 x_4,\ max_4 = \bar{x}_1 x_3 \bar{x}_4 \}$



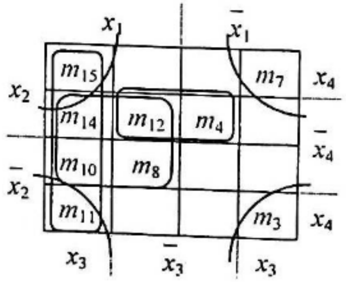
3. Este **cazul al doilea** al algoritmului de simplificare.
4. Singurul minterm neacoperit, m_0, poate fi acoperit în două moduri, la fel de simple, folosind max_2 sau max_3.

$f_1'(x_1, x_2, x_3, x_4) = \bar{x}_3 x_4 \vee \bar{x}_1 x_3 \bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3$

$f_2'(x_1, x_2, x_3, x_4) = \bar{x}_3 x_4 \vee \bar{x}_1 x_3 \bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_4$

Ambele forme disjunctive simplificate ale funcției conțin 8 apariții de variabile, 7 operatori binari și 6 operatori unari.

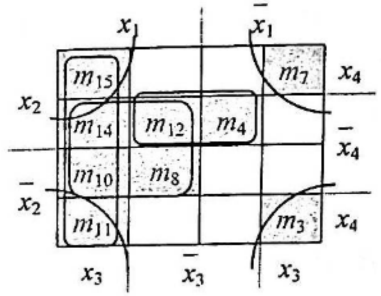
Continuăm cu simplificarea funcției:
$\bar{f}(x_1, x_2, x_3, x_4) = m_{15} \vee m_7 \vee m_{14} \vee m_{12} \vee m_4 \vee m_{10} \vee m_8 \vee m_{11} \vee m_3$



1. Se determină mulțimea monoamelor maximale.

$max'_1 = x_1 x_3 = m_{15} \vee m_{14} \vee m_{10} \vee m_{11}$
$max'_2 = x_1 \bar{x}_4 = m_{14} \vee m_{12} \vee m_{10} \vee m_8$
$max'_3 = x_2 \bar{x}_3 \bar{x}_4 = m_{12} \vee m_4$
$max'_4 = x_3 x_4 = m_{15} \vee m_{11} \vee m_7 \vee m_3$
$M(\bar{f}) = \{x_1 x_3,\ x_1 \bar{x}_4,\ x_2 \bar{x}_3 \bar{x}_4,\ x_3 x_4\}$

2. $C(\bar{f}) = \{\ max'_2 = x_1 \bar{x}_4,\ max'_3 = x_2 \bar{x}_3 \bar{x}_4,\ max'_4 = x_3 x_4 \}$



3. Este **cazul al doilea** al algoritmului de simplificare, dar, după cum se observă din diagramă, monoamele centrale acoperă în întregime mintermii expresiei funcției $\bar{f}$.
4. Astfel, există o soluție unică pentru forma simplificată a funcției $\bar{f}$, obținută ca disjuncție a tuturor monoamelor centrale:

$\bar{f}'(x_1, x_2, x_3, x_4) = x_1 \bar{x}_4 \vee x_2 \bar{x}_3 \bar{x}_4 \vee x_3 x_4$

Aplicând negația obținem:

$f'(x_1, x_2, x_3, x_4) = (\bar{x}_1 \vee x_4)(\bar{x}_2 \vee x_3 \vee x_4)(\bar{x}_3 \vee \bar{x}_4).$

Se observă că această formă simplificată a funcției inițiale conține doar 7 apariții de variabile, 6 operații binare și 4 operații unare, deci e mai simplă față de formele disjunctive simplificate ale funcției f obținute anterior.
Astfel, se va alege ca cea mai simplă formă a funcției f, forma conjunctivă simplificată a funcției inițiale, obținută ca negație a formei disjunctive simplificate a lui $\bar{f}$, adică:

$f'(x_1, x_2, x_3, x_4) = (\bar{x}_1 \vee x_4)(\bar{x}_2 \vee x_3 \vee x_4)(\bar{x}_3 \vee \bar{x}_4).$

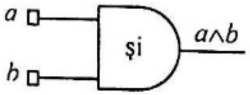
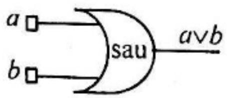
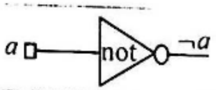
Simplificarea funcțiilor booleene este absolut necesară pentru elaborarea unor circuite logice simple, după cum se va vedea în capitolul următor.

8. CIRCUITE LOGICE

Calculatoarele, dar și alte aparate și echipamente electronice au în componența lor circuite electronice simple, ce se bazează pe cele două stări ale curentului electric: străbate sau nu o anumită porțiune din circuit. Modelarea unor astfel de circuite se face cu ajutorul funcțiilor booleene și a circuitelor logice care descriu algebric și grafic funcționarea acestora. Lucrările [4, 17, 18, 26, 36, 44] au fost utilizate pentru documentarea acestui capitol.

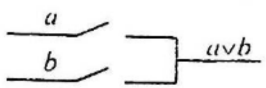
8.1. Noțiuni de bază

Elementul de bază în circuitele logice este circuitul *poartă*. Pentru o modelare ușoară a circuitelor sunt folosite simboluri speciale prin care se reprezintă cele trei operații logice fundamentale: „și", „sau", respectiv „not". *Porțile de bază*, corespunzătoare acestor operații fundamentale, sunt reprezentate conform standardelor IEEE astfel:

Reprezentarea grafică a circuitului	Funcția booleană asociată	Denumirea
	$a \wedge b$, $a \cap b$, $a * b$ sau ab	„și", conjuncția
	$a \vee b$, $a \cup b$ sau $a + b$	„sau", disjuncția
	$\neg a$ sau $\bar{a}$	„not", negația

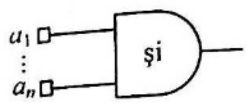
Circuitele electronice sunt formate dintr-o sursă și contacte. Corespunzător unei funcții booleene, există un circuit în care un contact este o variabilă ce poate lua valoarea 0 (circuit deschis) sau 1 (circuit închis), legat în serie: $\wedge$ sau în paralel: $\vee$. Fizic, porțile corespunzătoare celor două operații logice, sunt realizate astfel:

legare în serie

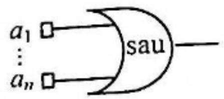


legare în paralel

Primele două porți, $\wedge$, respectiv $\vee$, pot avea mai multe intrări:



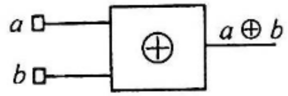
$a_1 \wedge a_2 \wedge \ldots \wedge a_n$, $a_1 \cap a_2 \cap \ldots \cap a_n$
$a_1 * a_2 * \ldots * a_n$ sau $a_1 a_2 \ldots a_n$



$a_1 \vee a_2 \vee \ldots \vee a_n$, $a_1 \cup a_2 \cup \ldots \cup a_n$ sau
$a_1 + a_2 + \ldots + a_n$

Observație:
Pentru simplificarea formei de scriere a expresiilor booleene, pe parcursul acestui capitol, operația de conjuncție se va reprezenta prin concatenarea variabilelor.

Exemplul 8.1.
Desenarea circuitului pentru operația „sau exclusiv", adică suma modulo 2.



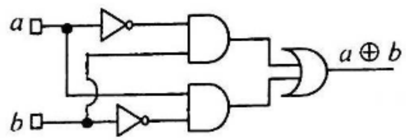
Primul pas este identificarea funcției booleene asociate.
$f: \{0,1\}^2 \to \{0,1\}, f(a,b) = a \oplus b$.

Pentru aceasta, vom construi mai întâi tabela de valori asociată funcției:

a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

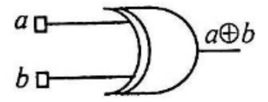
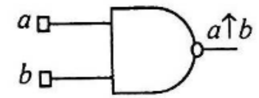
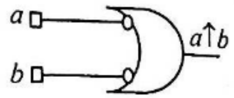
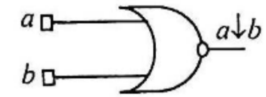
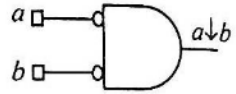
Forma canonică disjunctivă a funcției este:
$f(a,b) = \bar{a}b \vee a\bar{b}$.

Circuitul corespunzător este:



Definiția 8.1.
O ***poartă de bază*** este un minicircuit logic care realizează una dintre operațiile logice de bază: $\wedge$, $\vee$, $\bar{\ }$

Corespunzător operațiilor logice $\oplus$, $\uparrow$, $\downarrow$, des utilizate în modelarea circuitelor logice, s-au introdus următoarele ***porți derivate***:

Denumirea	Reprezentarea grafică a circuitului	Funcția booleană asociată	Simbol alternativ
„xor”		$a \oplus b = \bar{a}b \vee a\bar{b}$	
„nand”		$a \uparrow b = \overline{ab} = \bar{a} \vee \bar{b}$	
„nor”		$a \downarrow b = \overline{a \vee b} = \bar{a}\bar{b}$	

Porțile „și”, „sau”, „nand”, „nor” pot avea oricâte intrări, dar în mod practic, porțile de legătură au un număr limitat de intrări, de obicei două, trei sau patru, pentru a se adapta la standardul pachetelor de circuite integrate (IC).

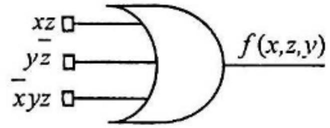
Din motive practice, de asamblare, un pachet de circuite integrate standard conține de la 14 la 16 „pini” (terminale metalice), dintre care o parte sunt porți de intrare, iar o altă parte sunt utilizate pentru conectarea la curent.

Modul de operare a unui circuit logic este descris de o funcție booleană. Circuitul logic corespunzător poate fi implementat direct din expresia funcției. Sunt prezentate în continuare câteva exemple de astfel de circuite.

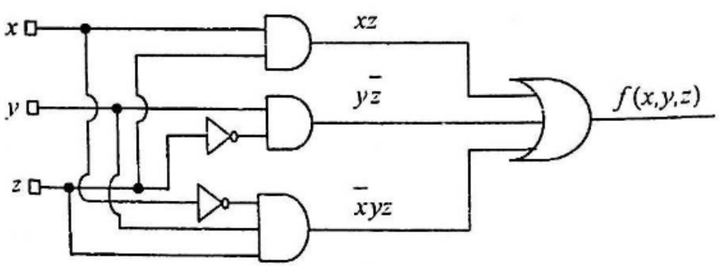
Exemplul 8.2.

$f(x, y, z) = xz \vee y\bar{z} \vee \bar{x}yz$

Această funcție booleană descrie funcționarea următorului circuit:



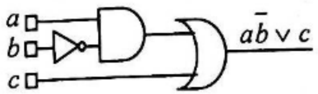
Detaliat:



În cele ce urmează se vor desena circuitele logice corespunzătoare unor funcții booleene simple.

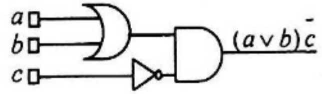
Exemplul 8.3.

$f(a,b,c) = a\bar{b} \vee c$



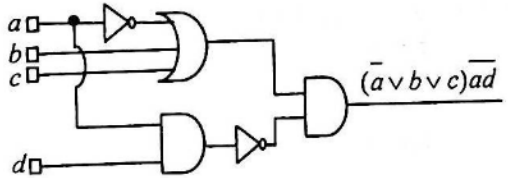
Exemplul 8.4.

$f(a,b,c) = (a \vee b)\bar{c}$



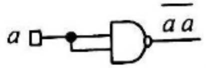
Exemplul 8.5.

$f(a,b,c,d) = (\bar{a} \vee b \vee c)\overline{ad}$



Exemplul 8.6.

$f(a) = \overline{aa}$. Circuitul logic este:



Din exemplele anterioare se observă că, pentru desenarea circuitelor booleene asociate, se ține cont de prioritatea operațiilor și de paranteze.

Exemplul 8.7.

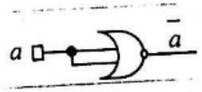
În funcție de operația „nor”, operațiile logice „not”, „și”, „sau” se exprimă astfel:

$\bar{a} = \overline{a \vee a} = a \downarrow a$

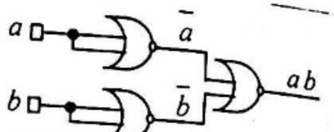
$ab = \overline{\bar{a} \vee \bar{b}} = (a \downarrow a) \downarrow (b \downarrow b)$

$a \vee b = \overline{\overline{a \vee b}} = \overline{a \downarrow b} = (a \downarrow b) \downarrow (a \downarrow b)$

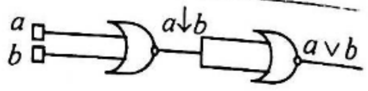
Circutele corespunzătoare sunt:

$\bar{a} = a \downarrow a$	

$ab = (a \downarrow a) \downarrow (b \downarrow b)$



$a \vee b = (a \downarrow b) \downarrow (a \downarrow b)$



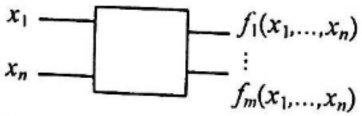
Similar se obțin circuitele corespunzătoare operațiilor logice de mai sus în funcție de operația „nand”

Observație:

Dacă un circuit logic are mai mult de o ieșire, atunci fiecare ieșire corespunde unei funcții booleene.

Definiția 8.2.

Un circuit logic cu m ieșiri se numește ***circuit combinațional*** și este simbolizat grafic astfel:



Observație:

Un circuit logic elementar este un caz particular al unui circuit combinațional (m=1).

8.2. Exemple utile

În partea hard a calculatoarelor se folosesc mai multe astfel de circuite logice combinaționale, cum sunt: *decodorul*, *codorul*, *circuitul comparator* și *detectorul de paritate*.

Exemplul 8.8.

Codorul este circuitul de codificare binară a cifrelor zecimale.
Circuitul are 10 intrări: variabilele $x_0, \ldots, x_9$ – corespunzătoare celor 10 cifre zecimale și 4 ieșiri: funcțiile booleene, f_1, f_2, f_3, f_4 – corespunzătoare celor patru cifre binare ale rezultatului.
Pentru codificarea în binar a cifrei zecimale „i”, variabilele de intrare sunt $x_i = 1$ și $x_j = 0$, $\forall j = \overline{0,9}, j \neq i$.

Primul pas pentru a construi circuitul este de a identifica expresiile funcțiilor booleene de ieșire. Vom utiliza tabelul următor:

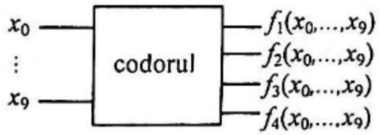
Cifră zecimală	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	f_1	f_2	f_3	f_4
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	0	1
2	0	0	1	0	0	0	0	0	0	0	0	0	1	0
3	0	0	0	1	0	0	0	0	0	0	0	0	1	1
4	0	0	0	0	1	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	1	0	0	0	0	0	1	0	1
6	0	0	0	0	0	0	1	0	0	0	0	1	1	0
7	0	0	0	0	0	0	0	1	0	0	0	1	1	1
8	0	0	0	0	0	0	0	0	1	0	1	0	0	0
9	0	0	0	0	0	0.	0	0	0	1	1	0	0	1

Al doilea pas este de a scrie forma disjunctivă, formă care este cel mai ușor de realizat în mod fizic:

$f_1(x_0,\ldots,x_9) = x_8 \vee x_9$ $\qquad$ $f_2(x_0,\ldots,x_9) = x_4 \vee x_5 \vee x_6 \vee x_7$
$f_3(x_0,\ldots,x_9) = x_2 \vee x_3 \vee x_6 \vee x_7$ $\qquad$ $f_4(x_0,\ldots,x_9) = x_1 \vee x_3 \vee x_5 \vee x_7 \vee x_9$

Pasul următor este cel de simplificare a funcțiilor booleene de ieșire. Funcțiile de mai sus sunt deja în forma simplificată

Ultimul pas este cel de desenare efectivă a circuitului



Mai sus s-a realizat doar o schemă a circuitului. Circuitul logic trebuie detaliat în interiorul dreptunghiului de mai sus. Deoarece expresiile funcțiilor sunt relativ simple, fiecare funcție fiind o disjuncție de variabile de intrare, lăsăm pentru cititor detalierea acestui circuit.

Exemplul 8.9.

