Wartości logiczne

Za *zdanie* będziemy uważać dowolne stwierdzenie, o którym można powiedzieć, że jest albo prawdziwe, albo fałszywe, i które nie może być jednocześnie i prawdziwe, i fałszywe.

Powiedzenie „studenci miewają trudności ze zdaniem egzaminu” jest zdaniem (jest bowiem albo prawdziwe, albo nie, i powiedzenie o nim, że jest prawdziwe lub fałszywe, ma sens), natomiast sformułowanie „czy logika jest trudna?” — zdaniem nie jest, bowiem nie można sensownie wypowiedzieć się o prawdziwości pytania.

W zgodzie z intuicją będziemy przypisywać zdaniom wartość logiczną *prawdy* lub *fałszu*. System taki nazywamy *logika dwuwartościowa*.

Przykład 1. Rozpatrzmy następujące wypowiedzenie „w pewnej wiosce jeden z jej mieszkańców goli wszystkich tych i tylko tych jej mieszkańców, którzy nie golą się sami”.

Czy to wypowiedzenie (będące w sensie gramatyki języka polskiego zdaniem oznajmującym) jest zdaniem w sensie podanej wyżej definicji? Na pozór tak.

Spróbujmy jednak odpowiedzieć na pytanie, kto goli owego golarza. Jeżeli goli się on sam, to nie może się sam golić, bo goli on tylko tych, którzy nie golą się sami. Jeśli jednak nie goli się sam, to goli się sam, bo właśnie goli on właśnie tych, którzy nie golą się sami. Rozumowanie to dowodzi tezy, że powyższa wypowiedź nie jest ani prawdziwa, ani fałszywa, stwierdzenie to nie jest zdaniem i nie ma żadnego sensu.

Formuły logiczne

Formuły rachunku zdań tworzy się z danych zdań, łącząc je za pomocą spójników zdaniowych „i”, „lub”, „nie”, „jeżeli” oraz „wtedy i tylko wtedy”. Każdy sensowny napis, który da się utworzyć ze zmiennych i spójników będziemy uważać za formułę rachunku zdań. Formuły będziemy oznaczać literami $\varphi, \psi, \ldots$

Wartość logiczna formuł „nie φ”, „φ i ψ”, „φ lub ψ”, „jeżeli φ, to ψ”, „φ wtedy i tylko wtedy, gdy ψ” zależy tylko od wartości logicznej formuł φ i ψ, a nie od ich sensu.

Negacja. Na oznaczenie słowa „nie” przyjmiemy znak negacji $\neg$. Zgodnie z intuicją wyrażenie $\neg\varphi$ jest prawdziwe, gdy φ jest fałszywe, natomiast jest fałszywe, gdy φ jest prawdziwe.

Koniunkcja. Na oznaczenie słowa „i” przyjmiemy znak koniunkcji $\wedge$. Zgodnie z intuicją wyrażenie $\varphi \wedge \psi$ (zwane iloczynem logicznym zdań φ i ψ) jest prawdziwe, gdy zarówno φ, jak i ψ (zwane *czynnikami*), są prawdziwe.

Alternatywa. Na oznaczenie słowa „lub” przyjmiemy znak alternatywy $\vee$. Zgodnie z intuicją wyrażenie $\varphi \vee \psi$ (zwane sumą logiczną zdań φ i ψ) jest prawdziwe, gdy co najmniej jedno ze zdań φ, ψ (zwanych *składnikami*) jest prawdziwe.

Implikacja. Na oznaczenie słowa „jeżeli” przyjęliśmy znak implikacji $\Rightarrow$. Wyrażenie „jeżeli φ, to ψ” jest fałszywe, gdy φ (zwane *poprzednikiem*) jest prawdziwe, a ψ (zwane *następnikiem*) — fałszywe; w pozostałych trzech przypadkach jest prawdziwe. Mówimy, że *z prawdy tylko prawda wynika, natomiast z fałszu zarówno prawda, jak i fałsz wynika*.

Intuicyjnie implikacja jest prawdziwa, gdy poprzednik daje się wywnioskować z poprzednika. Jednakże implikacja nie jest tożsama z wnioskowaniem. Zdania wchodzące w skład implikacji mogą nie mieć ze sobą żadnego związku, ponadto implikacja jest prawdziwa nawet wtedy, gdy oba te zdania są fałszywe. Wnioskowanie natomiast polega na wyprowadzeniu nowego zdania prawdziwego z innego zdania, uznanego wcześniej za prawdziwe.

Zapis $\varphi \Rightarrow \psi$ oznacza, że φ jest warunkiem wystarczającym dla ψ, natomiast ψ jest warunkiem koniecznym dla φ.

Równoważność. Na oznaczenie frazy „wtedy i tylko wtedy, gdy” przyjemiemy znak równoważności $\Leftrightarrow$. Zgodnie z intuicją wyrażenie $\varphi \Leftrightarrow \psi$ (zwane *równoważnością zdań φ i ψ*) jest prawdziwe wtedy, gdy oba jego człony mają taką samą wartość logiczną (tzn. oba równocześnie są prawdziwe lub fałszywe). Zapis $\varphi \Leftrightarrow \psi$ oznacza, że φ jest warunkiem koniecznym i wystarczającym dla ψ.

Definicja indukcyjna zbioru formuł rachunku zdań

Nie będziemy obecnie zajmować się zdaniami w rodzaju „Premier lubi Prezydenta" albo „Prezydent boi się Premiera". Zastąpimy je zmiennymi zdaniowymi p, q, r, ... z nieskończonego zbioru V; zmienne te będą mogły mieć wartość 1 (odpowiadającej prawdzie) lub 0 (odpowiadającej fałszowi).

Wprowadzimy teraz (w miejsce intuicyjnej) formalną definicję indukcyjną zbioru formuł rachunku zdań. Dostarczy nam ona wygodnego narzędzia służacego do dowodzenia twierdzeń dotyczących rachunku zdań.

Definicja 1. Niech $V = \{p, q, r, \ldots\}$ będzie nieskończonym zbiorem zmiennych zdaniowych, zaś $\Sigma = \{\bot, \top, \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow\}$ zbiorem spójników. Zbiorem $\mathcal{F}$ formuł rachunku zdań będziemy nazywać najmniejszy zbiór napisów złożony ze zmiennych ze zbioru V, spójników z Σ i nawiasów, spełniający następujące warunki:

1. $V \subseteq \mathcal{F}$
2. (a) $\bot, \top \in \mathcal{F}$,
   (b) jeżeli $\varphi \in \mathcal{F}$, to $\neg\varphi \in \mathcal{F}$,
   (c) jeżeli $\varphi, \psi \in \mathcal{F}$, to każda z formuła postaci $\varphi \circ \psi$, gdzie $\circ \in \{\wedge, \vee, \Rightarrow, \Leftrightarrow\}$, należy do $\mathcal{F}$.

Sens tej definicji jest następujący: z 1. wynika, że napis złożony z pojedynczej zmiennej jest formułą. Zasada wyrażona w punkcie 2. ilustruje sposób budowania formuły złożonej z jednej lub dwóch formuł oraz wybranego spójnika logicznego; dodatkowo twierdzi się, że napis złożony ze spójnika fałszu $\bot$ jest formułą.

Przykład 2. Niech $V = \{p, q, r, \ldots\}$. W zgodzie z punktem 1 powyższej definicji twierdzimy, że wyrażenia p, q, r są formułami rachunku zdań. Z 2b wynika, że zapis $\neg q$ jest forumułą. Korzystając z 2c stwierdzamy, że również napis $p \wedge (\neg q)$ jest formułą. Stosując ponownie 2c otrzymujemy kolejne formuły: $(p \wedge (\neg q)) \Rightarrow r$ lub np. $(p \wedge (\neg q)) \Leftrightarrow \bot$ itd.

Łączność

W niektórych formułach (np. powyższy przykład) musimy stosować nawiasy, celem uniknięcia niejednoznaczności w sposobie rozbioru formuły. Niekiedy nawiasy opuszczamy, zakładając następującą kolejność wiązania (od najsilniejszego do najsłabszego): $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$ i przyjmując, że $\wedge$ i $\vee$ łączą w lewo, tj. $p \vee q \vee s$ znaczy $(p \vee q) \vee s)$, zaś $\Rightarrow$ i $\Leftrightarrow$ łączą w prawo, tj. $p \Rightarrow q \Rightarrow s$ znaczy $p \Rightarrow (q \Rightarrow s)$.

Powiedzenie, że dany spójnik łączy w lewo (w prawo) można zrozumieć w ten sposób, że mając ciąg zdań połączonych danym spójnikiem zdaniowym nawiasy należy zacząć stawiać od lewej (prawej) strony.

Przykład 3. Niech φ oznacza formułę $p \vee \neg q \vee r \wedge s$. Negacja wiąże najsilniej, możemy więc zapisać tę formułę w następujący sposób: $p \vee (\neg q) \vee r \wedge s$. Drugim co do siły wiązania spójnikiem jest funktor koniunkcji — otrzymujemy $p \vee (\neg q) \vee (r \wedge s)$. Funktor alternatywy łączy w lewo, zatem ostatecznie φ jest równoważne $((p \vee (\neg q)) \vee (r \wedge s))$.

Wartość logiczna formuł

Definicja 2. Zbiór wartości logicznych $\mathcal{B} = \{0, 1\}$ zawiera dwa elementy: 0 — na określenie fałszu i 1 — na określenie prawdy. Wartościowanie zmiennych to funkcja $\sigma\colon\ V \to \mathcal{B}$.

Intuicyjnie — wartościowanie to przypisanie wartości logicznych (0 lub 1) poszczególnym zmiennym.

Zdefiniujemy obecnie funkcję w_σ przyporządkowującą formułom, dla danego wartościowania zmiennych σ, jedną z wartości logicznych prawdy lub fałszu. Jeżeli wartościowanie zmiennych jest ustalone w danym kontekście i taki zapis nie prowadzi do nieporozumień, będziemy pomijać symbol σ i zamiast w_σ pisać w.

Definicja funkcji w będzie przypominać sposób konstrukcji zbioru formuł r.z. Niech V oznacza zbiór zmiennych występujących w formule, natomiast σ niech będzie wartościowaniem tych zmiennych. Oczywiście dla każdej zmiennej $v \in V$ zachodzi $w(v) = \sigma(v)$. Nie jest także zaskoczeniem, że dla każdego wartościowania σ mamy $w(\perp) = 0$.

Tabele prawdy

Sposób obliczania wartości logicznej formuł złożonych obrazują poniższe tabele:

$w(\bot) = 0$

$w(\top) = 1$

$w(\varphi)$	$w(\neg\varphi)$
0	1
1	0

$w(\varphi)$	$w(\psi)$	$w(\varphi \wedge \psi)$
0	0	0
0	1	0
1	0	0
1	1	1

$w(\varphi)$	$w(\psi)$	$w(\varphi \vee \psi)$
0	0	0
0	1	1
1	0	1
1	1	1

$w(\varphi)$	$w(\psi)$	w$(\varphi \Rightarrow \psi)$
0	0	1
0	1	1
1	0	0
1	1	1

$w(\varphi)$	$w(\psi)$	$w(\varphi \Leftrightarrow \psi)$
0	0	1
0	1	0
1	0	0
1	1	1

Arytmetyka w logice

Zauważmy, że jeśli symbole 0 i 1 potraktować jako liczby naturalne, to:

$$w(\neg\varphi) = 1 - w(\varphi)$$
$$w(\varphi \wedge \psi) = w(\varphi) \cdot w(\psi)$$
$$w(\varphi \vee \psi) = w(\varphi) + w(\psi)$$

$+$ oznacza symbol zwykłego dodawania arytmetycznego, z wyjątkiem sumy $1 + 1$, którą uznajemy za równą 1.

Obserwacja ta pozwala zrozumieć genezę nazw iloczynu i sumy logicznej.

Spełnialność i tautologie

Mówimy, że formuła jest:

- *spełniona*, jeżeli dla danego wartościowania ma wartość logiczną 1,
- *spełnialna*, jeżeli istnieje wartościowanie, przy którym ma wartość logiczną 1,
- *tautologią*, jeśli ma wartość 1 dla każdego wartościowania zmiennych (lub — co na jedno wychodzi — nie istnieje wartościowanie, dla którego ma wartość 0),
- *niespełnialna* (albo *fałszywa*), jeżeli ma wartość 0 dla każdego wartościowania.

O formule spełnionej przez dane wartościowanie będziemy niekiedy mówić, że jest prawdziwa przy tym wartościowaniu, natomiast formuła nie spełniona będzie fałszywa.

Przykład 4.
$p \vee \neg p$ jest tautologią

$p \wedge \neg p$ jest niespełnialna

$p \Rightarrow q$ jest spełnialna, i jest spełniona przy wartościowaniu σ_1 takim, że $\sigma_1(p) = 0$ i $\sigma_1(q) = 1$, natomiast nie jest spełniona przy wartościowaniu σ_2 takim, że $\sigma_2(p) = 1$ i $\sigma_2(q) = 0$.

Metoda zerojedynkowa sprawdzania tautologii

Istnieje nieskończenie wiele tautologii, z których wiele znano już w czasach starożytnych. Stanowią one w logice odpowiednik np. tożsamości arytmetycznych czy trygonometrycznych, znanych z matematyki.

Aby dowieść, że dana formuła jest tautologią, można poddać ją sprawdzeniu metodą zerojedynkową. W tym celu należy sprawdzić, czy wartość logiczna badanej formuły jest równa 1 dla każdego możliwego wartościowania zmiennych w niej występujących.

Przykład 5. Rozważmy następującą tautologię

$$(p \Leftrightarrow q) \Leftrightarrow ((p \Rightarrow q) \wedge (q \Rightarrow p))$$

opisującą związek równoważności z implikacją.

p	q	$p \Leftrightarrow q$	$p \Rightarrow q$	$q \Rightarrow p$	$(p \Rightarrow q) \wedge (q \Rightarrow p)$	$(p \Leftrightarrow q) \Leftrightarrow ((p \Rightarrow q) \wedge (q$
0	0	1	1	1	1	1
0	1	0	1	0	0	1
1	0	0	0	1	0	1
1	1	1	1	1	1	1

„Szybka" metoda sprawdzania tautologii

Sprawdzenie, czy dana formuła jest tautologią, w podany wyżej sposób co prawda jest proste i zawsze prowadzi do celu, ale jest procesem żmudnym. Dlatego zamiast pracowicie sprawdzać wartość logiczną formuły dla każdego wartościowania zmiennych, korzystniej jest rozważyć takie wartościowania, dla których formuła staje się fałszywa. Jeśli takie wartościowania istnieją, badana formuła w oczywisty sposób nie jest tautologią. Natomiast jeśli założenie istnienia wartościowania, dla którego formuła jest fałszywa, prowadzi do sprzeczności, stwierdzamy, że takie wartościowanie nie istnieje, a zatem badana formuła musi być tautologią.