Decodorul (inversul *codorului*) are ca intrare o secvență de patru cifre binare, reprezentată prin variabilele x_1, x_2, x_3, x_4 și generează cifra zecimală corespunzătoare. Pentru cele zece cifre zecimale, există zece funcții de ieșire cu semnificația următoare:

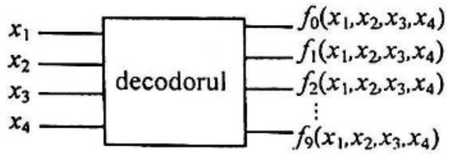
$f_i(x_1, x_2, x_3, x_4) = 1$ doar pentru $x_1x_2x_3x_4\,_{(2)} = i_{(10)}$, $i = \overline{0,9}$

- $f_0(0,0,0,0)=1$ deoarece $0000_{(2)} = 0_{(10)}$, pentru celelalte argumente valoarea funcției este 0;
- $f_7(0,1,1,1)=1$ deoarece $0111_{(2)} = 7_{(10)}$, pentru celelalte argumente valoarea funcției este 0;

x_1	x_2	x_3	x_4	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	forma canonică disjunctivă (cu un singur element)
0	0	0	0	1	0	0	0	0	0	0	0	0	0	$f_0(x_1,x_2,x_3,x_4) = \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$
0	0	0	1	0	1	0	0	0	0	0	0	0	0	$f_1(x_1,x_2,x_3,x_4) = \bar{x}_1\bar{x}_2\bar{x}_3x_4$
0	0	1	0	0	0	1	0	0	0	0	0	0	0	$f_2(x_1,x_2,x_3,x_4) = \bar{x}_1\bar{x}_2x_3\bar{x}_4$
0	0	1	1	0	0	0	1	0	0	0	0	0	0	$f_3(x_1,x_2,x_3,x_4) = \bar{x}_1\bar{x}_2x_3x_4$
0	1	0	0	0	0	0	0	1	0	0	0	0	0	$f_4(x_1,x_2,x_3,x_4) = \bar{x}_1x_2\bar{x}_3\bar{x}_4$
0	1	0	1	0	0	0	0	0	1	0	0	0	0	$f_5(x_1,x_2,x_3,x_4) = \bar{x}_1x_2\bar{x}_3x_4$
0	1	1	0	0	0	0	0	0	0	1	0	0	0	$f_6(x_1,x_2,x_3,x_4) = \bar{x}_1x_2x_3\bar{x}_4$
0	1	1	1	0	0	0	0	0	0	0	1	0	0	$f_7(x_1,x_2,x_3,x_4) = \bar{x}_1x_2x_3x_4$
1	0	0	0	0	0	0	0	0	0	0	0	1	0	$f_8(x_1,x_2,x_3,x_4) = x_1\bar{x}_2\bar{x}_3\bar{x}_4$
1	0	0	1	0	0	0	0	0	0	0	0	0	1	$f_9(x_1,x_2,x_3,x_4) = x_1\bar{x}_2\bar{x}_3x_4$

Se observă că funcțiile de ieșire sunt mintermi, deoarece acestea iau valoarea 1 pentru un singur argument.

Circuitul este de forma:



Exemplul 8.10.

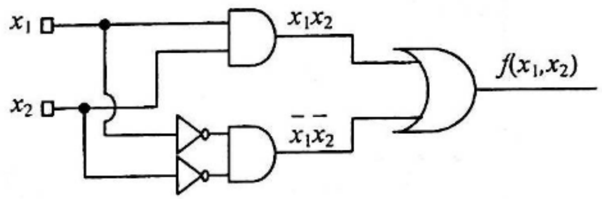
Circuitul comparator verifică dacă două cifre binare sunt sau nu identice.

Funcția booleană corespunzătoare operației de comparare a două cifre binare se furnizează tabelar în partea stângă.

x_1	x_2	$f(x_1,x_2)$
0	0	1
0	1	0
1	0	0
1	1	1

Expresia funcției în FCD este $f(x_1,x_2)= \bar{x}_1\bar{x}_2 \vee x_1x_2$.

Circuitul corespunzător este:

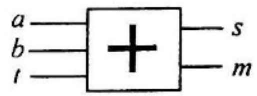


Exemplul 8.11.

Sumatorul binar calculează suma a două cifre binare: a și b de pe aceeași poziție dintr-un număr binar. Astfel, pe lângă cele două cifre, variabilă de intrare este și cifra de transport t de la adunarea cifrelor de rang imediat inferior, iar ca funcție de ieșire, pe lângă suma s, cifra m ce reprezintă transportul generat.

Funcțiile de ieșire sunt furnizate tabelar astfel:

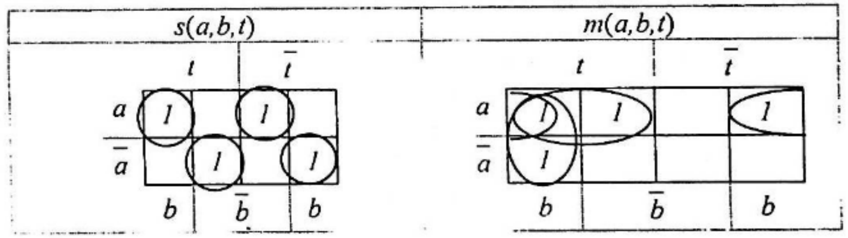
t	a	b	s	m
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Formele canonice disjunctive ale funcțiilor sunt:

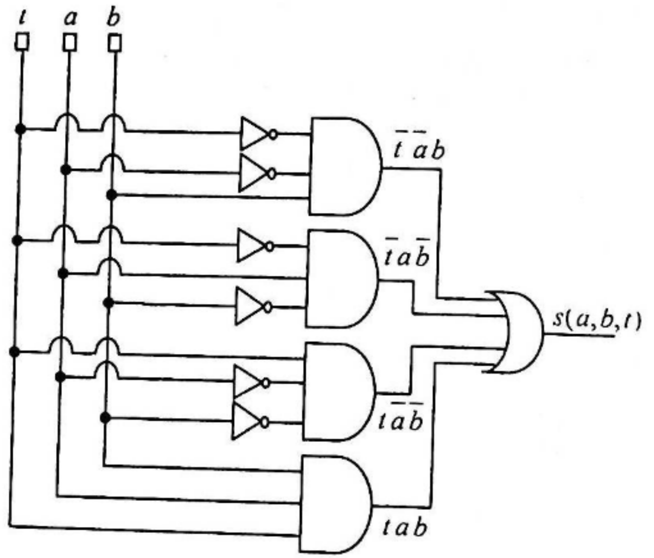
$$s(a,b,t) = \bar{t}\,\bar{a}b \vee \bar{t}a\bar{b} \vee t\bar{a}\bar{b} \vee tab$$
$$m(a,b,t) = \bar{t}\,ab \vee t\bar{a}b \vee ta\bar{b} \vee tab$$

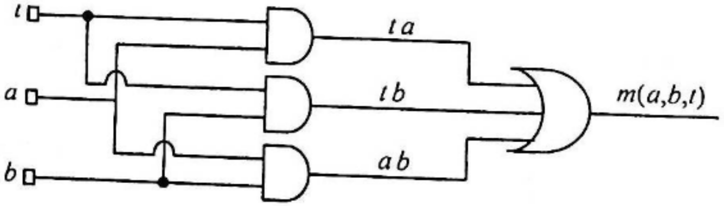
Înainte de a realiza cele două circuite, se încearcă obținerea unor forme mai simple ale funcțiilor s și m. Utilizăm metoda diagramelor Veitch .



Se observă că funcţia s nu se mai poate simplifica, deoarece toţi mintermii componenţi sunt monoame maximale. Funcţia m este în primul caz de simplificare, după simplificare devenind: $m(a,b,t) = ta \vee tb \vee ab$.

Se obţin următoarele circuite corespunzătoare funcţiilor $s(a,b,t)$ şi $m(a,b,t)$:





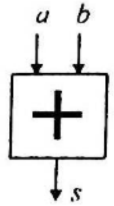
Exemplul 8.12. Sumatorul binar cu n poziţii

Fie două numere binare $a = a_{n-1}...a_{0(2)}$ şi $b = b_{n-1}...b_{0(2)}$, date prin şirurile cifrelor, unde cifra unităţilor este pe poziţia 0 şi cea mai semnificativă cifră se află pe poziţia n-1.

Se notează cu $s = s_{n-1}...s_{0(2)}$ suma celor două numere pe n poziţii.

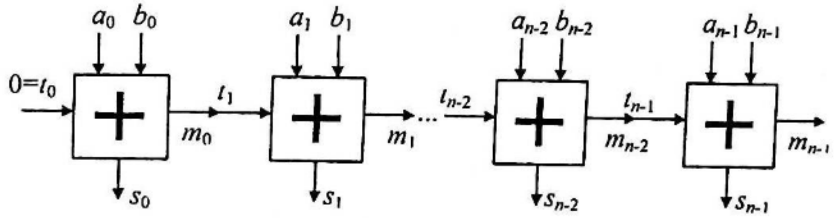
Ultima cifră de transport a sumei $a+b$, aflată pe poziţia n, este în realitate cifra cea mai semnificativă a sumei celor două numere binare. În practică, acest sumator este folosit pentru a aduna numere reprezentate în calculator pe acelaşi număr de biţi, iar suma are acelaşi număr de biţi ca ambii termeni. Astfel, în caz de depăşire, această cifră, cea mai semnificativă, se pierde.

Circuitul logic este simbolizat astfel:

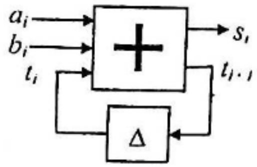


$a + b = s$

Circuitul detaliat este compus din mai multe sumatoare simple (Exemplul 8.11). Dacă notăm cifrele sumei cu s şi utilizăm indici pentru a face distincţie între cifrele de transport ale diferiţilor paşi, obţinem un circuit de forma:



Un astfel de circuit se numeşte circuit cu întârziere, deoarece cifra de transport obţinută la un pas se foloseşte în pasul următor, iar dacă am desena circuitul în mod compact, acesta ar fi:

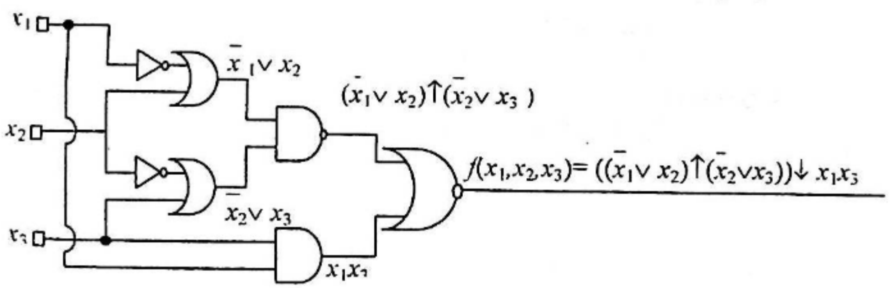


unde, partea Δ a circuitului îl va întârzia pe t.

Exemplul 8.13.

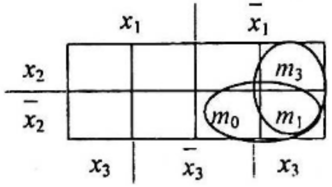
Fie funcţia booleană $f(x_1,x_2,x_3) = ((\bar{x}_1 \vee x_2) \uparrow (\bar{x}_2 \vee x_3)) \downarrow x_1x_3$.

Circuitul corespunzător conţine atât porţi de bază, cât şi porţi derivate



Se aduce funcția la forma canonică disjunctivă:

$$f(x_1,x_2,x_3) = ((\bar{x}_1 \vee x_2)\uparrow(\bar{x}_2 \vee x_3)) \downarrow x_1x_3 =$$
$$= \overline{\overline{(\bar{x}_1 \vee x_2)(\bar{x}_2 \vee x_3)} \vee x_1x_3} =$$
$$= (\bar{x}_1 \vee x_2)(\bar{x}_2 \vee x_3)(\bar{x}_1 \vee \bar{x}_3) = (\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_3 \vee x_2\bar{x}_2 \vee x_2x_3)(\bar{x}_1 \vee \bar{x}_3) =$$
$$= (\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_3 \vee 0 \vee x_2x_3)(\bar{x}_1 \vee \bar{x}_3) = (\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_3 \vee x_2x_3)(\bar{x}_1 \vee \bar{x}_3) =$$
$$= \bar{x}_1\bar{x}_2\bar{x}_1 \vee \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1x_3\bar{x}_1 \vee \bar{x}_1x_3\bar{x}_3 \vee x_2x_3\bar{x}_1 \vee x_2x_3\bar{x}_3 =$$
$$= \bar{x}_1\bar{x}_2 \vee \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1x_3 \vee 0 \vee \bar{x}_1x_2x_3 \vee 0 =$$
$$= \bar{x}_1\bar{x}_2x_3 \vee \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1x_2x_3 = m_1 \vee m_0 \vee m_3$$



Prin factorizare, folosind diagrama Veitch se obține:

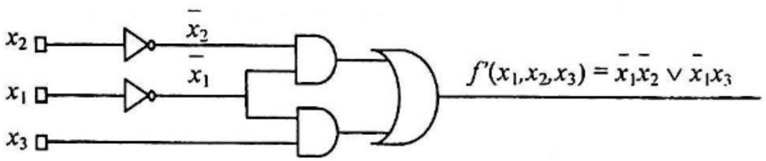
$max_1 = m_0 \vee m_1 = \bar{x}_1\bar{x}_2$;

$max_2 = m_1 \vee m_3 = \bar{x}_1x_3$

$M(f) = C(f) = \{max_1, max_2\}$

Forma simplificată a funcției este: $f'(x_1,x_2,x_3) = \bar{x}_1\bar{x}_2 \vee \bar{x}_1x_3$

Se redesenează circuitul în forma simplificată:

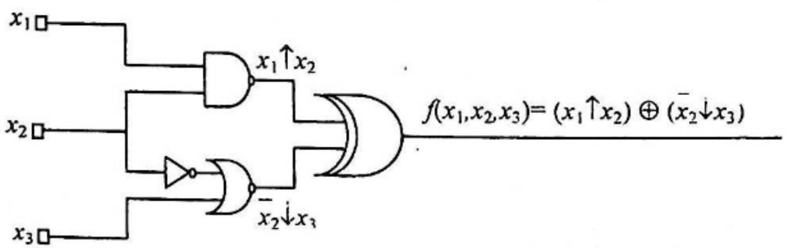


Exemplul 8.14.

Fie următoarea funcție booleană care conține operațiile $\oplus, \uparrow, \downarrow$:

$$f(x_1,x_2,x_3) = (x_1\uparrow x_2) \oplus (\bar{x}_2 \downarrow x_3).$$

Se construiește circuitul logic corespunzător folosind porțile derivate:



Se aduce funcția la o formă disjunctivă, nu neapărat forma canonică disjunctivă, utilizând definițiile operatorilor $\oplus,\uparrow,\downarrow$ și legile lui DeMorgan, ale distributivității, idempotenței, absorbției:

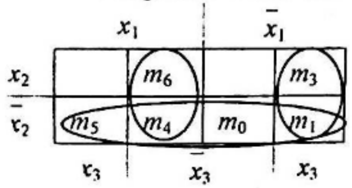
$$f(x_1,x_2,x_3) = (x_1 \uparrow x_2) \oplus (\bar{x}_2 \downarrow x_3) =$$
$$= \overline{(x_1 \uparrow x_2)}(\bar{x}_2 \downarrow x_3) \vee (x_1 \uparrow x_2)\overline{(\bar{x}_2 \downarrow x_3)} =$$
$$= (\overline{\overline{x_1x_2}})(\overline{\bar{x}_2 \vee x_3}) \vee (\overline{x_1x_2})(\overline{\overline{\bar{x}_2 \vee x_3}}) =$$
$$= x_1x_2x_2\bar{x}_3 \vee (\bar{x}_1 \vee \bar{x}_2)(\bar{x}_2 \vee x_3) = x_1x_2\bar{x}_3 \vee \bar{x}_2 \vee \bar{x}_1x_3$$

S-a obținut o formă destul de simplă, dar se încearcă simplificarea în continuare a acesteia.