Przykład 6. Rozważmy formułę $[(p \vee q) \wedge \neg p] \Rightarrow q$. Załóżmy, że istnieje wartościowanie σ, dla którego jest ona fałszywa. Ponieważ jest ona implikacją, to jest to możliwe tylko wówczas, gdy jej poprzednik jest prawdziwy, a następnik fałszywy. Zatem $w(q) = 0$ (skąd $\sigma(q) = 0$) oraz $w((p \vee q) \wedge \neg p) = 1$. Ale — wobec $\sigma(q) = 0$ — $p \vee q$ jest równoważne p i poprzednik przyjmuje postać koniunkcji $p \wedge \neg p$, która jest zawsze fałszywa. Założyliśmy jednak uprzednio, że poprzednik jest prawdziwy. Nie może on być jednak i prawdziwy, i fałszywy przy tym samym wartościowaniu. Zatem założenie istnienia wartościowania, dla którego badana formuła jest fałszywa prowadzi do sprzeczności.
Oznacza to, że wartościowanie takie nie może istnieć i formuła jest tautologią.

Tautologie rachunku zdań

$(\neg\neg p) \Leftrightarrow p$	podwójne przeczenie
$p \vee \bot \Leftrightarrow p$	
$p \vee \top \Leftrightarrow \top$	
$p \wedge \bot \Leftrightarrow \bot$	
$p \wedge \top \Leftrightarrow p$	($\top$ oznacza zdanie prawdziwe)
$p \vee \neg p \Leftrightarrow \top$	
$p \wedge \neg p \Leftrightarrow \bot$	
$(p \vee q) \Leftrightarrow (q \vee p)$	przemienność
$(p \wedge q) \Leftrightarrow (q \wedge p)$	
$(p \Leftrightarrow q) \Leftrightarrow (q \Leftrightarrow p)$	
$[(p \vee q) \vee r] \Leftrightarrow [p \vee (q \vee r)]$	łączność
$[(p \wedge q) \wedge r] \Leftrightarrow [p \wedge (q \wedge r)]$	
$[p \vee (q \wedge r)] \Leftrightarrow [(p \vee q) \wedge (p \vee r)]$	rozdzielność
$[p \wedge (q \vee r)] \Leftrightarrow [(p \wedge q) \vee (p \wedge r)]$	

$\neg(p \vee q) \Leftrightarrow (\neg p \wedge \neg q)$	prawa De Morgana
$\neg(p \wedge q) \Leftrightarrow (\neg p \vee \neg q)$	
$p \Rightarrow q \Leftrightarrow \neg q \Rightarrow \neg p$	
$p \Rightarrow q \Leftrightarrow \neg p \vee q$	
$[(p \Rightarrow r) \wedge (q \Rightarrow r)] \Leftrightarrow [(p \vee q) \Rightarrow r]$	
$[(p \Rightarrow q) \wedge (p \Rightarrow r)] \Leftrightarrow [(p \Rightarrow (q \wedge r)]$	
$(p \Leftrightarrow q) \Leftrightarrow [(p \Rightarrow q) \wedge (q \Rightarrow p)]$	zamiana równoważności na implikacje
$(p \Rightarrow q) \Leftrightarrow [(p \wedge \neg q) \Rightarrow \perp]$	sprowadzenie do sprzeczności

Przykład

Wyobraźmy sobie, że podczas pewnej kampanii wyborczej politycy Olek, Józek i Kazik wygłosili następujące oświadczenia:

Olek: Józek kłamie.
Józek: Kazik kłamie.
Kazik: Olek kłamie.

Co z tego wynika? Spróbujmy zapisać te wypowiedzi w języku logiki, oznaczając przez P_O, P_J, P_K fakty, że Olek, Józek, i Kazik zawsze mówią prawdę. Wtedy wypowiedzi powyższych prominentów mają postać:

$$P_O \Rightarrow \neg P_J \quad \text{inaczej} \quad \neg P_O \vee \neg P_J$$

$$P_J \Rightarrow \neg P_K \quad \text{inaczej} \quad \neg P_J \vee \neg P_K$$

$$P_K \Rightarrow \neg P_O \quad \text{inaczej} \quad \neg P_K \vee \neg P_O$$

Rozważmy hipotetyczną możliwość, że Olek rzeczywiście mówi wyłącznie prawdę. Wtedy z pewnością Józek jest kłamcą, no i niestety, musimy przyznać, że przynajmniej „pomylił się” Kazik. Podobne rozumowanie możemy łatwo przeprowadzić dla każdej z tych osób, a zatem tylko jeden z nich może mieć rację i być rzeczywiście prawdomównym. Tylko nie wiemy który, jeśli w ogóle.

Interpretacje

Przypomnijmy sobie funkcję wartościowania σ przypisującą każdej zmiennej zdaniowej ze zbioru V jedną z wartości 0 lub 1.

Definicja 3. Funkcję w_σ przypisującą każdej formule wartość logiczną 0 lub 1 zgodnie z wartościowaniem σ i tabelkami prawdy definiującymi operatory logiczne, nazywamy interpretacją.

To jest tylko uzupełnienie — nowa nazwa dla funkcji w_σ, którą posługiwaliśmy się już wcześniej.

Równoważność logiczna

Definicja 4. Jeśli dla wszystkich interpretacji w dla danych formuł φ i ψ mamy $w(\varphi) = w(\psi)$ to formuły φ i ψ nazywamy równoważnymi logicznie, co zapisujemy $\varphi \equiv \psi$.

Pojęcie równoważności logicznej formuł pojawiło się już wcześniej, trochę nieformalnie. Zwróćmy uwagę w powyższej definicji, że odwołujemy się do wszystkich możliwych interpretacji, czyli dowolnych funkcji wartościowania. Zauważmy, że dla formuł rachunku zdań mamy skończoną liczbę możliwych wartościowań, które rozpatrywaliśmy w tabelkach logicznych do sprawdzania tautologii, lub równoważności logicznej formuł.

Zwróćmy uwagę na dwa pojęcia: spójnika równoważności $\Leftrightarrow$, np. w formule $p \Leftrightarrow q$, oraz pojęcia równoważności logicznej formuł $\varphi \equiv \psi$. Te pojęcia mają ze sobą związek, ponieważ formuła $\varphi \Leftrightarrow \psi$ jest tautologią wtw gdy $\varphi \equiv \psi$. Ale są między nimi istotne różnice: $\equiv$ nie jest spójnikiem, a więc $\varphi \equiv \psi$ nie jest formułą rachunku zdań, i tym samym nie ma wartości logicznej.

Modele

Przypomnijmy sobie pojęcie spełnialności, i formuł spełnialnych, niespełnialnych, oraz prawdziwych, czyli tautologii.