Din această formă disjunctivă se poate completa diagrama Veitch astfel:

- pentru mintermul $x_1x_2\bar{x}_3 = m_6$ se completează căsuța corespunzătoare.
- pentru monomul $\bar{x}_2$ se marchează în diagramă cele patru căsuțe corespunzătoare mintermilor care au în comun variabila $\bar{x}_2$, adică m_1, m_0, m_5, m_4;
- pentru monomul $\bar{x}_1x_3$ care acoperă mintermii $\bar{x}_1\bar{x}_2x_3 = m_1$ și $\bar{x}_1x_2x_3 = m_3$ se marchează căsuțele corespunzătoare:

Diagrama Veitch este:



În urma factorizărilor se obțin monoamele maximale care sunt și monoame centrale:

$max_1 = \bar{x}_2 = m_1 \vee m_0 \vee m_5 \vee m_4$

$max_2 = x_1\bar{x}_3 = m_6 \vee m_4$

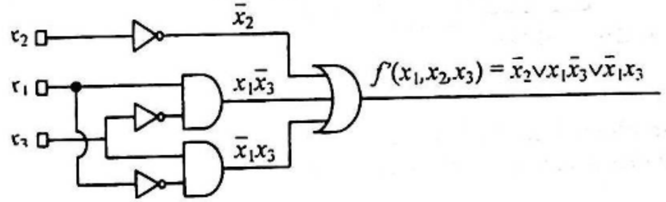
$max_3 = \bar{x}_1x_3 = m_1 \vee m_3$

$M(f) = C(f) = \{max_1, max_2, max_3\}$

Conform primului caz de simplificare există o unică formă simplificată a funcției:

$$f'(x_1,x_2,x_3) = \bar{x}_2 \vee x_1\bar{x}_3 \vee \bar{x}_1x_3$$

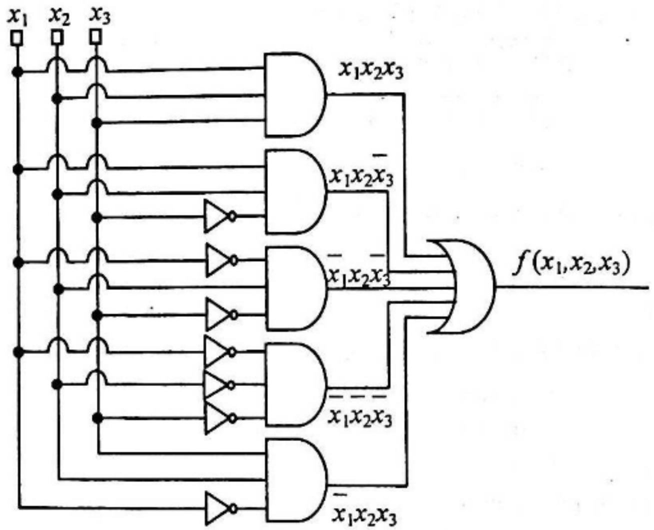
Circuitul în forma simplificată este următorul:

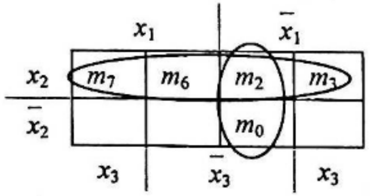


Exemplul 8.15.

Fie funcția $f(x_1,x_2,x_3) = x_1x_2x_3 \vee x_1x_2\bar{x}_3 \vee \bar{x}_1x_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1x_2x_3$.

Circuitul pentru această formă canonică disjunctivă este următorul:





Folosind diagrama Veitch se obțin:

$max_1 = m_2 \vee m_0 = \bar{x}_1\bar{x}_3$

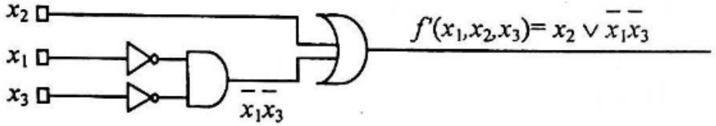
$max_2 = m_7 \vee m_6 \vee m_2 \vee m_3 = x_2$

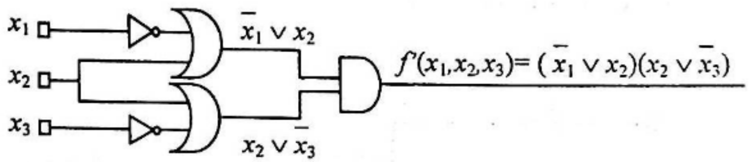
$M(f)=C(f)=\{\bar{x}_1\bar{x}_3,\ x_2\}$

$f' = x_2 \vee \bar{x}_1\bar{x}_3$ --- FCD

FCC(f')= $(x_2 \vee \bar{x}_1)(x_2 \vee \bar{x}_3)$.

Circuitele formei disjunctive, respectiv conjunctive simplificate sunt:





Concluzii:

Se observă că, pentru a se realiza circuite cât mai simple, funcțiile booleene corespunzătoare acestora trebuie simplificate.

9. PROBLEME PROPUSE

9.1. Probleme din logica propozițiilor

Problema 9.1.1.

Folosind metoda tabelelor de adevăr verificați:

1. asociativitatea conectivei „$\downarrow$": $p \downarrow (q \downarrow r) \equiv (p \downarrow q) \downarrow r$;
2. dacă se poate aplica o lege a lui De'Morgan pentru conectiva „$\uparrow$": $\neg(p \uparrow q) \equiv \neg p \downarrow \neg q$;
3. asociativitatea conectivei „$\uparrow$": $p \uparrow (q \uparrow r) \equiv (p \uparrow q) \uparrow r$;
4. dacă se poate aplica o lege a lui De'Morgan pentru conectiva „$\downarrow$": $\neg(p \downarrow q) \equiv \neg p \uparrow \neg q$;
5. dacă are loc proprietatea de absorbție: $p \uparrow (p \downarrow q) \equiv p$;
6. distributivitatea conectivei „$\uparrow$" față de conectiva „$\downarrow$": $p \uparrow (q \downarrow r) \equiv (p \uparrow q) \downarrow (p \uparrow r)$;
7. distributivitatea conectivei „$\downarrow$" față de conectiva „$\uparrow$": $p \downarrow (q \uparrow r) \equiv (p \downarrow q) \uparrow (p \downarrow r)$;
8. dacă are loc proprietatea de absorbție: $p \downarrow (p \uparrow q) \equiv p$.

Problema 9.1.2.

Utilizând metoda tabelelor de adevăr decideți tipul (consistentă, inconsistentă, tautologie, contingentă) formulei A:

1. $A = \neg p \vee \neg(q \wedge r) \rightarrow q \wedge \neg p$;
2. $A = p \vee \neg(q \wedge \neg r) \rightarrow p \wedge q \wedge \neg r$;
3. $A = \neg p \vee (q \wedge \neg r) \rightarrow p \wedge \neg q \wedge r$;
4. $A = \neg(\neg p \vee q) \vee r \rightarrow \neg p \vee \neg(q \wedge r)$;
5. $A = (p \vee q) \wedge \neg r \rightarrow p \wedge q \wedge r$;
6. $A = \neg p \vee (\neg q \vee \neg r) \rightarrow q \wedge \neg p$;
7. $A = p \rightarrow (q \wedge r) \vee q \wedge \neg p$;
8. $A = \neg p \vee (\neg q \wedge \neg r) \rightarrow q \wedge \neg p \wedge r$.

Dacă formula A este contingentă, scrieți toate modelele lui A, sau anti-modelele sale, dacă sunt mai puține decât modelele.

Problema 9.1.3.

Demonstrați că au loc următoarele relații de consecință logică

1. $p \to q \models (q \to r) \to (p \to r)$;
2. $p \to q \models (r \to p) \to (r \to q)$;
3. $p \to q \models (p \to r) \to (p \to q \wedge r)$.
4. $p \to q \models (q \to r) \to (p \to q \wedge r)$;
5. $p \to q \models (q \to r) \to (p \to q \vee r)$;
6. $p \to r \models (q \to r) \to ((p \vee q) \to r)$;
7. $p \to (q \to r) \models (p \to q) \to (p \to r)$.
8. $p \to q \models (\neg p \to q) \to q$;

folosind metoda tabelelor de adevăr

Problema 9.1.4.

Demonstrați că formulele următoare sunt tautologii utilizând metoda tabelelor de adevăr.

1. legea permutării premiselor: $(p \to (q \to r)) \to (q \to (p \to r))$.
2. semidistributivitatea la stânga a conectivei „$\to$" față de „$\wedge$" $(p \to (q \wedge r)) \to ((p \to q) \wedge (p \to r))$;
3. semidistributivitatea la stânga a conectivei „$\to$" față de „$\vee$" $(p \to (q \vee r)) \to ((p \to q) \vee (q \to r))$;
4. regula silogismului: $(p \to q) \wedge (q \to r) \to (p \to r)$;
5. semidistributivitatea la stânga a conectivei „$\vee$" față de „$\to$" $p \vee (q \to r) \to ((p \vee q) \to (p \vee r))$;
6. legea separării premiselor: $(p \wedge q \to r) \to (p \to (q \to r))$.
7. legea reunirii premiselor: $(p \to (q \to r)) \to (p \wedge q \to r)$.
8. teorema de tăiere: $(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$

Problema 9.1.5.

Demonstrați că mulțimile următoare sunt *mulțimi adecvate de conective*. adică toate celelalte conective propoziționale pot fi exprimate folosind elementele unei astfel de mulțimi (vezi subcapitolul 1.3. definirea conectivelor).

1. $\{\neg, \wedge\}$; 2. $\{\neg, \vee\}$; 3. $\{\neg, \to\}$; 4. $\{\uparrow\}$;
5. $\{\downarrow\}$; 6. $\{\oplus, \wedge\}$; 7. $\{\oplus, \vee\}$; 8. $\{\oplus, \to\}$.

Problema 9.1.6.

Aduceți la FNC (forma normală conjunctivă) și FND (forma normală disjunctivă)

1. legea separării premiselor: $(U \wedge V \to Z) \to (U \to (V \to Z))$.
2. semidistributivitatea la stânga a disjuncției față de implicație $U \vee (V \to Z) \to ((U \vee V) \to (U \vee Z))$;
3. legea permutării premiselor: $(U \to (V \to Z)) \to (V \to (U \to Z))$.
4. semidistributivitatea la stânga a implicației față de conjuncție: $(U \to (V \wedge Z)) \to ((U \wedge V) \to (U \wedge Z))$;
5. regula silogismului: $(U \to V) \wedge (V \to Z) \to (U \to Z)$.
6. axioma a doua a calculului propozițional: A2$= (U \to (V \to Z)) \to ((U \to V) \to (U \to Z))$;
7. teorema de tăiere: $(U \to V) \wedge (U \wedge V \to Z) \to (U \to Z)$.
8. legea reunirii premiselor: $(U \to (V \to Z)) \to (U \wedge V \to Z)$

Folosind una din aceste forme normale, demonstrați că această lege sau proprietate este o formulă validă în calculul propozițional

Problema 9.1.7.

Utilizând forma normală adecvată (FNC sau FND) demonstrați

1. $\models (U \to (V \to Z)) \to (U \wedge V \to Z)$;
2. $\models (U \to V) \to ((V \to Z) \to (U \to V \wedge Z))$;
3. $\models (U \to (V \wedge Z)) \to ((U \to V) \wedge (U \to Z))$;
4. $\models (U \to V) \to ((V \to Z) \to (U \to V \vee Z))$;
5. $\models (U \to V) \to ((Z \to U) \to (Z \wedge U \to V))$;
6. $\models (U \to V) \to ((U \to Z) \to (U \to (V \wedge Z))$.
7. $\models (U \to Z) \to ((V \to Z) \to (U \vee V \to Z))$;
8. $\models (U \to (V \vee Z)) \to ((U \to V) \vee (U \to Z))$

Problema 9.1.8.

Utilizând forma normală adecvată scrieți toate modelele formulelor

1. $(p \vee q \to r) \to (p \to r) \wedge q$;
2. $\neg(\neg p \vee q) \vee r \to \neg p \wedge \neg(q \wedge r)$.
3. $(p \wedge q \to r) \to (p \to r) \wedge q$.
4. $(p \vee q) \wedge \neg r \to p \wedge q \wedge r$.
5. $p \vee \neg(q \wedge \neg r) \to p \wedge q \wedge \neg r$.
6. $(p \vee q \to r) \to (q \to r) \wedge p$.
7. $(q \vee r \to p) \to (p \to r) \wedge q$.
8. $(q \wedge r \to p) \to (p \to r) \wedge q$

Problema 9.1.9.

Demonstrați că următoarele formule sunt inconsistente folosind forma normală adecvată:

1. $(U \to (V \to Z)) \wedge \neg((U \to V) \to (U \to Z))$;
2. $(\neg U \vee V) \wedge \neg(\neg V \to \neg U)$;
3. $(U \to V) \wedge (U \wedge V \to Z) \wedge (U \wedge \neg Z)$;
4. $(U \to (V \vee Z)) \wedge (\neg(U \to V) \wedge \neg(U \to Z))$;
5. $U \wedge (V \to Z) \wedge ((U \wedge V) \wedge \neg(U \wedge Z))$;
6. $(U \to (V \to Z)) \wedge (U \wedge V \wedge \neg Z)$;
7. $(U \to (V \to Z)) \wedge \neg(V \to (U \to Z))$;
8. $(U \wedge V \to Z) \wedge \neg(U \to (V \to Z))$;

Problema 9.1.10.

Să se demonstreze următoarele proprietăți ale relației de consecință logică, unde R, S $\subseteq F_P$ și U,V,Z $\in F_P$.

1. *monotonia*: dacă R $\models$U și R $\subseteq$ S atunci S $\models$U;
2. *eliminarea*: dacă S $\models V_i$, $\forall i \in I$ și S$\cup\{V_i | i \in I\} \models$ U atunci S $\models$U;
3. *tranzitivitatea*: dacă S $\models$U și {U}$\models$V atunci S $\models$V;
4. *conjuncția în concluzii* (*"și" la dreapta*):
   dacă S $\models$ U și S $\models$ V atunci S $\models$ U$\wedge$V;
5. *disjuncția în premise* (*"sau" la stânga*):
   dacă S$\cup$\{U\} $\models$ Z și S$\cup$\{V\} $\models$ Z atunci S$\cup$\{U$\vee$V\}$\models$Z;
6. *demonstrația pe cazuri:*
   dacă S$\cup$\{U\} $\models$ V și S$\cup$\{$\neg$U \} $\models$ V atunci S $\models$ V.

Problema 9.1.11.

Folosind definiția deducției să se demonstreze că au loc deducțiile:

1. $p \to q, r \to p, r \vdash q$;
2. $p \to r, p \vee r \to q, r \vdash q$;
3. $p \to q, q \to r, p \vdash r$;
4. $p \vee (q \to r), p \vee q, \neg p \vdash r$;
5. $p \to (q \to r), q, p \vdash r$;
6. $p \to (q \to r), p \wedge q, p \vdash r$;
7. $p \to (q \to r), p \to q, p \vdash r$;
8. $p \to q, r \to t, p \vee r, \neg q \vdash t$.

Problema 9.1.12.

Folosind teorema de deducție și inversa sa demonstrați:

1. $\vdash (p \to q) \to ((r \to t) \to (p \wedge r \to q))$;
2. $\vdash (p \to r) \to ((q \to r) \to (p \vee q \to r))$;
3. $\vdash (p \to q) \to ((r \to t) \to (p \wedge r \to t))$;
4. $\vdash (p \to r) \to ((p \wedge r \to q) \to (p \to q))$;
5. $\vdash (p \to q) \to ((r \to t) \to (p \wedge r \to q \wedge t))$;
6. $\vdash (p \to q) \to ((p \to r) \to (p \to q \wedge r))$;
7. $\vdash (p \to (q \to r)) \to (q \to (\neg r \to \neg p))$;
8. $\vdash (p \to q) \wedge (p \wedge q \to r) \to (p \to r)$.

Problema 9.1.13.

Folosind teorema de deducție și inversa sa demonstrați:

1. legea reunirii premiselor: $\vdash (U \to (V \to Z)) \to (U \wedge V \to Z)$;
2. semidistributivitatea la stânga a conectivei „$\to$" față de „$\vee$":
   $\vdash (U \to (V \vee Z)) \to ((U \to V) \vee (U \to Z))$;
3. legea separării premiselor: $\vdash (U \wedge V \to Z) \to (U \to (V \to Z))$;
4. semidistributivitatea la stânga a conectivei „$\wedge$" față de „$\to$":
   $\vdash U \wedge (V \to Z) \to ((U \wedge V) \to (U \wedge Z))$;
5. a doua axiomă a logicii propoziționale:
   $\vdash (U \to (V \to Z)) \to ((U \to V) \to (U \to Z))$;
6. legea permutării premiselor: $\vdash (U \to (V \to Z)) \to (V \to (U \to Z))$;
7. semidistributivitatea la stânga a conectivei „$\vee$" față de „$\to$":
   $\vdash U \vee (V \to Z) \to ((U \vee V) \to (U \vee Z))$;
8. $\vdash (U \to (V \to Z)) \to (U \to (\neg Z \to \neg V))$.

Problema 9.1.14.

Folosind metoda tabelelor semantice (construcția arborelui binar și/sau funcția TP) decideți tipul formulei A. Dacă A este consistentă, scrieți toate modelele sale:

1. $A = (p \wedge q) \vee (\neg p \wedge \neg r) \to (q \leftrightarrow r)$;
2. $A = (p \vee q \to r) \to (p \vee r \to q)$;
3. $A = (p \wedge q \to r) \to (p \to r) \wedge q$;
4. $A = (q \vee r \to p) \to (p \to r) \wedge q$;
5. $A = (r \vee q) \vee (p \to \neg r) \to (p \leftrightarrow q)$;
6. $A = (r \wedge q) \vee (\neg p \vee \neg r) \to (p \leftrightarrow q)$;

7. $A = (q \wedge r \to p) \to (p \to r) \wedge q$;
8. $A = (p \wedge r) \vee (\neg p \wedge \neg r) \to (q \leftrightarrow r)$

Problema 9.1.15.

Demonstrați că formula A este tautologie folosind metoda tabelelor semantice (construcția arborelui binar și/sau funcția TP):

1. legea permutării premiselor: $A = (p \to (q \to r)) \to (q \to (p \to r))$.
2. legea separării premiselor: $A = (p \wedge q \to r) \to (p \to (q \to r))$;
3. distributivitatea la stânga a conectivei „$\to$” față de „$\vee$”: $A = (p \to q \vee r) \leftrightarrow (p \to q) \vee (p \to r)$;
4. teorema de tăiere: $A = (p \to q) \wedge (p \wedge q \to r) \to (p \to r)$.
5. legea reunirii premiselor: $A = (p \to (q \to r)) \to (p \wedge q \to r)$;
6. $A = (p \to q) \to ((r \to t) \to (p \wedge r \to q \wedge t))$.
7. distributivitatea la stânga a conectivei „$\to$” față de „$\wedge$” $A = (p \to q \wedge r) \leftrightarrow (p \to q) \wedge (p \to r)$;
8. $A = (\neg (p \to r) \to \neg p) \to ((p \to q) \to (p \to r))$

Problema 9.1.16.

Folosind metoda tabelelor semantice (construcția arborelui binar și/sau funcția TP) demonstrați că au loc relațiile de consecință logică:

1. $p \to q \vee r \models (p \to q) \vee (p \to r)$.
2. $p \to (q \to r), r \vee \neg q \models (p \to \neg q) \vee r$.
3. $\neg (p \to r) \to \neg p \models (p \to q) \to (p \to r)$
4. $p \to (\neg q \vee \neg r \wedge s), p, r \models \neg q$.
5. $p \to q \models (r \to t) \to (p \wedge r \to q \wedge t)$
6. $p \to q, r \to t, p \wedge r \models q \wedge t$;
7. $p \to (q \to r) \models q \to (p \to r)$.
8. $p \wedge (q \to r), q \vee r \models p \to (q \to r)$

Problema 9.1.17.

Folosind calculul secvențelor verificați dacă secventele următoare sunt adevărate:

1. $p \wedge q \vee \neg p \wedge \neg r \Rightarrow q \to r$.
2. $q \vee r \to p, q \Rightarrow (p \to r) \wedge q$.
3. $p \wedge q \to r \Rightarrow (p \to r) \wedge q$;
4. $p, (r \vee q) \wedge (p \to \neg r) \Rightarrow (p \to q) \wedge (q \to p)$.
5. $p, p \vee q \to r \Rightarrow (p \vee r) \to q$.
6. $p, r \wedge q \vee \neg p \wedge \neg r \Rightarrow (p \to q) \wedge (q \to p)$;
7. $q \wedge r \to p, q \Rightarrow (p \to r) \wedge q$;
8. $p \vee r, \neg p \vee \neg q \Rightarrow q \to r$.

Problema 9.1.18.

Utilizând comparativ calculul secvențelor și calculul anti-secvențelor verificați validitatea formulei A:

1. distributivitatea la stânga a conectivei „$\to$” față de „$\wedge$”: $A = (p \to q \wedge r) \leftrightarrow (p \to q) \wedge (p \to r)$;
2. teorema de tăiere: $A = (p \to q) \wedge (p \wedge q \to r) \to (p \to r)$;
3. legea reunirii și a separării premiselor: $A = (p \to (q \to r)) \leftrightarrow (p \wedge q \to r)$;
4. semidistributivitatea la stânga a conectivei „$\wedge$” față de „$\to$”: $A = p \wedge (q \to r) \to (p \wedge q \to p \wedge r)$;
5. legea permutării premiselor: $A = (p \to (q \to r)) \leftrightarrow (q \to (p \to r))$;
6. distributivitatea la stânga a conectivei „$\to$” față de „$\vee$”: $A = (p \to q \vee r) \leftrightarrow (p \to q) \vee (p \to r)$;
7. distributivitatea la stânga a conectivei „$\vee$” față de „$\to$”: $A = p \vee (q \to r) \leftrightarrow (p \vee q) \to (p \vee r)$;
8. axioma A3 a calculului propozițional: $A = (p \to q) \leftrightarrow (\neg q \to \neg p)$.

Problema 9.1.19.

Folosind calculul secvențelor verificați dacă au loc relațiile:

1. $\neg p \to (\neg q \to r) \models (\neg p \to q) \vee r$;
2. $p \vee (q \to r), q \wedge r \models p \to (q \to r)$;
3. $p \wedge (q \to r), q \vee r \models p \to (q \to r)$;
4. $p \to r, q \to t, p \wedge q \models r \wedge t$;
5. $p \to (q \vee r \wedge s), p, \neg r \models q \vee r$;
6. $p \to (\neg q \vee r \wedge s), p, \neg s \models \neg q \vee s$;
7. $p \to q, r \to t, p \wedge r \models q \wedge t$;
8. $p \to (q \vee r \wedge s), p, \neg r \models q \wedge p$.

Problema 9.1.20.

Demonstrați nevaliditatea formulelor următoare folosind calculul anti-secvențelor. Câte anti-modele are fiecare formulă?

1. $(p \wedge q \to r) \to (p \to (\neg q \to \neg r))$;
2. $(p \to (q \to r)) \wedge \neg (p \to (q \to r))$;
3. $(p \to r) \vee (q \to r) \to (p \vee q \to r)$;

4. $\neg((p \to r) \to ((\neg p \to q) \to (\neg q \to r)))$;
5. $(p \land q \to p \land r) \to p \land (q \to r)$;
6. $(p \to (q \to r)) \land \neg((p \to q) \to (p \to r))$;
7. $\neg p \lor q \to (\neg q \to \neg p \land r)$;
8. $(p \land r \to p \land q) \to (p \to q) \land r$.

Problema 9.1.21.

Utilizând calculul anti-secvențelor arătați că nu au loc deducțiile următoare. Furnizați câte un anti-model.

1. $p \land r \lor \neg p \land \neg r \vdash q \leftrightarrow r$;
2. $q \land r \to p \vdash (p \to r) \land q$;
3. $r \land q \lor \neg p \lor \neg r \vdash p \leftrightarrow q$;
4. $r \lor q \lor (p \to \neg r) \vdash p \leftrightarrow q$;
5. $q \lor r \to p \vdash (p \to r) \land q$;
6. $p \land q \to r \vdash (p \to r) \land q$;
7. $p \lor q \to r \vdash (p \lor r) \to q$;
8. $p \land q \lor \neg p \land \neg r \vdash q \leftrightarrow r$.

Problema 9.1.22.

Folosind rezoluția generală demonstrați că formulele următoare sunt tautologii:

1. $(\neg B \to \neg A) \to ((\neg B \to A) \to B)$;
2. $(B \to A) \land (C \to A) \to (B \land C \to A)$;
3. $(B \to A) \land (C \to A) \to (B \lor C \to A)$;
4. $(A \to C) \to ((\neg A \to B) \to (\neg B \to C))$;
5. $A \lor (B \to C) \to (A \lor B) \to (A \lor C)$;
6. $(A \to B) \to ((C \to A) \to (C \to B))$;
7. $(A \to B) \to ((\neg A \to C) \to (\neg B \to C))$;
8. $(A \to B \land C) \to (A \to B) \land (A \to C)$.

Problema 9.1.23.

Demonstrați inconsistența următoarelor mulțimi de clauze folosind rezoluția blocării. Alegeți două indexări diferite pentru literalii din clauze.

1. $\{p \lor q, p \lor \neg q \lor r, p \lor \neg q \lor \neg r, \neg p \lor r, \neg p \lor \neg r\}$;
2. $\{\neg p \lor \neg q, \neg p \lor q \lor \neg r, p \lor \neg r, \neg p \lor r, p \lor r\}$;
3. $\{p \lor q, p \lor \neg q \lor \neg r, \neg p \lor \neg r, r, \neg p \lor r\}$;
4. $\{p \lor q, \neg p \lor q \lor \neg r, \neg p \lor q \lor r, \neg q \lor \neg r, \neg q \lor r\}$;
5. $\{p \lor \neg q, \neg p \lor \neg q \lor r, \neg p \lor q \lor r, p \lor q, \neg r\}$;
6. $\{p \lor q, \neg p \lor q \lor \neg r, \neg p \lor \neg q \lor \neg r, p \lor \neg q, r\}$;
7. $\{p \lor \neg q, \neg p \lor \neg q \lor r, \neg p \lor \neg q \lor \neg r, r \lor q, \neg r \lor q\}$;
8. $\{p \lor r, p \lor q \lor \neg r, \neg p \lor \neg q \lor r, \neg p \lor q \lor r, \neg r\}$.

Problema 9.1.24.

Construiți o respingere liniară din mulțimea de clauze S. Există respingeri input și unit din S?

1. $S = \{p \lor q \lor r, \neg q \lor r, \neg r, \neg p \lor r\}$;
2. $S = \{p \lor \neg r, q \lor r, \neg q \lor r, \neg p \lor \neg r\}$;
3. $S = \{q \lor r, \neg p, \neg q \lor r, p \lor \neg r\}$;
4. $S = \{\neg p \lor q, p \lor \neg q \lor r, \neg r, p \lor q \lor r, \neg p \lor \neg q\}$;
5. $S = \{p \lor r, \neg q, p \lor q \lor \neg r, \neg p \lor \neg r, q \lor r\}$;
6. $S = \{p \lor q, \neg p \lor q, \neg p \lor \neg q, p \lor \neg q\}$;
7. $S = \{p, q \lor r, \neg p \lor q \lor \neg r, \neg p \lor \neg q\}$;
8. $S = \{p \lor \neg q \lor r, q, \neg p \lor \neg q \lor r, \neg p \lor \neg q \lor \neg r, p \lor \neg r\}$.

Problema 9.1.25.

Utilizând strategia mulțimii suport demonstrați că au loc următoarele deducții:

1. $\neg(p \lor q) \to r, \neg p \lor q \lor r, \neg r \vdash q \land \neg r$;
2. $p \lor \neg r, \neg q \to r, \neg q \vdash \neg(p \to q)$;
3. $q \land r \to p, p \lor q, q \to r \vdash p$;
4. $r \to p \lor q, \neg p \to r, \neg q \vdash p \land \neg q$;
5. $\neg p \to q, (q \to r) \land \neg r \vdash p \land \neg r$;
6. $q \to p, q \lor r, p \to r \vdash r$;
7. $\neg p \to q \lor r, \neg q, p \to q \vdash \neg(p \lor q) \land r$;
8. $r \to p, \neg p, q \to p \lor r \vdash \neg(\neg p \to q \lor r)$.

Problema 9.1.26.

Demonstrați legea silogismului: $(p \to q) \to ((q \to r) \to (p \to r))$ utilizând:

1. o metodă sintactică;
2. o metodă semantică;
3. o metodă directă;
4. o metodă prin respingere;
5. o metodă semantică și directă;
6. o metodă sintactică și directă;
7. o metodă semantică și prin respingere;
8. o metodă sintactică și prin respingere.

9.2. Probleme din logica predicatelor

Problema 9.2.1.

Folosind definiția deducției să se demonstreze că au loc deducțiile:

1. $(\forall y)(\forall z)(p(y) \wedge q(z)) \vdash (\forall x)(p(x) \wedge q(x))$;
2. $(\forall y)(p(y) \vee q(y)), \neg(\forall z)p(z) \vdash (\exists x)q(x)$;
3. $(\exists y)p(y) \to (\forall z)q(z) \vdash (\forall x)(p(x) \to q(x))$;
4. $(\forall y)(p(y) \to q(y)), (\forall z)p(z) \vdash (\forall x)q(x)$;
5. $(\forall y)p(y) \vdash (\forall x)(q(x) \to p(x))$;
6. $(\forall y)p(y) \vdash (\forall x)(q(x) \vee p(x))$;
7. $(\forall y)p(y) \vdash (\exists x)p(x)$;
8. $(\forall y)(\forall z)(p(y) \to q(z)) \vdash (\forall x)(p(x) \to q(x))$.

Problema 9.2.2.

Transformați din limbaj natural în formule predicative afirmațiile următoare, alegând corespunzător constantele, simbolurile de funcții și simbolurile de predicate.

1. Pentru orice număr întreg pozitiv *x*, dacă *x* este pătrat perfect atunci există un întreg y astfel încât (*y*+1)*(*y*-1)=*x*-1.
2. Pentru orice număr întreg pozitiv *x*, dacă *x* nu este număr prim atunci există un număr prim *y* astfel încât *y* divide pe *x* si *y* este mai mic decât *x*.
3. Suma a două numere pare este un număr par.
4. Oricare ar fi două cercuri, dacă distanța dintre centrele lor este mai mică decât suma razelor lor, atunci cele două cercuri se intersectează.
5. Orice punct de pe bisectoarea interioară a unui unghi este egal depărtat de laturile unghiului.
6. Orice punct de pe mediatoarea unui segment este egal depărtat de capetele segmentului.
7. Ușile oricărui lift ar trebui sa se deschidă doar dacă liftul este oprit exact la un etaj.
8. Oricare două persoane posesoare de telefon mobil pot conversa dacă nu sunt suficient de apropiate una de cealalta, dacă ambele au semnal, baterie și nu vorbesc nici una cu o a treia persoană.

Problema 9.2.3.

Să se evalueze formulele următoare în interpretările precizate:

1. $U = (\exists x)A(x) \wedge (\exists x)B(x) \to (\forall x)(A(x) \vee B(x))$
   Interpretarea $I = \langle D,m \rangle$, unde:
   D = mulțimea tuturor dreptelor unui plan P
   Fie $d \in P$, un obiect (dreaptă) constant din domeniul interpretării.
   $m(A)$: $D \to \{T,F\}$, $m(A)(x)$: " $x \perp d$ ";
   $m(B)$: $D \to \{T,F\}$, $m(B)(x)$: " $x \parallel d$ ";

2. $U = (\exists x)(p(x) \wedge q(x)) \to (\exists x)p(x) \vee q(12)$
   Interpretarea: $I = \langle D,m \rangle$, unde:
   $D = \mathbf{N}$ (mulțimea numerelor naturale)
   $m(p)$: $\mathbf{N} \to \{T,F\}$, $m(p)(x)$: "$x \vdots 5$";
   $m(q)$: $\mathbf{N} \to \{T,F\}$, $m(q)(x)$: "$x \vdots 7$";

3. $U(z) = (\exists x)(\forall y)p(f(x,y),z)$
   Interpretarea $I = \langle D,m \rangle$, unde:
   $D = \mathbf{Z}$ (mulțimea numerelor întregi),
   $m(f)$: $\mathbf{Z}^2 \to \mathbf{Z}$, $m(f)(x,y) = (x+y)^2$ și
   $m(p)$: $\mathbf{Z}^2 \to \{T,F\}$, $m(p)(x,y)$:"$x > y$";

4. $U = (\forall x)(\exists y)p(x,y) \to (\exists y)(\forall x)p(x,y)$
   Interpretarea $I = \langle D,m \rangle$, unde:
   D = mulțimea tuturor triunghiurilor,
   $m(p)$: $D^2 \to \{T,F\}$, $m(p)(x,y)$: "$Aria(x) \le Aria(y)$";

5. $U = (\forall x)(\exists y)(\exists z)p(x, f(g(y), g(z)))$
   Interpretarea $I = \langle D,m \rangle$, unde:
   $D = \mathbf{N}$ (mulțimea numerelor naturale)
   $m(f)$: $\mathbf{N}^2 \to \mathbf{N}$, $m(f)(x,y) = x + y$;
   $m(g)$: $\mathbf{N} \to \mathbf{N}$, $m(g)(x) = x^2$;
   $m(p)$: $\mathbf{N}^2 \to \{T,F\}$, $m(p)(x,y)$:"$x = y$";

6. $U = (\forall x)(\forall y)(\forall z)(\forall w)(p(x,y,z,w) \to q(x,y,z,w))$
   Interpretarea $I = \langle D,m \rangle$, unde:
   $D = \mathbf{R}^2$ (mulțimea tuturor punctelor din spațiu)
   $m(p)$: $D^4 \to \{T,F\}$, $m(p)(x,y,z,w)$ = "dreapta *xz* se intersectează cu dreapta *yw*";

$m(q)$: $D^4 \to \{T,F\}$, $m(q)(x,y,z,w)$ = "punctele x, y, z şi w formează un patrulater";

7. $U = (\exists x)(p(x) \vee q(x)) \to (\exists x)p(x) \wedge (\exists x)q(x)$

   Interpretarea: I =<D,m>, unde:

   D = mulţimea tuturor maşinăriilor

   $m(p)$: $D \to \{T,F\}$, $m(p)(x)$: "x are nevoie de o sursă de energie pentru a funcţiona";

   $m(q)$: $D \to \{T,F\}$, $m(q)(x)$: "x este un *perpetuum mobile*":

8. $U = ((\forall x)A(x) \leftrightarrow (\forall x)B(x)) \to (\forall x)(A(x) \leftrightarrow B(x))$

   Interpretarea: I =<D,m>, unde:

   D = mulţimea tuturor persoanelor

   $m(A)$: $D \to \{T,F\}$, $m(A)(x)$: "persoana x ştie conduce";

   $m(B)$: $D \to \{T,F\}$, $m(B)(x)$: "persoana x este posesoarea unui automobil".