Definicja 5. Modelem formuły φ nazywamy każdą interpretację spełniającą tę formułę.

Zatem formuła niespełnialna nie ma modelu. Natomiast dla tautologii każda interpretacja jest modelem.

Będziemy też stosować notację $\models \varphi$ oznaczającą, że formuła φ jest tautologią.

Zbiory formuł

Będziemy się posługiwali zbiorami formuł $U = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ które będziemy traktowali jak koniunkcje wszystkich formuł ze zbioru. Wprowadzamy pojęcia modeli i niespełnialności dla zbiorów formuł:

Definicja 6. Modelem zbioru formuł $U = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ nazywamy interpretację, dla której $w(\varphi_1) = w(\varphi_2) = \ldots = w(\varphi_n) = 1$.

Zbiór U jest niespełnialny wtw gdy dla każdej interpretacji w istnieje takie i, że $w(\varphi_i) = 0$.

Zauważmy, że zgodnie z przedstawionymi definicjami pusty zbiór formuł $U = \emptyset$ musimy uznać za spełnialny, a wręcz za prawdziwy, czyli tautologię. Może to się wydawać dziwne, ale zauważmy, że zbiór formuł traktujemy jak koniunkcję, a ponieważ w prawdziwej koniunkcji wszystkie elementy muszą być prawdziwe, więc odrywanie elementów nie zmienia prawdziwości koniunkcji. Zatem nie powinno jej zmienić oderwanie ostatniego prawdziwego elementu, które tworzy „pustą” koniunkcję. Pustych koniunkcji normalnie nie zapisujemy (koniunkcja jest spójnikiem dwuargumentowym), ale pusty zbiór formuł oczywiście istnieje.

Własności zbiorów formuł

Oznaczmy: $U = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$

Fakt: Jeśli zbiór U jest spełnialny to zbiór $U - \{\varphi_i\}$ jest spełnialny dla dowolnego $1 \leq i \leq n$.

Fakt: Jeśli zbiór U jest spełnialny, a formuła ψ jest prawdziwa, to zbiór $U \cup \{\psi\}$ jest również spełnialny.

Fakt: Jeśli zbiór U jest niespełnialny, to dla dowolnej formuły ψ zbiór $U \cup \{\psi\}$ jest również niespełnialny.

Fakt: Jeśli zbiór U jest spełnialny, a dla pewnego $1 \leq i \leq n$ formuła φ_i jest prawdziwa, to zbiór $U - \{\varphi_i\}$ jest również spełnialny.

Fakt: Jeśli zbiór U jest niespełnialny, a dla pewnego $1 \leq i \leq n$ formuła φ_i jest prawdziwa, to zbiór $U - \{\varphi_i\}$ jest również niespełnialny.

Zauważmy jednak, że dla dwóch spełnialnych zbiorów formuł U_1 i U_2 nie jest wcale pewne, że $U_1 \cup U_2$ jest spełnialny, np. $U_1 = \{p\}$, $U_2 = \{\neg p\}$

Konsekwencje logiczne

Definicja 7. Niech U będzie zbiorem formuł, a φ dowolną formułą. Jeśli każdy model U jest jednocześnie modelem φ to mówimy, że φ jest *konsekwencją logiczną* U, co zapisujemy $U \models \varphi$.

Mówi się również czasami, że φ *wynika logicznie* z U, który bywa nazywany zbiorem *przesłanek*.

Przykład 7. Rozważmy formułę $\varphi = p \vee \neg q$. Formuła φ jest konsekwencją logiczną zarówno zbioru $U_1 = \{p\}$, czyli $U_1 \models \varphi$, jak i $U_2 = \{\neg q\}$, czyli $U_2 \models \varphi$. Jednocześnie φ nie jest konsekwencją logiczną zbioru $U_3 = \{\neg p, q\}$, a więc $U_3 \not\models \varphi$.

Zwróćmy uwagę, że stosowana poprzednio notacja dla tautologii $\models \varphi$ jest spójna z notacją wynikania logicznego, jeśli zauważymy, że wtedy $\emptyset \models \varphi$. Jest tak dlatego, że dowolna interpretacja jest modelem pustego zbioru, który jest z definicji prawdziwy, a jednocześnie dowolna interpretacja jest modelem tautologii φ. Możemy więc powiedzieć, że tautologia jest konsekwencją logiczną pustego zbioru przesłanek.

Własności wynikania logicznego

Zauważmy, że podobnie jak dla pojęć $\Leftrightarrow$ i $\equiv$ istnieje subtelny związek pomiędzy spójnikiem implikacji $\Rightarrow$ i pojęciem konsekwencji logicznej $\models$. Mamy:

Fakt: Jeśli dla pewnego $U = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ i ψ zachodzi $U \models \psi$ to formuła $\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n \Rightarrow \psi$ jest tautologią, inaczej:

$$\models \varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n \Rightarrow \psi$$

Nie możemy jednak powiedzieć, że tautologią jest $U \models \psi$ ponieważ nie jest to formuła logiczna, a $\models$ nie jest spójnikiem logicznym.

Fakt: Jeśli $U \models \varphi$ to dla dowolnej formuły ψ zachodzi: $U \cup \{\psi\} \models \varphi$

Fakt: Jeśli $U \models \varphi$ a ψ jest formułą prawdziwą, to zachodzi: $U - \{\psi\} \models \varphi$

Teorie

Definicja 8. Zbiór formuł $\mathcal{T}$ nazywamy *teorią* jeśli jest on *zamknięty* na konsekwencje (albo wynikanie) logiczne. Zbiór formuł $\mathcal{T}$ jest *zamknięty* na konsekwencje logiczne wtw gdy dla wszystkich formuł φ zachodzi zależność: jeśli $\mathcal{T} \models \varphi$, to $\varphi \in \mathcal{T}$. Elementy teorii $\mathcal{T}$ nazywamy *twierdzeniami* tej teorii.

Teorią $\mathcal{T}(U)$ zbioru formuł U nazywamy zbiór $\mathcal{T}(U) = \{\varphi \,|\, U \models \varphi\}$.

Fakt: Dla dowolnego zbioru formuł U teoria tego zbioru $\mathcal{T}(U)$ jest teorią.

Przykład 8. Załóżmy, że mamy jakąś teorię $\mathcal{T}$ i formuła $\varphi = p$ należy do tej teorii. Zauważmy, że następująca formuła musi również należeć do tej teorii: $p \vee \neg p$, jak również: $p \vee q$, jak również: $p \wedge (p \vee q)$, jak również: $(p \wedge \neg p) \Rightarrow q$, jak również: $(q \vee \neg q) \Rightarrow p$, jak również wiele innych.
W rzeczywistości, każda teoria jest nieskończonym zbiorem formuł.

Teoria zbioru formuł $\mathcal{T}(\{p\})$ na pewno zawiera się w teorii $\mathcal{T}$, ale nie musi być jej dokładnie równa. Na przykład, formuła $\psi = q$ z całą pewnością nie jest elementem $\mathcal{T}(\{p\})$, natomiast może być elementem teorii $\mathcal{T}$, na przykład w przypadku, gdy $\mathcal{T} = \mathcal{T}(\{p, q\})$.