Problema 9.2.4.

Demonstraţi că formulele următoare nu sunt valide construind anti-modele pentru acestea:

1. $U = (\exists x)p(x) \to (\forall x)p(x)$;
2. $U = (\exists x)(p(x) \to q(x)) \to ((\exists x)p(x) \to (\exists x)q(x))$;
3. $U = ((\forall x)p(x) \to (\forall x)q(x)) \to (\forall x)(p(x) \to q(x))$;
4. $U = (\exists x)p(x) \wedge (\exists x)q(x) \to (\exists x)(p(x) \wedge q(x))$;
5. $U = (\forall x)(p(x) \vee q(x)) \to (\forall x)p(x) \vee (\forall x)q(x)$;
6. $U = ((\forall x)p(x) \to (\exists x)q(x)) \to (\forall x)(p(x) \to q(x))$;
7. $U = ((\exists x)p(x) \to (\exists x)q(x)) \to (\forall x)(p(x) \to q(x))$;
8. $U = (\exists x)p(x) \wedge (\exists x)q(x) \to (\forall x)(p(x) \wedge q(x))$.

Problema 9.2.5.

Alegeţi o interpretare arbitrară pentru formula U şi arătaţi că aceasta este model al formulei.

1. semidistributivitatea cuantificatorului „ $\forall$ " faţă de „ $\leftrightarrow$ ".
   $U = (\forall x)(A(x) \leftrightarrow B(x)) \to ((\forall x)A(x) \leftrightarrow (\forall x)B(x))$;
2. semidistributivitatea cuantificatorului „ $\forall$ " faţă de „ $\to$ ".
   $U = (\forall x)(A(x) \to B(x)) \to ((\forall x)A(x) \to (\forall x)B(x))$;
3. $U = (\forall x)(A(x) \leftrightarrow B(x)) \to ((\exists x)A(x) \leftrightarrow (\exists x)B(x))$;
4. $U = (\exists x)(A(x) \to B(x)) \leftrightarrow ((\forall x)A(x) \to (\exists x)B(x))$:
5. $U = ((\exists x)A(x) \to (\forall x)B(x)) \to (\forall x)(A(x) \to B(x))$;
6. $U = (\forall x)(A(x) \vee B(x)) \to ((\forall x)A(x) \vee (\exists x)B(x))$;
7. $U = (\forall x)(A(x) \to B(x)) \to ((\exists x)A(x) \to (\exists x)B(x))$;
8. $U = (\forall x)(A(x) \to B(x)) \to ((\forall x)A(x) \to (\exists x)B(x))$.

Observaţie: Toate formulele de la Problema 9.2.5. sunt valide.

Problema 9.2.6.

Construiţi toate formele normale prenexe, Skolem şi clauzale ale următoarelor formule:

1. $(\exists x)\,(\neg(\exists y)p(y) \to (\forall y)(q(y) \to r(x)))$;
2. $(\exists y)\,\neg((\exists x)r(x) \wedge (\forall x)(q(y) \to \neg p(x)))$;
3. $(\forall y)\,((\exists x)p(x) \vee (\forall z)(q(z) \to \neg r(y)))$;
4. $(\forall x)\,((\exists y)q(y) \to \neg(\forall y)(\neg r(y) \vee p(x)))$;
5. $(\exists z)\,(\neg(\forall y)p(y) \vee \neg(\exists x)(\neg r(z) \to q(x)))$;
6. $\neg(\exists x)\,(\neg(\forall y)\neg r(y) \wedge \neg(\exists y)(q(y) \vee p(x)))$;
7. $(\forall z)\,\neg((\forall y)p(y) \wedge (\exists y)(\neg q(z) \to r(y)))$;
8. $\neg(\forall y)\,((\forall x)q(x) \wedge (\exists x)(\neg p(y) \wedge \neg r(x)))$.

Problema 9.2.7.

Aduceţi la o formă normală prenexă şi la o formă normală clauzală următoarele formule:

1. $(\forall x)(\forall y)((\exists z)p(z) \wedge (\exists u)(q(x,u) \to (\exists z)q(y,z)))$;
2. $(\exists y)\neg(\forall z)((\forall x)\neg p(x) \vee (\forall u)(q(y,u) \wedge (\forall x)\neg q(x,z)))$;
3. $\neg(\forall x)(\exists y)((\exists z)p(z) \to \neg(\exists u)(q(x,u) \to (\exists z)q(y,z)))$;
4. $(\exists x)(\exists y)\neg((\exists z)p(z) \to (\exists u)(q(x,u) \wedge (\forall z)\neg q(y,z)))$;
5. $(\forall z)(\exists y)((\exists x)p(x) \wedge \neg(\exists u)(q(z,u) \wedge \neg(\exists x)q(y,x)))$;
6. $(\forall x)(\forall y)((\exists z)p(z) \wedge (\forall u)(\neg q(x,u) \vee (\exists z)q(y,z)))$;
7. $(\forall x)(\forall y)\neg((\forall z)\neg p(z) \vee (\forall u)\neg(\neg q(x,u) \vee (\forall z)q(y,z)))$;
8. $(\exists x)\neg(\exists z)((\exists y)p(y) \to (\exists u)\neg(q(x,u) \to (\exists y)q(z,y)))$.

Problema 9.2.8.

Sunt unificabili atomii din perechile următoare? Dacă da, aflaţi cel mai general unificator al acestora. Prin convenţie: a,b,c sunt constante, x,y,z,u sunt variabile, iar f,g,h sunt simboluri de funcţii.

1. $P(a,x,g(g(y)))$ şi $P(y,f(z),f(z))$;
   $P(x,g(f(a)),f(x))$ şi $P(f(y),z,y)$;
   $P(a,x,g(g(y)))$ şi $P(z,h(z,u),g(u))$;

2. $P(a,x,f(g(y)))$ şi $P(y,f(z),f(z))$;
$P(x,g(f(a)),f(b))$ şi $P(f(y),z,z)$;
$P(a,x,f(g(y)))$ şi $P(z,h(z,u),f(b),z)$;
3. $P(a,f(x),g(h(y)))$ şi $P(y,f(z),g(z))$;
$P(x,g(f(a)),h(x,y))$ şi $P(f(z),g(z),y)$;
$P(g(y),x,f(g(y)))$ şi $P(z,h(z,u),f(u))$;
4. $P(a,g(x),f(g(y)))$ şi $P(y,z,f(z))$;
$P(b,g(f(a)),z)$ şi $P(f(y),z,g(y))$;
$P(a,h(x,b),f(g(y)))$ şi $P(z,h(z,u),f(u))$;
5. $P(a,x,g(f(y)))$ şi $P(f(z),z,g(x))$;
$P(a,x,g(f(y)))$ şi $P(x,y,g(f(b)))$;
$P(a,h(x,u),g(z))$ şi $P(y,h(y,f(z)),g(x))$;
6. $P(a,y,g(f(z)))$ şi $P(z,f(z),x)$;
$P(y,f(x),z)$ şi $P(y,f(y),f(y))$;
$P(h(x,y),x,y)$ şi $P(h(y,x),f(z),z)$;
7. $P(a,x,g(f(y)))$ şi $P(f(y),z,x)$;
$P(x,a,g(b))$ şi $P(f(y),f(y),g(x))$;
$P(h(x,a),f(z),z)$ şi $P(h(y,x),f(x),a)$;
8. $P(a,x,g(f(y)))$ şi $P(f(y),f(z),g(z))$;
$P(x,g(f(a)),x)$ şi $P(f(y),z,h(y,f(y)))$;
$P(a,h(x,u),f(g(y)))$ şi $P(z,h(z,u),f(u))$.

Problema 9.2.9.

Utilizând metoda lui Herbrand demonstraţi că următoarele mulţimi de formule predicative sunt contradictorii (inconsistente):

1. $S=\{p(x)\to\neg q(x)\vee r(x), \neg q(y)\vee\neg r(y), p(a)\wedge q(a)\}$;
2. $S=\{p(a)\vee q(x)\vee r(x), p(y)\to r(y), \neg q(x)\wedge\neg r(a)\}$;
3. $S=\{p(x)\to q(x)\wedge r(x), q(y)\to r(y), p(b), \neg r(b)\}$;
4. $S=\{\neg p(x)\wedge q(x)\to\neg r(x), q(y)\vee\neg r(y), \neg p(a)\wedge r(a)\}$;
5. $S=\{\neg(p(a)\wedge r(x))\vee\neg q(x), \neg r(y)\to\neg q(y), p(x), q(a)\}$;
6. $S=\{p(x)\vee q(x)\to r(x), q(y)\vee r(y), p(a)\wedge\neg r(a)\}$;
7. $S=\{\neg q(a)\wedge\neg p(x)\vee r(x), p(x)\vee q(x), \neg r(a)\}$;
8. $S=\{\neg p(x)\vee q(x), p(a)\vee\neg r(x), q(y)\to\neg r(y), r(a)\}$.

Problema 9.2.10.

Utilizaţi metoda lui Herbrand pentru a verifica următoarele teoreme:

1. semidistributivitatea cuantificatorului „$\exists$" faţă de „$\to$":
$\vdash((\exists x)p(x)\to(\exists x)q(x))\to(\exists x)(p(x)\to q(x))$;
2. semidistributivitatea cuantificatorului „$\forall$" faţă de „$\to$":
$\vdash(\forall x)(p(x)\to q(x))\to((\forall x)p(x)\to(\forall x)q(x))$;
3. semidistributivitatea cuantificatorului „$\forall$" faţă de „$\vee$":
$\vdash(\forall x)p(x)\vee(\forall x)q(x)\to(\forall x)(p(x)\vee q(x))$.
4. $\vdash(\forall x)p(x)\wedge(\forall x)q(x)\to(\forall x)(p(x)\wedge q(x))$;
5. $\vdash(\exists x)(p(x)\vee q(x))\to(\exists x)p(x)\vee(\exists x)q(x)$;
6. $\vdash(\forall x)(p(x)\wedge q(x))\to(\forall x)p(x)\wedge(\forall x)q(x)$;
7. $\vdash(\exists x)p(x)\vee(\exists x)q(x)\to(\exists x)(p(x)\vee q(x))$;
8. $\vdash(\forall x)(p(x)\wedge q(x))\to(\exists x)p(x)\wedge(\exists x)q(x)$;

Problema 9.2.11.

Utilizând metoda calculului secvenţelor, verificaţi:

1. semidistributivitatea cuantificatorului „$\exists$" faţă de „$\wedge$":
$\vdash(\exists x)(p(x)\wedge q(x))\to(\exists x)p(x)\wedge(\exists x)q(x)$;
Arătaţi că implicaţia inversă nu are loc.
2. semidistributivitatea cuantificatorului „$\forall$" faţă de „$\vee$":
$\vdash(\forall x)p(x)\vee(\forall x)q(x)\to(\forall x)(p(x)\vee q(x))$;
Arătaţi că implicaţia inversă nu are loc.
3. distributivitatea cuantificatorului „$\exists$" faţă de „$\vee$":
$\vdash(\exists x)(p(x)\vee q(x))\leftrightarrow(\exists x)p(x)\vee(\exists x)q(x)$;
4. distributivitatea cuantificatorului „$\forall$" faţă de „$\wedge$":
$\vdash(\forall x)p(x)\wedge(\forall x)q(x)\leftrightarrow(\forall x)(p(x)\wedge q(x))$;
5. semidistributivitatea cuantificatorului „$\forall$" faţă de „$\to$":
$\vdash(\forall x)(p(x)\to q(x))\to((\forall x)p(x)\to(\forall x)q(x))$;
Arătaţi că implicaţia inversă nu are loc.
6. semidistributivitatea cuantificatorului „$\exists$" faţă de „$\to$":
$\vdash((\exists x)p(x)\to(\exists x)q(x))\to(\exists x)(p(x)\to q(x))$
Arătaţi că implicaţia inversă nu are loc.
7. $\vdash(\exists x)p(x)\vee(\exists x)(p(x)\wedge q(x))\leftrightarrow(\exists x)p(x)$;
8. $\vdash(\forall x)p(x)\wedge((\forall x)p(x)\vee(\forall x)q(x))\leftrightarrow(\forall x)p(x)$;

Problema 9.2.12.

Utilizând calculul secvențelor, verificați dacă au loc deducțiile:

1. $p(a), (\forall x)(p(x) \to p(f(x))) \vdash (\forall x)p(x)$;
2. $(\forall x)(p(x) \to q(x)), (\forall x)p(x) \vdash (\forall x)q(x)$;
3. $(\forall x)(\forall y)(q(x,y) \to p(x,y)), (\forall z)q(z,z) \vdash (\forall x)p(x,x)$;
4. $(\exists x)(\forall y)(p(x,y) \to r(x)), (\forall x)(\forall y)p(x,y) \vdash (\exists z)r(z)$;
5. $(\forall x)(p(x) \to q(x)), (\exists x)p(x) \vdash (\exists x)q(x)$;
6. $(\exists x)(\forall y)(q(x,y) \to p(x,y)), (\forall z)q(z,z) \vdash (\exists x)p(x,x)$;
7. $(\forall x)(\forall y)(p(x,y) \to r(x)), (\exists x)(\exists y)p(x,y) \vdash (\exists z)r(z)$;
8. $(\forall x)(\forall y)((p(x,y) \to p(y,x)) \vdash (\forall x)p(x,x)$.

Problema 9.2.13.

Utilizând metoda tabelelor semantice (construind arborele binar și/sau funcția TP), demonstrați:

1. distributivitatea cuantificatorului „$\exists$" față de „$\vee$":
   $\vdash (\exists x)(p(x) \vee q(x)) \leftrightarrow (\exists x)p(x) \vee (\exists x)q(x)$;
2. distributivitatea cuantificatorului „$\forall$" față de „$\wedge$":
   $\vdash (\forall x)p(x) \wedge (\forall x)q(x) \leftrightarrow (\forall x)(p(x) \wedge q(x))$;
3. semidistributivitatea cuantificatorului „$\forall$" față de „$\to$":
   $\vdash (\forall x)(p(x) \to q(x)) \to ((\forall x)p(x) \to (\forall x)q(x))$;
   Arătați că implicația inversă nu are loc.
4. semidistributivitatea cuantificatorului „$\exists$" față de „$\wedge$":
   $\vdash (\exists x)(p(x) \wedge q(x)) \to (\exists x)p(x) \wedge (\exists x)q(x)$;
   Arătați că implicația inversă nu are loc.
5. semidistributivitatea cuantificatorului „$\forall$" față de „$\vee$":
   $\vdash (\forall x)p(x) \vee (\forall x)q(x) \to (\forall x)(p(x) \vee q(x))$;
   Arătați că implicația inversă nu are loc.
6. $\vdash (\forall x)p(x) \wedge ((\forall x)p(x) \vee (\forall x)q(x)) \leftrightarrow (\forall x)p(x)$;
7. $\vdash (\exists x)p(x) \vee (\exists x)(p(x) \wedge q(x)) \leftrightarrow (\exists x)p(x)$;
8. $\vdash (\exists x)(p(x) \to q(x)) \leftrightarrow ((\forall x)p(x) \to (\exists x)q(x))$.

Problema 9.2.14.

Utilizând metoda tabelelor semantice (construind arborele binar și/sau funcția TP), verificați dacă au loc:

1. $\vdash (\forall x)(\forall y)p(x,y) \leftrightarrow (\exists x)(\forall y)p(x,y)$;
2. $\vdash (\exists x)(\forall y)p(x,y) \leftrightarrow (\forall y)(\exists x)p(x,y)$;
3. $\vdash (\forall y)(\exists x)p(x,y) \leftrightarrow (\exists y)(\exists x)p(x,y)$;
4. $\vdash (\forall x)(\forall y)p(x,y) \leftrightarrow (\forall y)(\forall x)p(x,y)$;
5. $\vdash (\forall y)(\forall x)p(x,y) \leftrightarrow (\forall x)(\exists y)p(x,y)$;
6. $\vdash (\exists y)(\exists x)p(x,y) \leftrightarrow (\exists x)(\forall y)p(x,y)$;
7. $\vdash (\exists y)(\exists x)p(x,y) \leftrightarrow (\forall x)(\exists y)p(x,y)$;
8. $\vdash (\exists y)(\exists x)p(x,y) \leftrightarrow (\exists x)(\exists y)p(x,y)$.

Problema 9.2.15.

Demonstrați inconsistența următoarelor mulțimi de clauze utilizând rezoluția blocării. Utilizați două indexări diferite pentru literalii din clauze.

1. $S = \{\neg p(x) \vee q(x),\ p(a),\ \neg q(x) \vee \neg r(x),\ \neg w(a),\ r(y) \vee w(y)\}$;
2. $S = \{p(x) \vee \neg q(x),\ \neg p(a) \vee r(x),\ q(x),\ w(z),\ \neg r(y) \vee \neg w(y)\}$;
3. $S = \{p(x) \vee q(x) \vee r(x),\ \neg p(b),\ \neg q(x),\ \neg w(b),\ \neg r(y) \vee w(y)\}$;
4. $S = \{p(x) \vee q(x),\ \neg p(x) \vee r(x),\ \neg q(y) \vee r(y),\ \neg r(x) \vee w(x),\ \neg w(f(z))\}$
5. $S = \{p(x) \vee q(x),\ \neg p(a) \vee w(x),\ \neg q(y) \vee r(y),\ \neg r(x) \vee w(x),\ \neg w(a)\}$;
6. $S = \{\neg p(x) \vee \neg q(x),\ p(z) \vee w(x),\ q(y) \vee w(y) \vee \neg r(y),$
   $\neg r(x) \vee \neg w(x),\ r(g(a,b)\}$;
7. $S = \{p(x) \vee q(x),\ \neg p(x),\ \neg q(f(a)) \vee r(z),\ \neg w(z),\ \neg r(y) \vee w(y)\}$
8. $S = \{\neg p(x) \vee q(x) \vee \neg r(x),\ p(f(b)),\ \neg q(x), \neg w(y),\ r(y) \vee w(y)\}$.

Problema 9.2.16.

Verificați dacă următoarele formule sunt teoreme utilizând rezoluția generală:

1. $(\forall x)(\exists y)(\forall z)p(x,y,z) \to (\exists y)(\forall z)p(z,y,z)$;
2. $(\forall x)(\forall y)((p(x,y) \to p(y,x)) \to (\forall x)p(x,x)$;
3. $((\exists x)p(x) \to (\exists x)q(x)) \to (\exists x)(p(x) \to q(x))$;
4. $(\forall x)(p(x) \to q(x)) \to ((\exists x)p(x) \to (\exists x)q(x))$;
5. $p(a) \wedge (\forall x)(p(x) \to p(f(x))) \to (\forall x)p(x)$;
6. $(\forall x)(\forall y)(q(x,y) \to p(x,y)) \to ((\forall z)q(z,z) \to (\forall x)p(x,x))$;
7. $((\forall x)p(x) \to (\forall x)q(x)) \to (\forall x)(p(x) \to q(x))$;
8. $(\exists x)(\forall y)(p(x,y) \to r(x)) \to ((\forall x)(\forall y)p(x,y) \to (\exists z)r(z))$.

Problema 9.2.17.

Verificați dacă se poate obține concluzia pornind de la ipoteze, prin utilizarea rezoluției cu strategia input și a clauzei rădăcină negativă:

1. $(\forall x)(\forall y)(q(x,y) \to p(x,y)), (\forall z)q(z,z) \vdash (\forall x)p(x,x)$;
2. $(\exists x)(\forall y)(p(x,y) \to r(x)), (\forall x)(\forall y)p(x,y) \vdash (\exists z)r(z)$;
3. $(\forall x)p(a,x,x), (\forall x)(\forall y)(\forall h)(\forall t)(p(t,x,y) \to p(f(h,t),x,f(h,y)))$
   $\vdash p(f(2,a), f(3,a), f(2,f(3,a)))$;
4. $(\forall x)(p(x) \to r(x)), (\forall y)(r(y) \to q(y)), p(a), p(b) \vdash (\exists z)q(z)$;
5. $(\forall x)(\neg p(x) \wedge \neg q(x) \to r(x)), (\forall y)(r(y) \to w(y)), (\forall x)(w(x) \to p(x)),$
   $\neg p(a), \neg p(b), \neg w(c) \vdash (\exists z)q(z)$;
6. $(\forall x)(p(x) \to r(x)), (\forall y)(r(y) \to q(y)), r(a), r(b), \neg r(c) \vdash (\exists z)q(z)$;
7. $(\forall x)(p(x) \to r(x)), (\forall y)(p(y) \to q(y)), p(a), p(b), \neg p(c) \vdash (\exists z)q(z)$;
8. $(\forall x)(\forall y)(p(y,x) \wedge q(x) \to q(y)), (\forall x)(\forall y)(r(y,x) \wedge q(x) \to q(y)),$
   $q(a), r(b,a), p(c,b) \vdash q(c)$.

Problema 9.2.18.

Să se demonstreze deducțiile următoare utilizând o strategie/rafinare a rezoluției:

1. $(\forall x)(\forall y)(p(y,x) \wedge q(x) \to q(y)), (\forall x)(\forall y)(r(y,x) \to q(y)),$
   $r(b,a), p(c,b) \vdash (\exists z)q(z)$;
2. $(\forall x)(p(x) \to r(x)), (\forall y)(r(y) \to q(y)), p(a), p(b) \vdash (\exists z)q(z)$;
3. $(\forall x)(\neg p(x) \wedge \neg q(x) \to r(x)), (\forall y)(r(y) \to w(y)), (\forall x)(w(x) \to p(x)),$
   $\neg p(a), \neg p(b), \neg w(c) \vdash (\exists z)q(z)$;
4. $(\forall x)(p(x) \to r(x)), (\forall y)(r(y) \to q(y)), r(a), r(b), \neg r(c) \vdash (\exists z)q(z)$;
5. $(\forall x)(\neg p(x) \wedge \neg q(x) \to r(x)), (\forall y)(r(y) \to w(y)), (\forall x)(w(x) \to p(x)),$
   $\neg p(a), \neg w(c) \vdash (\exists z)q(z)$;
6. $(\forall x)(\forall y)(\neg p(y,x) \to q(y)), (\forall x)(\forall y)(r(y,x) \wedge q(x) \to q(y)),$
   $r(b,a), \neg p(a,b) \vdash (\exists z)q(z)$;
7. $(\forall x)(p(x) \to r(x)), (\forall y)(p(y) \to q(y)), p(a), \neg r(c) \vdash (\exists z)q(z)$;
8. $(\forall x)(p(x) \to r(x)), (\forall y)(p(y) \to q(y)), p(a), p(b), \neg p(c) \vdash (\exists z)q(z)$.

Problema 9.2.19.

Utilizând rezoluția liniară, demonstrați:

1. semidistributivitatea cuantificatorului „ $\forall$ ” față de „ $\vee$ ”:
   $\vdash (\forall x)p(x) \vee (\forall x)q(x) \to (\forall x)(p(x) \vee q(x))$;
   Arătați că implicația inversă nu are loc.
2. semidistributivitatea cuantificatorului „ $\exists$ ” față de „ $\wedge$ ”:
   $\vdash (\exists x)(p(x) \wedge q(x)) \to (\exists x)p(x) \wedge (\exists x)q(x)$;
   Arătați că implicația inversă nu are loc.
3. $\vdash (\exists x)(p(x) \to q(x)) \leftrightarrow ((\forall x)p(x) \to (\exists x)q(x))$;
4. distributivitatea cuantificatorului „ $\exists$ ” față de „ $\vee$ ”:
   $\vdash (\exists x)(p(x) \vee q(x)) \leftrightarrow (\exists x)p(x) \vee (\exists x)q(x)$;
5. $\vdash (\exists x)p(x) \vee (\exists x)(p(x) \wedge q(x)) \leftrightarrow (\exists x)p(x)$;
6. semidistributivitatea cuantificatorului „ $\forall$ ” față de „ $\to$ ”:
   $\vdash (\forall x)(p(x) \to q(x)) \to ((\forall x)p(x) \to (\forall x)q(x))$;
   Arătați că implicația inversă nu are loc.
7. $\vdash (\forall x)p(x) \wedge ((\forall x)p(x) \vee (\forall x)q(x)) \leftrightarrow (\forall x)p(x)$;
8. distributivitatea cuantificatorului „ $\forall$ ” față de „ $\wedge$ ”:
   $\vdash (\forall x)p(x) \wedge (\forall x)q(x) \leftrightarrow (\forall x)(p(x) \wedge q(x))$.

Problema 9.2.20.

Verificați următoarele echivalențe utilizând o strategie/rafinare a rezoluției predicative:

1. $(\forall x)(\forall y)p(x,y) \leftrightarrow (\forall y)(\forall x)p(x,y)$;
2. $(\exists y)(\exists x)p(x,y) \leftrightarrow (\exists x)(\exists y)p(x,y)$;
3. $(\forall x)(\forall y)p(x,y) \leftrightarrow (\exists x)(\forall y)p(x,y)$;
4. $(\exists x)(\forall y)p(x,y) \leftrightarrow (\forall y)(\exists x)p(x,y)$;
5. $(\exists y)(\exists x)p(x,y) \leftrightarrow (\forall x)(\exists y)p(x,y)$;
6. $(\forall y)(\forall x)p(x,y) \leftrightarrow (\forall x)(\exists y)p(x,y)$;
7. $(\exists y)(\exists x)p(x,y) \leftrightarrow (\exists x)(\forall y)p(x,y)$;
8. $(\forall y)(\exists x)p(x,y) \leftrightarrow (\exists y)(\exists x)p(x,y)$.

Problema 9.2.21.

Utilizând rezoluția generală, verificați dacă formulele următoare sunt sau nu sunt teoreme:

1. $(\forall x)(\exists y)\neg(p(y,x) \leftrightarrow \neg p(y,y))$;

2. $(\forall x)(\exists y)\neg(p(x,y)\leftrightarrow\neg p(y,y))$;
3. $(\forall x)(\exists y)\neg(p(y,y)\leftrightarrow\neg p(x,y))$;
4. $(\forall x)(\exists y)\neg(p(y,y)\leftrightarrow\neg p(y,x))$;
5. $(\exists y)(\exists x)p(x,y)\leftrightarrow(\exists x)(\forall y)p(x,y)$;
6. $(\exists y)(\exists x)p(x,y)\leftrightarrow(\forall x)(\exists y)p(x,y)$;
7. $(\forall y)(\forall x)p(x,y)\leftrightarrow(\forall x)(\exists y)p(x,y)$;
8. $(\forall y)(\exists x)\neg(p(x,y)\leftrightarrow\neg p(x,x))$.

9.3. Probleme cu algebre booleene, funcții booleene şi circuite logice

Problema 9.3.1.

Pentru următoarele funcții booleene de trei variabile, date prin intermediul tabelelor de valori, scrieți cele două forme canonice: *conjunctivă* (FCC) şi *disjunctivă* (FCD). Simplificați funcțiile utilizând diagrame Veitch.

x	y	z	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
0	0	0	0	0	1	1	0	1	0	1
0	0	1	1	1	1	0	1	0	0	0
0	1	0	0	0	1	0	1	1	1	1
0	1	1	1	0	0	1	0	1	0	0
1	0	0	1	1	1	0	0	0	1	0
1	0	1	0	1	0	0	1	0	1	1
1	1	0	0	0	0	1	1	1	0	1
1	1	1	1	1	0	1	0	0	1	0

Problema 9.3.2.

Simplificați următoarele funcții booleene de patru variabile, date prin formele canonice disjunctive, utilizând diagrame Veitch:

1. $f_1(x_1,x_2,x_3,x_4)=x_1x_2x_3x_4\vee\bar x_1x_2\bar x_3x_4\vee\bar x_1x_2x_3x_4\vee x_1x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3\bar x_4$
$\vee x_1\bar x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3x_4\vee\bar x_1\bar x_2\bar x_3x_4\vee\bar x_1\bar x_2x_3x_4$;
2. $f_2(x_1,x_2,x_3,x_4)=x_1x_2x_3x_4\vee x_1x_2x_3\bar x_4\vee\bar x_1x_2\bar x_3x_4\vee x_1\bar x_2\bar x_3\bar x_4\vee\bar x_1\bar x_2\bar x_3\bar x_4$
$\vee\bar x_1\bar x_2x_3\bar x_4\vee x_1\bar x_2\bar x_3x_4\vee\bar x_1\bar x_2\bar x_3x_4\vee\bar x_1\bar x_2x_3x_4$;
3. $f_3(x_1,x_2,x_3,x_4)=\bar x_1x_2x_3x_4\vee\bar x_1x_2\bar x_3\bar x_4\vee\bar x_1x_2\bar x_3x_4\vee x_1\bar x_2x_3\bar x_4\vee x_1\bar x_2\bar x_3\bar x_4$
$\vee\bar x_1\bar x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3x_4\vee x_1\bar x_2\bar x_3x_4\vee\bar x_1\bar x_2\bar x_3x_4$;
4. $f_4(x_1,x_2,x_3,x_4)=\bar x_1x_2\bar x_3x_4\vee x_1x_2\bar x_3\bar x_4\vee\bar x_1x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3\bar x_4\vee x_1\bar x_2\bar x_3\bar x_4$
$\vee\bar x_1\bar x_2x_3\bar x_4\vee x_1\bar x_2x_3x_4\vee x_1\bar x_2\bar x_3x_4\vee\bar x_1\bar x_2x_3x_4$;
5. $f_5(x_1,x_2,x_3,x_4)=x_1x_2x_3x_4\vee x_1x_2\bar x_3x_4\vee x_1x_2x_3\bar x_4\vee\bar x_1x_2\bar x_3\bar x_4\vee\bar x_1x_2x_3\bar x_4$
$\vee x_1\bar x_2x_3\bar x_4\vee x_1\bar x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3\bar x_4\vee x_1\bar x_2\bar x_3x_4$;
6. $f_6(x_1,x_2,x_3,x_4)=x_1x_2x_3x_4\vee\bar x_1x_2\bar x_3x_4\vee\bar x_1x_2x_3x_4\vee x_1x_2\bar x_3x_4\vee\bar x_1x_2\bar x_3\bar x_4$
$\vee\bar x_1x_2x_3\bar x_4\vee x_1\bar x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3x_4\vee x_1\bar x_2\bar x_3x_4$;
7. $f_7(x_1,x_2,x_3,x_4)=x_1x_2x_3x_4\vee\bar x_1x_2\bar x_3x_4\vee\bar x_1x_2x_3x_4\vee x_1\bar x_2\bar x_3x_4\vee x_1\bar x_2x_3\bar x_4$
$\vee x_1\bar x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3x_4\vee\bar x_1\bar x_2\bar x_3x_4\vee\bar x_1\bar x_2x_3x_4$;
8. $f_8(x_1,x_2,x_3,x_4)=x_1x_2x_3x_4\vee x_1x_2\bar x_3x_4\vee\bar x_1x_2\bar x_3x_4\vee\bar x_1x_2\bar x_3\bar x_4\vee\bar x_1x_2x_3\bar x_4$
$\vee x_1\bar x_2x_3\bar x_4\vee x_1\bar x_2\bar x_3\bar x_4\vee x_1\bar x_2x_3x_4\vee x_1\bar x_2\bar x_3x_4$.

Problema 9.3.3.

Simplificați următoarele funcții booleene de trei variabile, date prin mintermii expresiilor, utilizând diagrame Karnaugh:

1. $f_1(x_1,x_2,x_3)=m_0\vee m_3\vee m_4\vee m_5\vee m_6\vee m_7$;
2. $f_2(x_1,x_2,x_3)=m_1\vee m_2\vee m_4\vee m_5\vee m_6\vee m_7$;
3. $f_3(x_1,x_2,x_3)=m_1\vee m_2\vee m_3\vee m_4\vee m_5\vee m_7$;
4. $f_4(x_1,x_2,x_3)=m_0\vee m_1\vee m_2\vee m_3\vee m_5\vee m_6$;
5. $f_5(x_1,x_2,x_3)=m_0\vee m_1\vee m_2\vee m_4\vee m_6\vee m_7$;
6. $f_6(x_1,x_2,x_3)=m_0\vee m_1\vee m_3\vee m_5\vee m_6\vee m_7$;
7. $f_7(x_1,x_2,x_3)=m_0\vee m_1\vee m_2\vee m_3\vee m_4\vee m_7$;
8. $f_8(x_1,x_2,x_3)=m_0\vee m_2\vee m_3\vee m_4\vee m_5\vee m_6$.

Problema 9.3.4.

Utilizând metoda lui Quine simplificați următoarele funcții booleene de trei variabile:

1. $f_1(x_1,x_2,x_3)=x_1(\bar x_2\uparrow x_3)\vee\bar x_1x_2$;
2. $f_2(x_1,x_2,x_3)=x_3(x_1\vee x_2)\vee(\bar x_2\downarrow x_3)$;
3. $f_3(x_1,x_2,x_3)=x_2(x_1\uparrow\bar x_3)\vee\bar x_2x_3$;
4. $f_4(x_1,x_2,x_3)=x_1(x_2\vee x_3)\vee(x_1\downarrow\bar x_3)$;
5. $f_5(x_1,x_2,x_3)=x_3(\bar x_1\uparrow x_2)\vee x_1\bar x_3$;
6. $f_6(x_1,x_2,x_3)=x_2(x_1\vee x_3)\vee(\bar x_1\downarrow x_2)$;
7. $f_7(x_1,x_2,x_3)=x_2(\bar x_1\uparrow x_3)\vee x_1\bar x_2$;
8. $f_8(x_1,x_2,x_3)=x_3(x_1\vee x_2)\vee(\bar x_1\downarrow x_3)$.