Wyprowadzanie formuł

Tautologie i formuły niespełnialne rachunku zdań nie są interesujące same w sobie. Jednak pojęcie konsekwencji logicznych i teorii zbioru formuł pozwalają formułować ciekawe zagadnienia. Na przykład możemy opisać jakąś hipotetyczną rzeczywistość zbiorem formuł, i zadawać sobie pytania czy konkretne fakty będą spełnione w tej rzeczywistości, czyli czy są twierdzeniami teorii tego zbioru formuł. Sprawdzanie czy formuła jest twierdzeniem danej teorii nazywamy *problemem decyzyjnym* w logice, a procedura umożliwiająca rozstrzygnięcie tego problemu nazywa się *procedurą decyzyjną*.

Zgodnie z przedstawioną metodologią, sprawdzanie czy dana formuła jest twierdzeniem teorii pewnego zbioru formuł powinno wziąć pod uwagę wszystkie interpretacje będące modelami tego zbioru formuł. Jednak takie podejście jest często kłopotliwe i niepraktyczne.

Zamiast sprawdzać interpretacje i spełnialność stosuje się w logice inne podejście, zwane wyprowadzaniem (albo dowodzeniem) formuł.

Reguły wnioskowania

Twierdzenia matematyczne mają na ogół postać implikacji. Dowodzi się ich następująco: mając pewien zbiór zdań, zwanych *założeniami* lub *przesłankami*

$$\varphi_1, \varphi_2, \ldots, \varphi_n$$

uznaje się je za prawdziwe i dołącza do nich nowe zdanie

$$\psi$$

zwane *wnioskiem* lub *konkluzją*, które wynika z nich zgodnie z prawami logiki.

Dla formalnego wprowadzenia dowodów logicznych używa się *reguł wnioskowania*, postaci:

$$\frac{\varphi_1, \varphi_2, \ldots, \varphi_n}{\psi}$$

Znaczenie reguły wnioskowania jest takie, że jeśli wiemy, że prawdziwe są formuły $\varphi_1, \varphi_2, \ldots, \varphi_n$ to możemy również uznać za prawdziwą formułę ψ.

Reguły wnioskowania nazywa się również *regułami dowodzenia*, ponieważ służą one do budowy dowodów twierdzeń.

Pojęcie dowodu

Przykładową regułą wnioskowania jest:

$$\frac{\varphi, \varphi \Rightarrow \psi}{\psi}$$

zwana *regułą odrywania* (*modus ponens*). Sens jej jest następujący: jeżeli uznajemy za prawdziwe zdanie φ oraz implikację $\varphi \Rightarrow \psi$, to możemy też uznać za prawdziwe zdanie ψ.

Definicja 9. Załóżmy, że dany jest pewien zbiór formuł Δ zwany zbiorem przesłanek, oraz formuła ψ. *Dowodem* formuły ψ jest ciąg $\varphi_1, \varphi_2, \ldots, \varphi_n, \psi$, składający się z formuł, z których każda spełnia jeden z warunków:

1. jest jedną z przesłanek,
2. jest tautologią,
3. została otrzymana w wyniku użycia jednej z reguł wnioskowania zastosowanej do formuł leżących na lewo od niej w dowodzie.

Poprawność reguł wnioskowania

Zmierzamy do tego, żeby za pomocą reguł wnioskowania określać wynikanie logiczne formuł, aby nie trzeba było w tym celu sprawdzać interpretacji i modeli. Jednak w tym celu reguły wnioskowania muszą być w jakimś sensie dobre(?).

Definicja 10. Regułę wnioskowania

$$\frac{\varphi_1, \varphi_2, \ldots, \varphi_n}{\psi}$$

nazywamy *poprawną* wtedy i tylko wtedy, gdy formuła

$$\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n \Rightarrow \psi$$

jest tautologią.

Łatwo zauważymy, że reguła wnioskowania *modus ponens* jest poprawna.

Czego jeszcze potrzeba do efektywnego dowodzenia twierdzeń?

Weźmy przykładowy zbiór przesłanek $\Delta = \{p, q\}$, i rozważmy następującą prostą formułę: $p \wedge q$. Jeśli odwołamy się do interpretacji i modeli, to łatwo stwierdzimy, że każdy model zbioru Δ jest równocześnie modelem formuły $p \wedge q$, a więc mamy $\Delta \models p \wedge q$.

Chcemy skonstruować dowód formuły $p \wedge q$, czyli ciąg formuł zbudowany według reguł budowania dowodów, i kończący się formułą: $p \wedge q$. Jednak tego nie da się osiągnąć przy użyciu dotychczas wprowadzonych pojęć. Zgodnie z definicją dowodu, aby się to udało, nasza dowodzona formuła $p \wedge q$ musiałaby być albo:

1. jedną z przesłanek, ale nie jest,
2. tautologią, ale też niestety nie jest,
3. otrzymana w wyniku użycia jednej z reguł wnioskowania zastosowanej do formuł leżących na lewo od niej w dowodzie.

Potrzebujemy więcej reguł wnioskowania. Na przykład następująca poprawna reguła wnioskowania, zwana regułą wprowadzania koniunkcji pozwoliłaby rozwiązać przedstawiony problem: $\dfrac{\varphi, \psi}{\varphi \wedge \psi}$

Formuły równoważne

Zwróćmy uwagę, że konstruując dowody formuł, nie możemy korzystać z faktu równoważności formuł, na przykład, wiedząc, że $p \Rightarrow q$ nie możemy udowodnić formuły $\neg p \vee q$, ani zresztą na odwrót. Dlaczego?

Bo równoważność formuł była własnością opartą o niespełnialność, czyli o interpretacje. Po prostu wiedzieliśmy, że formuła $p \Rightarrow q$ jest prawdziwa zawsze, i tylko wtedy, gdy prawdziwa jest również $\neg p \vee q$, więc z punktu widzenia wartości prawdy lub fałszu możemy zawsze jedną z tych formuł zastąpić drugą.

Dowodzenie twierdzeń musi być oparte o reguły wnioskowania. Aby korzystać z takiej własności jak wyżej, musimy wprowadzić regułę wnioskowania:

$$\frac{\varphi \Rightarrow \psi}{\neg\varphi \vee \psi}$$

i

$$\frac{\neg\varphi \vee \psi}{\varphi \Rightarrow \psi}$$

(nie ma reguł wnioskowania działających w obie strony).

Wprowadzanie prawdy i fałszu do formuł

Wyobraźmy sobie, że mamy zbiór przesłanek $\Delta = \{p, \neg q, r\}$. Z tego zbioru formuł możemy łatwo udowodnić formułę r, ale sięgając znów do pojęć wynikania logicznego mamy również własność: $\Delta \models r \vee q$. Łatwo to sprawdzić za pomocą odpowiednich tabelek prawdy, ale można też sobie wytłumaczyć w ten sposób: skoro zakładamy, że q jest fałszywe, to możemy dołączyć fałszywy element alternatywy do prawdziwej formuły, i będzie nadal prawdziwa. Podobnie: $\Delta \models r \wedge p$.