Problema 9.3.5.

Simplificați următoarele funcții booleene de patru variabile date prin valorile de 1, utilizând metoda lui Quine:

1. $f_1(1,1,1,1)=1$, $f_1(1,1,0,1)=1$, $f_1(0,1,1,1)=1$, $f_1(1,1,0,0)=1$, $f_1(0,1,0,0)=1$, $f_1(0,0,0,0)=1$, $f_1(0,0,0,1)=1$, $f_1(0,0,1,1)=1$;

2. $f_2(1,1,0,1)=1$, $f_2(0,1,0,1)=1$, $f_2(0,1,0,0)=1$, $f_2(0,0,0,0)=1$, $f_2(0,0,1,0)=1$, $f_2(1,0,1,1)=1$, $f_2(1,0,0,1)=1$, $f_2(0,0,1,1)=1$;
3. $f_3(0,1,0,1)=1$, $f_3(0,1,0,0)=1$, $f_3(0,1,1,0)=1$, $f_3(1,0,1,0)=1$, $f_3(1,0,0,0)=1$, $f_3(0,0,1,0)=1$, $f_3(1,0,0,1)=1$, $f_3(0,0,0,1)=1$;
4. $f_4(0,1,0,1)=1$, $f_4(0,1,1,1)=1$, $f_4(1,1,1,0)=1$, $f_4(1,1,0,0)=1$, $f_4(0,1,1,0)=1$, $f_4(1,0,0,0)=1$, $f_4(0,0,0,0)=1$, $f_4(0,0,0,1)=1$;
5. $f_5(1,1,1,1)=1$, $f_5(0,1,0,1)=1$, $f_5(0,1,1,1)=1$, $f_5(1,1,1,0)=1$, $f_5(1,1,0,0)=1$, $f_5(1,0,0,0)=1$, $f_5(1,0,0,1)=1$, $f_5(0,0,0,1)=1$;
6. $f_6(1,1,0,1)=1$, $f_6(0,1,0,1)=1$, $f_6(0,1,1,1)=1$, $f_6(1,1,1,0)=1$, $f_6(0,1,1,0)=1$, $f_6(1,0,1,0)=1$, $f_6(1,0,1,1)=1$, $f_6(1,0,0,1)=1$;
7. $f_7(1,1,1,1)=1$, $f_7(1,1,0,1)=1$, $f_7(0,1,0,1)=1$, $f_7(0,1,0,0)=1$, $f_7(0,1,1,0)=1$, $f_7(0,0,1,0)=1$, $f_7(1,0,1,1)=1$, $f_7(0,0,1,1)=1$;
8. $f_8(1,1,1,1)=1$, $f_8(1,1,1,0)=1$, $f_8(1,1,0,0)=1$, $f_8(1,0,0,0)=1$, $f_8(0,0,0,0)=1$, $f_8(0,0,1,0)=1$, $f_8(1,0,1,1)=1$, $f_8(0,0,1,1)=1$.

Problema 9.3.6.

Simplificați următoarele funcții booleene de trei variabile date prin zerourile acestora, utilizând metoda de simplificare a lui Moisil și metoda lui Quine pentru identificarea monoamelor maximale:

1. $f_1(0,1,0) = f_1(0,1,1) = f_1(1,0,1) = 0$;
2. $f_2(0,0,0) = f_2(0,0,1) = f_2(1,1,1) = 0$;
3. $f_3(0,0,1) = f_3(0,1,0) = f_3(1,1,0) = 0$;
4. $f_4(0,0,0) = f_4(0,1,1) = f_4(1,0,0) = 0$;
5. $f_5(0,0,0) = f_5(1,1,0) = f_5(1,1,1) = 0$;
6. $f_6(0,1,0) = f_6(1,0,0) = f_6(1,0,1) = 0$;
7. $f_7(0,1,1) = f_7(1,0,0) = f_7(1,1,1) = 0$;
8. $f_8(0,0,1) = f_8(1,0,1) = f_8(1,1,0) = 0$.

Problema 9.3.7.

Simplificați următoarele funcții booleene de patru variabile utilizând metoda de simplificare a lui Moisil și diagrame Veitch-Karnaugh pentru identificarea monoamelor maximale:

1. $f_1(x_1,x_2,x_3,x_4)= x_1\bar{x}_4 \vee x_1x_2\bar{x}_3x_4 \vee \bar{x}_1x_2x_4 \vee \bar{x}_1x_3 \vee x_3\bar{x}_4$;
2. $f_2(x_1,x_2,x_3,x_4)= x_1x_2 \vee x_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2x_4 \vee \bar{x}_1x_3 \vee x_2x_3$;
3. $f_3(x_1,x_2,x_3,x_4)= x_1x_4 \vee x_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_4 \vee \bar{x}_1x_3 \vee x_3\, x_4$;
4. $f_4(x_1,x_2,x_3,x_4)= x_1\bar{x}_2 \vee x_1x_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1x_2\bar{x}_4 \vee \bar{x}_1x_3 \vee \bar{x}_2x_3$;
5. $f_5(x_1,x_2,x_3,x_4)= x_3\bar{x}_4 \vee x_1x_2x_3x_4 \vee \bar{x}_3x_2x_4 \vee \bar{x}_1\bar{x}_3 \vee \bar{x}_1\bar{x}_4$;
6. $f_6(x_1,x_2,x_3,x_4)= \bar{x}_1\bar{x}_4 \vee \bar{x}_1x_2x_3x_4 \vee x_1x_2x_4 \vee x_1\bar{x}_3 \vee \bar{x}_3\bar{x}_4$;
7. $f_7(x_1,x_2,x_3,x_4)= \bar{x}_3\bar{x}_4 \vee \bar{x}_1x_2\bar{x}_3x_4 \vee x_2x_3x_4 \vee x_1x_3 \vee x_1\bar{x}_4$;
8. $f_8(x_1,x_2,x_3,x_4)= x_3x_4 \vee x_1\bar{x}_2x_3\bar{x}_4 \vee \bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_3 \vee \bar{x}_1x_4$.

Problema 9.3.8.

Desenați circuitele logice asociate funcțiilor booleene de mai jos folosind porți derivate. Pentru fiecare funcție desenați circuitele logice corespunzătoare tuturor formelor simplificate ale funcției utilizând doar porți de bază:

1. $f_1(x, y, z) = x(y \oplus z) \vee y(x \oplus z) \vee x(\bar{y} \downarrow \bar{z}) \vee (x \downarrow y)\bar{z}$;
2. $f_2(x, y, z) = x(y\uparrow z) \vee \bar{x}(\bar{y} \oplus z) \vee y(\bar{x} \oplus \bar{z})$;
3. $f_3(x, y, z) = x(\bar{y} \oplus z) \vee y(\bar{x} \oplus z) \vee \bar{x}(\bar{y} \downarrow z) \vee (\bar{x} \downarrow y)z$;
4. $f_4(x, y, z) = \bar{x}(y\uparrow \bar{z}) \vee x(\bar{y} \oplus z) \vee \bar{y}(\bar{x} \oplus z)$;
5. $f_5(x, y, z) = \bar{x}(y \oplus \bar{z}) \vee \bar{y}(x \oplus z) \vee \bar{x}(y \downarrow z) \vee (\bar{x} \downarrow y)\bar{z}$;
6. $f_6(x, y, z) = x(\bar{y}\uparrow \bar{z}) \vee \bar{x}(y \oplus z) \vee \bar{y}(\bar{x} \oplus z)$;
7. $f_7(x, y, z) = x(y \oplus \bar{z}) \vee y(\bar{x} \oplus z) \vee x(y \downarrow z) \vee (x \downarrow y)\bar{z}$;
8. $f_8(x, y, z) = x(\bar{y}\uparrow z) \vee \bar{x}(\bar{y} \oplus z) \vee y(x \oplus \bar{z})$.

Problema 9.3.9.

Desenați un circuit logic având trei variabile de intrare și conținând toate porțile de bază și derivate. Scrieți funcția booleană corespunzătoare și simplificați-o, iar apoi desenați un circuit logic simplificat.

9.4. Probleme de modelare a raționamentului

Setul de probleme din acest subcapitol ilustrează modul de formalizare a raționamentului cotidian folosind logica propozițiilor și cea a predicatelor.

Problema 9.4.1.

Scrieți o mică bază de cunoștințe (4-5 afirmații) într-un domeniu ales. Transformați afirmațiile din limbaj natural în formule propoziționale/predicative. Folosind două strategii/rafinări ale rezoluției verificați validitatea a două inferențe.

Problema 9.4.2. [6]

La un tribunal, doi martori au afirmat „a”, respectiv „b”. Referindu-se la aceste afirmații, procurorul a declarat „x”. În încheiere președintele spune: „Dacă atât procurorul cât și primul martor au dreptate, atunci al doilea martor nu are dreptate.”

Aşadar, procurorul are dreptate sau martorii au dreptate?
Precizaţi în ce caz este eronat raţionamentul preşedintelui.

Problema 9.4.3. [25]

La o întreprindere există trei secţii: *A*, *B*, *C* care au făcut următoarea convenţie privind aprobarea proiectelor:

a. Dacă *A* nu participă la aprobarea unui proiect, atunci nu participă nici *B*;
b. Dacă *A* participă, atunci participă atât *B* cât şi *C*.

Aflaţi dacă această convenţie obligă sau nu secţia *C* să participe la orice proiect la care participă şi secţia *B*.

Problema 9.4.4. [26]

Se dau premisele:

a. Niciun om care merge la o recepţie nu uită să-şi pieptene părul.
b. Nimeni nu este prezentabil dacă este neglijent.
c. Fumătorii de opium nu sunt stăpâni pe sine.
d. Toţi cei care-şi piaptănă părul sunt prezentabili.
e. Nu se poartă mănuşi albe dacă nu se merge la o recepţie.
f. Un om este întotdeauna neglijent dacă nu este stăpân pe sine.

Să se deducă din premisele de mai sus că: „Fumătorii de opium nu poartă mănuşi albe". Ce alte concluzii se mai pot obţine?

Problema 9.4.5. [19]

Trei doamne: *a*, *b*, *c*, cu feţele mânjite, stau într-un compartiment de tren şi toate trei râd. Deodată *a* are o licărire: „Dumnezeule! Probabil că şi eu am ceva ridicol." Poate oare *a* să se convingă că are motiv să se alarmeze?

Problema 9.4.6. [38]

Pe trei borcane de compot, unul cu cireşe, altul cu vişine, iar al treilea cu cireşe şi vişine, toate etichetele au fost puse greşit. Scoţând un fruct dintr-un singur borcan, să se precizeze conţinutul fiecăruia.

Problema 9.4.7. [26]

Se consideră mulţimea de premise:

a. Niciun mamifer înaripat nu are pene.
b. Toate animalele cu pene au aripi.
c. Toate bipedele sunt fie păsări, fie mamifere, dar nu sunt şi una şi alta.

Deduceţi de aici afirmaţia:
„Toate bipedele cu pene sunt păsări."

Problema 9.4.8. [25]

Sculptorul Albu, violonistul Negrea şi pictorul Roşcău sunt prieteni.

- Este interesant că unul dintre voi are păr cărunt, altul roşcat iar eu negru, dar la niciunul culoarea părului nu corespunde cu numele – observă unul din ei.
- Ai dreptate, răspunde Albu.

Ce culoare avea părul fiecăruia?

Problema 9.4.9. [31]

Cunoscutul explorator I.St.Ţul, înaintând anevoie în jungla nepătrunsă, urma cursul năvalnic al unui afluent necunoscut al fluviului Ucazali. Se aştepta ca, din clipă în clipă, să se întâlnească cu indienii triburilor *Capaxoih* ori *Hioxapac*, care trăiau în bună vecinătate prin preajma râului. Ce ştia, în definitiv, curajosul explorator despre cele două triburi? Nu ştia nimic. Cunoştea doar că *capaxoihii* erau mândri de obârşia lor, se lăudau că fac parte din acest trib, iar orice vorbă ieşită din gura lor era adevărul adevărat. În schimb, *hioxapacii* minţeau de îngheţau apele. Orice cuvânt al lor era minciună.
Ajuns la o cataractă, exploratorul se trezi deodată faţă în faţă cu trei indieni. Să fie *capaxoihi*? Să fie *hioxapaci*? – se întreba exploratorul. Sau or fi amestecaţi? Şi deoarece cunoştea toate dialectele vorbite şi nevorbite din Amazonia, el se adresă primului indian:

- Ce eşti, *capaxoih* sau *hioxapac*?

Indianul răspunse îndată. Dar din pricina zgomotului cataractelor, exploratorul n-a putut auzi răspunsul. Atunci se apropie mai mult de cei trei indieni şi (ştiind că acestora nu le place să fie întrebaţi de două ori acelaşi lucru) se adresă celui de-al doilea indian:

- Ce-a spus primul că este?
- A spus că este *capaxoih*, preciză acesta.
- Dar oare într-adevăr este *capaxoih*? insistă exploratorul.
- Da, răspunse indianul.

Socotind că l-a chestionat îndeajuns pe cel de-al doilea indian, exploratorul se adresează celui de-al treilea:

- Mă pot încrede în cele spuse de-al doilea indian?
- Nu, zise scurt cel de-al treilea indian.

I. St. Ţul nu mai întrebă nimic, socotind că scurta discuţie cu cei trei indieni îi este suficientă pentru a şti din ce trib face parte fiecare dintre ei.

Cum a raţionat exploratorul?

Problema 9.4.10. [6]

La o casă de bani au acces trei persoane: şeful şi cei doi casieri.

Concepeţi un sistem de siguranţă astfel încât şeful să poată avea acces şi singur, iar cei doi casieri numai împreună.

Problema 9.4.11. [26]

Condițiile de înmânare a unei polițe de asigurare precizează că această poliță nu poate fi subscrisă decât de persoanele care îndeplinesc una din condițiile:

a. Să fi subscris la polița 19 și să fie bărbat căsătorit.
b. Să fi subscris la polița 19 și să fie persoană căsătorită cu vârsta sub 25 ani.
c. Să nu fi subscris la polița 19 și să fie femeie căsătorită.
d. Să fie bărbat sub 25 ani.
e. Să fie persoană căsătorită cu vârsta peste 25 ani.

Exprimați cât mai concis ansamblul condițiilor de mai sus.

9.5. Indicații și soluții

Problema 9.1.1.

Nu toate proprietățile care au loc pentru conectivele „și" și „sau" au loc pentru conectivele „nand" și „nor". Astfel, subpunctele 2 și 4 se verifică, celelalte proprietăți nu au loc.

Problema 9.1.2.

Există trei modele pentru subpunctul 1, două modele pentru 2, trei modele pentru 3, șapte modele pentru 4, cinci modele pentru 5, trei modele pentru 6, cinci modele pentru 7 și patru modele pentru 8.

Problema 9.1.5.

Se vor exprima toate celelalte conective din mulțimea tuturor conectivelor $\{\neg,\wedge,\vee,\rightarrow,\leftrightarrow,\uparrow,\downarrow,\oplus\}$ utilizând doar conectivele specificate. Pentru subpunctele 6, 7 și 8, se va utiliza echivalența logică $U \oplus T \equiv \neg U$.

Problemele 9.1.6, 9.1.7 și 9.1.9.

Cu ajutorul FNC se demonstrează tautologiile, iar cu FND contradicțiile.

Problema 9.1.8.

Subpunctul 1 are cinci modele; 2 are patru modele; 3 are patru modele; 4 are cinci modele; 5 are două modele; 6 are cinci modele; 7 are patru modele; 8 are trei modele.

Problema 9.1.10.

Se va utiliza Definiția 1.3 a relației de consecință logică.

Problema 9.1.11.

Dacă apar conectivele „și" și „sau", acestea se vor exprima utilizând implicații și negații.

2. Se utilizează A1;
6. Se va demonstra mai întâi $\vdash U \wedge V \rightarrow V$, utilizând A1 și A3;
8. Se aplică A3.

Problemele 9.1.12 și 9.1.13.

Se vor scrie conectivele „și" și „sau" utilizând doar negațiile și implicațiile, după care se va aplica inversa teoremei deducției, identificându-se cât mai multe ipoteze. Cu ajutorul câtorva teoreme ajutătoare (vezi indicațiile de mai jos), utilizând modus ponens, se demonstrează concluzia, iar apoi, aplicând teorema deducției, se va demonstra și formula inițială.

Pentru problema 9.1.12.