Jakie reguły wnioskowania są potrzebne, aby osiągnąć to samo za pomocą dowodzenia?

$$\frac{\varphi}{\varphi \vee (\psi \wedge \neg\psi)}$$

$$\frac{\varphi}{\varphi \wedge (\psi \vee \neg\psi)}$$

Systemy dowodzenia twierdzeń

W logice zdefiniowano szereg alternatywnych *systemów dowodzenia twierdzeń* wprowadzających różne reguł wnioskowania (zestawy reguł), a także pewne formuły, których nie potrzeba udowadniać, zwane *aksjomatami*. Dokładna konstrukcja tych systemów jest dość skomplikowana i w tym wykładzie zapoznamy się tylko pobieżnie z dwoma przykładowymi takimi systemami.

Fakt, że w danym systemie wnioskowania z jego aksjomatów i danego zbioru przesłanek Δ można wywieść zapisujemy: $\Delta \vdash \varphi$. Jeśli jesteśmy w stanie wywieść formułę φ jedynie za pomocą reguł wnioskowania i aksjomatów danego systemu, to zapisujemy to jako: $\vdash \varphi$.

Stosuje się również wariant tej notacji odnoszący się do kokretnego systemu dowodzenia, np. jeśli formułę φ można udowodnić w *hilbertowskim* systemie dowodzenia to zapisujemy to: $\vdash_H \varphi$. Analogicznie, wywodliwość formuły w innym popularnym systemie dowodzenia *gentzenowskim* można zapisać: $\vdash_G \varphi$.

System naturalnej dedukcji

wprowadzenie negacji: $\dfrac{\neg\neg\varphi}{\varphi}$

eliminacja negacji: $\dfrac{\neg\neg\varphi}{\varphi}$

wprowadzenie koniunkcji: $\dfrac{\varphi, \psi}{\varphi \wedge \psi}$

eliminacja koniunkcji: $\dfrac{\varphi \wedge \psi}{\varphi, \psi}$

wprowadzenie alternatywy: $\dfrac{\varphi}{\varphi \vee \psi, \psi \vee \varphi}$

eliminacja alternatywy: $\dfrac{\varphi \Rightarrow \psi, \omega \Rightarrow \psi, \varphi \vee \omega}{\psi}$

wprowadzenie implikacji: $\dfrac{\neg\varphi \vee \psi}{\varphi \Rightarrow \psi}$

eliminacja implikacji: $\dfrac{\varphi \Rightarrow \psi}{\neg\varphi \vee \psi}$

modus ponens: $\dfrac{\varphi, \varphi \Rightarrow \psi}{\psi}$

modus tollens: $\dfrac{\varphi \Rightarrow \psi, \neg\psi}{\neg\varphi}$

Przykład 9. Mamy zbiór aksjomatów $\Delta = \{p, \neg q\}$. Czy jest dowodem ciąg formuł: $p, \neg q, \neg q \vee p, q \Rightarrow p$?

Odpowiedź: Tak, jest to poprawny dowód formuły $q \Rightarrow p$.

W dowodzie mogą pojawić się formuły ze zbioru aksjomatów, oraz formuły uzyskane z reguł wnioskowania. Zauważmy, że pierwsze dwie formuły dowodu (p i $\neg q$) przepisane są żywcem ze zbioru aksjomatów Δ. Nie jest konieczne przepisywanie w dowodzie formuł ze zbioru aksjomatów, ale jest to dozwolone, a przy dużym zbiorze aksjomatów może uczynić dowód bardziej przejrzystym i łatwiejszym do sprawdzenia.

Trzecia formuła ($\neg q \vee p$) daje się uzyskać z reguły wprowadzania alternatywy przyjmując $\varphi = \neg q$ i $\psi = p$. Zauważmy, że przesłanki reguły wnioskowania muszą być obecne w zbiorze aksjomatów (i w tym przypadku są), albo we wcześniejszej części dowodu.

Czwarta formuła dowodu ($q \Rightarrow p$) daje się uzyskać z reguły wprowadzania implikacji jeśli przyjmiemy $\varphi = q$ i $\psi = p$. Reguła ma jedną przesłankę, i w tej roli przyjmujemy formułę trzecią dowodu, dopiero co wcześniej otrzymaną.

Ponieważ wszystkie formuły w ciągu formuł zostały zapisane zgodnie z definicją dowodu, a wiec podany ciąg jest dowodem.

Dowodzenie twierdzeń nie wprost

$$\frac{\Delta, \neg\varphi \vdash \perp}{\Delta \vdash \varphi}$$

Poprawność i pełność systemu dowodzenia

System dowodzenia nazwiemy *poprawnym* jeśli pozwala on wywodzić jedynie prawdziwe formuły, czyli jeśli: $\Delta \vdash \varphi$ to na pewno: $\Delta \models \varphi$.

Jeśli system dowodzenia pozwala na zbudowanie dowodu każdej formuły prawdziwej (tautologii), czyli jeśli zawsze gdy: $\Delta \models \varphi$ to również: $\Delta \vdash \varphi$, to będziemy taki system dowodzenia nazywać *pełnym*.

Fakt: istnieje pełny system dowodzenia dla rachunku zdań. Mówimy, że rachunek zdań jest *rozstrzygalny*.

Normalizacja formuł rachunku zdań

Formuły w których symbol negacji występuje przed formułą złożoną są często trudne do zrozumienia (wyrażenie „nie prawda, że liczba n dzieli się przez 2 lub przez 3" jest bardziej skomplikowane, niż równoważne mu wyrażenie „liczba n nie dzieli się przez 2 i nie dzieli się przez 3").

Fakt:
dla każdej formuły możemy znaleźć formułę jej równoważną, w której negacja występuje jedynie przed zmiennymi zdaniowymi. Na początek zauważmy, że ponieważ formuły ze spójnikami innymi niż koniunkcja, alternatywa, i negacja (czyli implikacja $\Rightarrow$, równoważność $\Leftrightarrow$, i alternatywa wykluczająca $\oplus$), można zamienić na równoważne im formuły zawierające tylko koniunkcję, alternatywę, i negację. Dalej, zgodnie z prawami de Morgana dla dowolnych formuł φ_1 i φ_2 formuły $\neg(\varphi_1 \wedge \varphi_2)$ oraz $\neg\varphi_1 \vee \neg\varphi_2$, a także formuły $\neg(\varphi_1 \vee \varphi_2)$ oraz $\neg\varphi_1 \wedge \neg\varphi_2$ są równoważne.

Definicja 11. *Literałem* nazywamy formułę, która ma postać pojedynczego symbolu zmiennej zdaniowej, np. p, albo zmiennej zaprzeczonej $\neg p$.
Literał o postaci p, dla pewnej zmiennej zdaniowej p, nazywamy pozytywnym, a literał o postaci $\neg p$ nazywamy negatywnym.

Dysjunkcyjna postać normalna DNF

Definicja 12. Formuła ma *dysjunkcyjną postać normalną* (DNF, ang. *Disjunctive Normal Form*), jeśli jest postaci

$$\bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{m_i} l_{ij} \right),$$

gdzie l_{ij} są literałami, dla $i = 1, \ldots, n$ oraz $j = 1, \ldots, m_i$. Mówiąc skrótowo, dysjunkcyjna postać normalna, to alternatywa koniunkcji literałów.

Przykład 10. Rozważmy formułę $p \wedge (q \vee r)$, która nie jest w postaci DNF. Możemy znaleźć równoważną formułę DNF przekształcając ją zgodnie z prawami rozdzielności alternatywy względem koniunkcji: $(p \wedge q) \vee (p \wedge r)$

Przykład 11. Formuła $\varphi = (p \wedge \neg q) \vee (\neg p \wedge q)$ jest w postaci DNF. Zauważmy, że jest ona równoważna formule $p \oplus q$:

p	q	$\neg p$	$\neg q$	p$\wedge\neg q$	$\neg p \wedge q$	φ	$p \oplus q$
0	0	1	1	0	0	0	0
0	1	1	0	0	1	1	1
1	0	0	1	1	0	1	1
1	1	0	0	0	0	0	0

Dla każdej formuły rachunku zdań istnieje równoważna jej formuła w postaci DNF. Można ją otrzymać przekształcając formułę zgodnie z prawami de Morgana i rozdzielności. Istnieje ponadto prosty schemat wyznaczania tej postaci na podstawie tabelki prawdy dowolnej formuły.

p	q	r	...	φ
0	0	0	...	0
...	...	...	...	...
0	0	1	...	1
...	...	...	...	...
0	1	0	...	1
...	...	...	...	...

Dla każdego wiersza tabeli, który ma jedynkę w kolumnie formuły φ konstruujemy koniunkcję literałów odpowiadających wszystkim zmiennym zdaniowym formuły, z negacją tylko przy zmiennych, dla których w danym wierszu występuje zero:

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

Formuła wygenerowana zgodnie z tym schematem rzadko jest jednak optymalna, tzn. na ogół istnieje krótsza postać DNF równoważna danej formule.

Koniunkcyjna postać normalna CNF

Definicja 13. *Klauzulą* nazywamy formułę postaci

$$\bigvee_{j=1}^{m} l_j,$$

gdzie l_j są literałami, dla $j = 1, \ldots, m$. Mówiąc krótko, klauzula to alternatywa literałów.

Definicja 14. Formuła ma *koniunkcyjną postać normalną* (CNF, ang. *Conjunctive Normal Form*), jeśli jest postaci

$$\bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} l_{ij} \right),$$

gdzie l_{ij} są literałami, dla $i = 1, \ldots, n$ oraz $j = 1, \ldots, m_i$, tj. gdy jest koniunkcją klauzul.

Przykład 12. Formuła $\varphi = (p \wedge q) \vee (r \wedge s)$ nie jest w postaci CNF (w istocie formuła jest w czystej postaci DNF). Korzystając kilkukrotnie z prawa rozdzielności alternatywy względem koniunkcji otrzymujemy formułę równoważną: $(p \vee r) \wedge (q \vee r) \wedge (p \vee s) \wedge (q \vee s)$

Przekształcanie formuł do postaci CNF

Fakt: Dla każdej formuły rachunku zdań istnieje formuła równoważna w postaci CNF. Algorytm przekształcania dowolnej formuły na równoważną w postaci CNF:

1. Usuń wszystkie spójniki logiczne z wyjątkiem negacji, koniunkcji i alternatywy, tworząc równoważne formuły wykorzystujące tylko te formuły. Na przykład:

$$(p \Rightarrow q) \equiv (\neg p \vee q)$$
$$(p \Leftrightarrow q) \equiv ((\neg p \vee q) \wedge (p \vee \neg q))$$

2. Przenieś wszystkie negacje do środka, korzystając z praw de Morgana:

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

3. Usuń podwójne zaprzeczenia, używając równoważności $\neg\neg p \equiv p$.
4. Zastosuj prawa rozdzielności dla usunięcia koniunkcji z wnętrza alternatyw.
5. Zastosuj łączność dla połączenia koniunkcji i alternatyw binarnych w n-arne.

Zapis formuł w postaci CNF jako zbiorów

Przykład 13. Rozważmy formułę: $(\neg p \Rightarrow \neg q) \Rightarrow (p \Rightarrow q)$ i znajdźmy dla niej formułę równoważną w postaci CNF.

$$\begin{aligned}(\neg p \Rightarrow \neg q) \Rightarrow (p \Rightarrow q) &\equiv \neg(\neg\neg p \vee \neg q) \vee (\neg p \vee q)\\ &\equiv (\neg\neg\neg p \wedge \neg\neg q) \vee (\neg p \vee q)\\ &\equiv (\neg p \wedge q) \vee (\neg p \vee q)\\ &\equiv (\neg p \vee \neg p \vee q) \wedge (q \vee \neg p \vee q)\end{aligned}$$

Definicja 15. Formułą w *postaci klauzulowej* nazywamy zbiór zbiorów literałów, gdzie zbiory literałów (traktowanych jako alternatywy) będą reprezentowały klauzule, a cały zbiór zbiorów (traktowany jako koniunkcja) będzie reprezentował całą formułę.

Taka notacja formuł jako zbiorów podkreśla niezależność wartości logicznej formuły w postaci CNF od kolejności występowania w niej klauzul, a w klauzulach literałów, i od ewentualnych powtórzeń literałów w klauzulach, albo identycznych klauzul w formule.

Postać klauzulowa formuły z powyższego przykładu jest następująca:
$\{\{\neg p, \neg p, q\}, \{q, \neg p, q\}\} = \{\{\neg p, q\}\}$.

Puste klauzule i zbiory klauzul

Definicja 16. Pojedynczy literał będziemy uważać za klauzulę, którą nazywamy *klauzulą unarną*. Ponadto, pusty zbiór literałów będziemy uważać za *klauzulę pustą*, i oznaczać: $\Box$.

Fakt: klauzula pusta ($\{\} = \Box$) jest niespełnialna (fałszywa), to znaczy jest równoważna formule $\bot$. Wynika to z prostego uogólnienia tabelki prawdy logicznej dla alternatywy n-arnej — taka alternatywa jest spełniona wtedy i tylko wtedy gdy przynajmniej jeden z jej elementów (literałów) jest spełniony. Intuicyjnie: alternatywa jest łatwiej spełnialna, gdy ma więcej elementów, czyli im mniej ma elementów tym trudniej ją spełnić.

Fakt: formuła pusta ($\{\} = \emptyset$) jest tautologią (formułą prawdziwą), to znaczy jest równoważna formule $\top$. Wynika to z prostego uogólnienia tabelki prawdy logicznej dla koniunkcji n-arnej — taka koniunkcja jest spełniona wtedy i tylko wtedy gdy wszystkie jej elementy (klauzule) są spełnione. Intuicyjnie: koniunkcja jest łatwiej spełnialna, gdy ma mniej elementów.

Uwaga: formuła $\{\}$ jest prawdziwa, ale klauzula $\{\}$ jest fałszywa.
Ponadto fałszywa jest formuła $\{\{\}\} = \{\Box\}$.

Reguła i metoda rezolucji

Definicja 17. Następująca reguła wnioskowania dla klauzul jest nazywana *regułą rezolucji*:

$$\frac{\{\varphi_1, \ldots, \varphi_n, \omega\}, \{\psi_1, \ldots, \psi_m, \neg\omega\}}{\{\varphi_1, \ldots, \varphi_n, \psi_1, \ldots, \psi_m\}}$$

Klauzule wejściowe rezolucji nazywamy *kolidującymi* ze względu na literał ω. Klauzula wynikowa rezolucji nazywana jest *rezolwentą* klauzul wejściowych.