1. Se va demonstra mai întâi $\vdash U \wedge V \rightarrow U$, utilizând teorema din Exemplul 1.10. $\vdash \neg U \rightarrow (U \rightarrow V)$ și axioma A3;
2. Se utilizează silogismul și teorema $\vdash (U \rightarrow V) \rightarrow (\neg U \rightarrow V) \rightarrow V$.
3. Se va demonstra mai întâi $\vdash U \wedge V \rightarrow V$ utilizând axiomele A1 și A3.
4. Se va utiliza regula separării premiselor.
5. Se va demonstra mai întâi $\vdash U \wedge V \rightarrow U$, utilizând teorema din Exemplul 1.10. $\vdash \neg U \rightarrow (U \rightarrow V)$ și axioma A3, iar apoi se demonstrează $\vdash U \wedge V \rightarrow V$ utilizând axiomele A1 și A3; și se va aplica regula reunirii premiselor.

Pentru subpunctele 6 și 7, se va utiliza $\vdash U \rightarrow (V \rightarrow U \wedge V)$, ce se demonstrează utilizând forma corespunzătoare a consecinței 1 a teoremei de deducție: $\vdash U \rightarrow ((U \rightarrow V) \rightarrow V)$ și axioma A3.

8. Înainte de a aplica inversa teoremei deducției, se aplică regula separării premiselor, care poate fi utilizată și mai târziu în demonstrație.

Pentru problema 9.1.13.

1. Se va demonstra mai întâi $\vdash U \wedge V \rightarrow U$, utilizând teorema din Exemplul 1.10 $\vdash \neg U \rightarrow (U \rightarrow V)$ și axioma A3, apoi se demonstrează $\vdash U \wedge V \rightarrow V$ utilizând axiomele A1 și A3.
2. Se vor utiliza axiomele A1 și A3.
3. Se va utiliza $\vdash U \rightarrow (V \rightarrow U \wedge V)$, ce se demonstrează utilizând forma corespunzătoare a consecinței 1 a teoremei de deducție: $\vdash U \rightarrow ((U \rightarrow V) \rightarrow V)$ și axioma A3.

4. Se demonstrează $\vdash U \wedge V \to U$, utilizând teorema din Exemplul 1.10. $\vdash \neg U \to (U \to V)$ şi axioma A3, apoi teorema $\vdash U \wedge V \to V$ utilizând axiomele A1 şi A3, iar apoi se va utiliza $\vdash U \to (V \to U \wedge V)$, ce se demonstrează utilizând forma corespunzătoare a consecinţei 1 a teoremei de deducţie: $\vdash U \to ((U \to V) \to V)$ şi axioma A3.

8. Se va utiliza axioma A3.

Problema 9.1.14.

Subpunctul 1 are şase modele, 2 are şase modele, 3 are patru modele; 4 are patru modele, 5 are patru modele, 6 are cinci modele, 7 are trei modele, 8 are şase modele.

Problemele 9.1.15 şi 9.1.16.

Se va ţine cont de faptul că metoda tabelelor semantice este o metodă de demonstrare prin respingere.

Problema 9.1.17.

Secvenţele de la subpunctele 4, 6 şi 8 au loc, celelalte nu au loc.

Problema 9.1.19.

Toate relaţiile de consecinţă logică au loc.

Problema 9.1.24.

Cele două strategii input şi unit sunt echivalente. Deşi pot fi identificate rafinări care sunt doar input sau doar unit, dacă se găseşte una, se va găsi şi cealaltă. Există însă cazuri în care, doar prin rezoluţia liniară simplă, se derivează clauza vidă (subpunctele 2, 4, 6 şi 8), chiar dacă există în mulţimea iniţială clauze unitate.

Problema 9.2.1.

Pe lângă regulile de deducţie de la calculul predicatelor, silogismul şi axioma A4; la subpunctele 5 şi 6 se utilizează axioma A1, la subpunctele 3 şi 7 axioma A3, iar la subpunctul 2 axiomele A3 şi A5.

Problema 9.2.2.

1. O soluţie are două constante, două funcţii şi trei predicate.
2. Sunt patru predicate.
3. O soluţie are două variabile, o funcţie şi un predicat.
4. Sunt două funcţii şi două predicate.
5. O soluţie are patru variabile, o funcţie şi două predicate.
6. O soluţie are trei variabile, o funcţie şi două predicate.
7. O soluţie are o variabilă şi trei predicate, dar, are o expresie nebanală.
8. O soluţie are trei variabile şi trei predicate.

Problema 9.2.3.

La subpunctul 6 se va ţine cont de faptul că punctele pot să se afle toate pe aceeaşi dreaptă.

Problema 9.2.4.

Se va căuta o interpretare în care premisa implicaţiei să fie adevărată, iar concluzia implicaţiei să fie falsă.

Problema 9.2.6.

Se redenumesc variabilele legate astfel încât să fie distincte. În final, se obţin câte două forme prenexe/Skolem/clauzale. Pentru subpunctele 1, 2, 5 şi 8 cea „mai complicată" formă Skolem se obţine utilizând o funcţie Skolem de o variabilă; iar pentru celelalte subpuncte, cea „mai complicată" formă Skolem se obţine utilizând o funcţie Skolem de două variabile.

Problema 9.2.7.

Se rezolvă similar cu problema precedentă. Se alege doar o formă prenexă, după care se aplică skolemizarea şi apoi se aduce la forma clauzală.

Problema 9.2.8.

La fiecare subpunct, una, maxim două perechi sunt unificabile.

Problema 9.2.10.

Metoda lui Herbrand este o metodă de demonstrare prin respingere. Înainte de a aplica metoda, formula trebuie adusă la o formă Skolem. Se recomandă să se utilizeze cea mai simplă formă Skolem.

Problema 9.2.11.

Trebuie redenumite toate variabilele legate, astfel încât să fie distincte.

Problema 9.2.12.

Subpunctele 1 şi 8 nu au loc, iar toate celelalte se verifică.

Problema 9.2.13.

Metoda tabelelor semantice este o metodă prin respingere.

Problema 9.2.14.

Sunt teoreme formulele de la subpunctele 4 şi 8. Pentru subpunctele 1, 2, 3 şi 5 sunt teoreme doar implicaţiile de la stânga la dreapta, iar pentru subpunctele 6 şi 7 sunt teoreme doar implicaţiile de la dreapta spre stânga.

Problema 9.2.16.

Formulele de la subpunctele 1, 2, 5 şi 7 nu sunt teoreme, iar celelalte formule sunt teoreme.

Problema 9.2.20.

Sunt valide echivalențele de la subpunctele 1 şi 2, pentru 3, 4, 6 şi 8 sunt valide doar implicațiile de la stânga la dreapta, iar pentru 5 şi 7 sunt valide doar implicațiile de la dreapta spre stânga.

Problema 9.2.21.

Formulele de la subpunctele 5, 6 şi 7 nu sunt teoreme.

Problema 9.3.1.

Toate funcțiile sunt în primul caz al algoritmului de simplificare.

Problema 9.3.2.

Toate funcțiile sunt în cazul al doilea al algoritmului de simplificare şi au câte două forme simplificate.

Problema 9.3.3.

Toate funcțiile sunt în primul caz al algoritmului de simplificare.

Problema 9.3.4.

Se aduce mai întâi funcția la FCD şi abia apoi se simplifică. Toate funcțiile sunt în primul caz de simplificare.

Problema 9.3.5.

Toate subpunctele sunt în cazul al treilea de simplificare şi au câte două forme simplificate.

Problema 9.3.6.

Toate subpunctele sunt în cazul al doilea de simplificare şi au câte două forme simplificate.

Problema 9.3.7.

Se aduce funcția la FCD şi apoi se simplifică. Toate funcțiile sunt în cazul al doilea de simplificare şi au o unică formă simplificată. Rezolvarea prin metoda lui Moisil e uşoară dacă se foloseşte absorbția.

Problema 9.3.8.

Se înlocuiesc operatorii „nand" şi „nor" cu expresiile echivalente care conțin doar $\wedge, \vee, \neg$. Se aduce funcția la FCD, se simplifică şi se redesenează circuitul.

Problema 9.4.2.

Se notează cu *a*/*b* propozițiile: „martorul a/b are dreptate" şi cu *x* propoziția analoagă relativă la preşedinte. Problema de rezolvat se reduce la a identifica anti-modelele formulei: $(x \wedge a \rightarrow \neg b) \rightarrow x \vee a \wedge b$.

Problema 9.4.3.

Se notează cu *A*, *B* şi *C* propozițiile care afirmă că secția corespunzătoare participă la un proiect. Problema se reduce la a demonstra că are loc: $\neg A \rightarrow \neg B, A \rightarrow B \wedge C \models B \rightarrow C$.

Problema 9.4.4.

Se transformă afirmațiile din limbaj natural în formule predicative folosind cuantificatorii adecvați şi se va demonstra apoi concluzia folosind o metodă de demonstrare dorită. Pentru partea a doua a problemei, se va încerca demonstrarea altor concluzii, sau, aplicând modus ponens, se vor deduce cât mai multe consecințe pornind de la premisele existente.

Problema 9.4.5.

Se identifică predicatele: „*x* crede că *y* râde de *z*", „*x* râde" şi „*x* este mânjit pe față". Ştim că o persoană nu râde dacă aceasta crede că se râde de ea.

Problema 9.4.6.

Se vor utiliza predicate de forma: „În borcanul cu eticheta *x* este conținutul *z*" şi se va scoate un singur fruct din borcanul unde sigur nu sunt amestecate fructele.

Problema 9.4.7.

Se transformă afirmațiile din limbaj natural în formule predicative folosind cuantificatorii şi simbolurile predicative unare: *mamifer*, *pasăre*, *cu_aripi*, *cu_ pene*, *biped*. Se aduc premisele şi negația concluziei la formele clauzale, apoi se aplică rezoluția predicativă sau metoda lui Herbrand.

Problema 9.4.8.

Se vor utiliza predicate de forma: "*x* are părul de culoarea *y*".

Problema 9.4.9.

Se transformă afirmațiile din limbaj natural în limbaj logic.
Se vor asocia celor două tipuri de indieni două funcții: una care păstrează semnificația răspunsurilor, iar una care o schimbă. În final, se descoperă că primii doi indieni sunt *capaxoili*, iar ultimul *hioxapac*.

Problema 9.4.10.

Se construieşte o funcție de trei variabile, care se va simplifica, iar în final se va trasa circuitul.

Problema 9.4.11.

Se notează afirmațiile simple cu variabile propoziționale iar apoi se va simplifica expresia condițională obținută.

INDEX ALFABETIC

BIBLIOGRAFIE

1. **Ben-Ari, M.**, *Mathematical Logic for Computer Science*, Editura Springer, 2001.
2. **Benzaken, C.**, *Systemes Formels. Introduction a la Logique*. Editura Masson, 1991.
3. **Bibel, W.**, *Automated Theorem Proving*, View Verlag, Braunschweig, second edition, 1987.
4. **Boian, F. M.**, *De la aritmetică la calculatoare*, Presa Universitară Clujeană, 1996.
5. **Bonatti, P., Olivetti, N.**, *Sequent Calculi for Propositional Nonmonotonic Logics*, ACM Trans. Comput. Log., 2002, pag. 226-278.
6. **Both, N.**, *Algebra logicii cu aplicații*, Editura Dacia, Cluj-Napoca, 1984.
7. **Both, N.**, *Capitole Speciale de Logică Matematică*, Universitatea Babeș-Bolyai, Cluj-Napoca, 1994.
8. **Chang, C.L, Lee, R.C.**, *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, 1973.
9. **Cocan, M., Pop, B.**, *Bazele Matematice ale Sistemelor de Calcul*, Editura Albastră, Cluj-Napoca, 2001.
10. **Delahaye, J.P.**, *Outils Logiques pour l'Intelligence Artificielle*, Editura Eyrolls, 1986.
11. **Dreben, B., Goldfarb, W.**, *The Decision Problem: Solvable Classes of Quantificational Formulas*, Addison-Wesley, 1979.
12. **Duffy, D.A.**, *Principles of Automated Theorem Proving*, John Wiley & Sons, 1991.
13. **Dumitrescu, D.**, *Principiile Inteligenței Artificiale*, Editura Albastră, Cluj-Napoca, 2002.
14. **Fitting, M.**, *First-order logic and Automated Theorem Proving*, Springer Verlag, 1990.
15. **Floarea, A., Boangiu, A.**, *Inteligența Artificială*, Universitatea Tehnică București, 1994.
16. **Genesereth, M.R., Nilsson, N.J.**, *Logical Foundations of Artificial Intelligence*. Morgan Kaufman, 1992.
17. **Hsu, J. Z.**, *Computer logic. Design Principles and Applications*. Springer-Verlag, New York, 2002.
18. **Livovschi, L.**, *Circuite cu contacte de relee*, Editura Academiei Republicii Socialiste România, 1968.
19. **Littlewood, J.E.**, *Varietăți matematice*, București, 1969.
20. **Loveland, D.W.**, *Automated Theorem Proving: A Logical Basis*, North Holland, Amsterdam, 1978.
21. **Lupea, M.**, *Raționament nemonoton prin logici implicite*, Teza doctorat, Universitatea Babeș-Bolyai, Cluj-Napoca, 2002.
22. **Lupea, M.**, *Axiomatization of credulous reasoning in rational default logic*, Studia Universitas Babeș-Bolyai, Informatica, LII(1), pag:101-111, 2007.
23. **Malița, M., Mircea M.**, *Bazele Inteligenței Artificiale, Vol.I. Logici propoziționale*, Editura Tehnică, București, 1987.
24. **Mihiș, A.D., Chisăliță-Crețu C., Mihăilă C., Șerban C.**, *BOOFS - a tool that supports simplifying conditional expressions using boolean functions simplification methods*, *Proceedings of International Conference of Mathematics & Informatics, ICMI45*, Bacău, Septembrie 18-20, 2006, Studii și cercetări științifice, nr.16 – 2006 Supplement, Universitatea din Bacău, Facultatea de Stiințe, Seria Matematica, ISN 1224-2519.
25. **Moșcenski, V.A.**, *Lecții po matematiceskoi logike*, Minsk, 1973.
26. **Nasin, P.**, *Circuite logice și automatizări secvențiale*, Editura Tehnică, București 1967.
27. **Nilsson, N.J.**, *Principles of Artificial Intelligence*, Tioga, Palo Alto, Morgan Kaufmann, 1980.
28. **Paulson, L.C.**, *Logic and Proof*, Universitatea Cambridge, 2000, www.cl.cam.ac.uk/teaching/2000/LogicProof/notes.pdf.
29. **Popa, C.**, *Logica predicatelor*, Editura Hyperion, București, 1992.
30. **Possega, M.**, *Deduction Systems*, Institute of Informatics, 2002, curs on-line
31. **Rădulescu, V.**, *Revanșa minții*, Editura Militară, București, 1974
32. **Reeves S., Clarke, M.**, *Logic for Computer Science*, Eddison-Wesley, 1990.
33. **Rich, E.**, *Artificial Inteligence*, Mac Graw-Hill, NewYork, 1983.
34. **Risch, V., Schwind, C.B.**, *Tableau-Based Characterization and Theorem Proving for Default Logic*, Journal of Automated Reasoning, Vol 13, Nr. 2, 1994, pag. 223-242.
35. **Robinson J.A.**, *Logic, Form and Function. The Mechanization of Deductive Reasoning*. University Press, Edinburg, 1979.
36. **Roul de Palma**, *Algebra binară a lui Boole și aplicațiile ei în informatică*, Editura Tehnică, București, 1976.
37. **Russel, S., Norwig, P.**, *Artificial Intelligence*, Prentice Hall, 1995.
38. **Rusu, E.**, *Cum gândim și rezolvăm 200 de probleme*, Editura Albastros, București, 1972.
39. **Schwind, C.B.**, *A tableaux-based theorem prover for a decidable subset of default logic*. 10-th International Conference on Automated Deduction. Lecture Notes in A.I. 449, 1990.
40. **Smullyan, R.M.**, *First-Order Logic*, Springer Verlag, Berlin, 1968. Revised Edition, Dover Press, New York, 1996.
41. **State, L.**, *Elemente de Logică Matematică și Demonstrarea Automată a Teoremelor*. Universitatea București, 1989.
42. **Tarski, A.**, *Introduction to Logic*, Oxford University Press, 1976.
43. **Thayse, A.**, editor: *From Standard Logic to Logic Programming*, John Wiley & Sons, vol1 (1989), vol2 (1989), vol3 (1990).
44. **Tătar, D.**, *Bazele Matematice ale Calculatoarelor*, Lito, Universitatea Babeș-Bolyai, Cluj-Napoca, 1999.
45. **Tătar, D.**, *Inteligența Artificială: Demonstrare Automată de Teoreme și NLP*, Editura Microinformatica, Cluj-Napoca, 2001.
46. **Tîrnoveanu, M.**, *Elemente de logică matematică, vol.1. Logica propozițiilor bivalente*, Editura didactică și pedagogică, București, 1964.

BIBLIOTECA CENTRALĂ UNIVERSITARĂ „LUCIAN BLAGA"
MATEMATICĂ
CLUJ-NAPOCA