Definicja 18. *Metodą rezolucji* nazywamy następujący algorytm generowania klauzul z początkowego zbioru klauzul $S = S_0$. W kolejnym i-tym kroku algorytmu wybieramy wcześniej jeszcze niewybraną parę klauzul kolidujących C_1, C_2, tworzymy rezolwentę C klauzul C_1 i C_2, oraz zbiór klauzul $S_i = S_{i-1} \cup \{C\}$. Jeśli $C = \Box$ to zatrzymaj algorytm stwierdzając, że zbiór S jest niespełnialny. Jeśli $S_i = S_{i-1}$ dla wszystkich możliwych wyborów klauzul kolidujących, to zatrzymaj algorytm stwierdzając, że zbiór S jest spełnialny.

Fakt: 1. Algorytm metody rezolucji zatrzymuje się dla każdego skończonego zbioru klauzul rachunku zdań. 2. Dla każdego niespełnialnego zbioru klauzul algorytm wygeneruje klauzulę pustą. 3. Algorytm może wygenerować klauzulę pustą tylko z niespełnialnego zbioru klauzul.

Dowodzenie metodą rezolucji

Zauważmy, że metoda rezolucji nigdy nie pozwala na stwierdzenie, że formuła jest prawdziwa, a jedynie że jest spełnialna lub niespełnialna. Zatem rezolucja jest metodą dowodzenia nie wprost, czyli jeśli chcemy udowodnić jakieś twierdzenie, to musimy utworzyć jego zaprzeczenie, i wykazać metodą rezolucji, że jest niespełnialne.

W szczególności, chcąc wykazać $\Delta \vdash \varphi$ musimy utworzyć formułę $\Delta \cup \{\neg\varphi\}$, przekształcić ją do zbioru klauzul CNF, i uruchomić metodę rezolucji.

Wniosek: metoda rezolucji jest poprawnym i pełnym systemem dowodzenia rachunku zdań.

Drzewa rezolucji

Kolejne kroki w dowodzie niespełnialności formuły metodą rezolucji można zobrazować graficznie za pomocą następującego diagramu, gdzie C_1 i C_2 są klauzulami kolidującymi, a C ich rezolwentą:

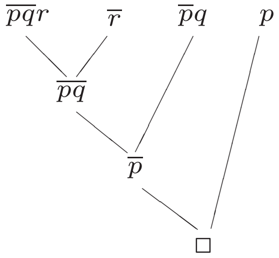
$$\begin{tikzcd} C_1 \arrow[dr, no head] & & C_2 \arrow[dl, no head] \\ & C & \end{tikzcd}$$

Definicja 19. Pełny dowód będzie się składał z zestawu takich diagramów łączących się w strukturę drzewiastą, zwane *drzewem rezolucji*.

W symbolicznym zapisie dowodów rezolucyjnych przydaje się jeszcze dalsze uproszczenie notacji. Oznaczając przez $\overline{l}$ negację literału jeśli jest on pozytywny, a pozytywną wersję tego literału jeśli jest on negatywny, możemy stosować jeszcze bardziej uproszczony zapis formuł, na przykład klauzulę:
$\{\{\neg q, \neg p, q\}, \{p, \neg p, q\}\}$ możemy zapisać w skrócie jako: $\{\overline{qp}q, p\overline{p}q\}$.

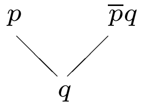
Przykłady

Rozważmy formułę $\{p, \overline{p}q, \overline{r}, \overline{pq}r\}$.



Pusta klauzula została wygenerowana. Formuła jest niespełnialna.

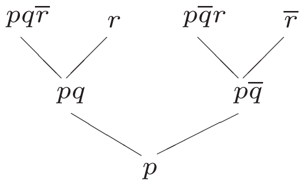
Rozważmy formułę $\{p, \overline{p}q\}$.



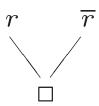
Klauzula pusta nie została wygenerowana, i nie ma żadnej innej możliwości wybrania pary klauzul kolidujących. Algorytm rezolucji zatrzyma się w tym momencie, konkludując, że wyjściowy zbiór klauzul jest spełnialny. Natomiast otrzymana rezolwenta jest pełnoprawnym twierdzeniem teorii rozważanej formuły. (Ale ta teoria jest znacznie większa, zawiera inne twierdzenia takie jak $p \vee \neg p, p \vee r$, itd.)

Przykłady (cd.)

Rozważmy formułę $\{pq\overline{r}, r, p\overline{q}r, \overline{r}\}$.



Pusta klauzula nie została wygenerowana. Czy to oznacza, że rozważana formuła jest spełnialna i z rozważanej formuły można wywieść jedynie p? Nie! Zauważmy, że to nie jest jedyne możliwe drzewo rezolucji dla tej formuły, i algorytm rezolucji nie zatrzymałby się w tym momencie, tylko dalej generował rezolwenty. Całkowicie wystarczającym dowodem niespełnialności rozważanej formuły jest natomiast następujące drzewo:



Klauzule i formuły Horna

Niech $C = \bigvee_{j=1}^{m} l_j$ będzie dowolną klauzulą i niech

$$P_C = \{j \in \{1, \ldots, m\} \mid \text{literał } l_j \text{ jest pozytywny}\},$$

$$N_C = \{j \in \{1, \ldots, m\} \mid \text{literał } l_j \text{ jest negatywny}\}$$

oraz

$$C' = \bigwedge_{j \in N_C} \overline{l_j} \;\Rightarrow\; \bigvee_{j \in P_C} l_j.$$

gdzie $\overline{l_j}$ oznacza pozytywną wersję negatywnego literału.

Fakt: formuły C i C' są równoważne.

Fakt (wniosek z powyższego): dowolną klauzulę można zapisać w postaci

$$\bigwedge_{i=1}^{m} p_i \Rightarrow \bigvee_{j=1}^{n} q_j,$$

gdzie p_i i q_j są zmiennymi zdaniowymi (bez negacji).

Definicja 20. Klauzula $\bigvee_{j=1}^{m} l_j$ jest klauzulą *hornowską* (albo klauzulą Horna), jeżeli co najwyżej jeden spośród literałów $l_1, \ldots, l_m$ jest pozytywny. Formuła ma *postać hornowską*, jeżeli jest koniunkcją klauzul hornowskich.

Klauzule i formuły Horna (cd.)

Klauzule hornowskie można więc zapisywać w postaci

$$\bigwedge_{j=1}^{m} p_j \Rightarrow q \quad \text{lub} \quad \bigwedge_{j=1}^{m} p_j \Rightarrow \bot$$

gdzie p_j oraz q są zmiennymi zdaniowymi. Nie jest to preferowana postać zapisu formuł, ponieważ będziemy dążyć do normalizacji i ograniczenia się do spójników koniunkcji, alternatywy, i negacji, a w rzeczywistości do postaci CNF, ale warto intuicyjnie rozumieć taką interpretację formuł i klauzul Horna.

Przykład 14. Dla następujących formuł podano równoważną im formułę w postaci hornowskiej:

p	p
$p \vee q$	nie ma
$p \wedge q$	$p \wedge q$
$p \wedge q \wedge r$	$p \wedge q \wedge r$
$p \wedge q \Rightarrow r$	$\neg p \vee \neg q \vee r$
$p \vee q \Rightarrow r$	$(\neg p \vee r) \wedge (\neg q \vee r)$