Logika

Logika — rachunek zdań

Materiały pomocnicze do wykładu dla Studentów

Informatyki

Wydział EAIiIB AGH

Antoni Ligęza

Materiały pomocnicze:

`http://home.agh.edu.pl/~ligeza`

Logiczna konsekwencja — podstawowe problemy logiki

Definicja 1 Logiczna konsekwencja *Formuła ψ jest logiczną konsekwencją formuły ϕ wtw. gdy dla każdej interpretacji I zachodzi*

$$\text{jeżeli} \qquad \models_I \phi \qquad \text{to} \qquad \models_I \psi. \tag{1}$$

Podstawowe problemy logiki:

- dowodzenie twierdzeń — badanie logicznej konsekwencji:

$$\Delta \models H,$$

- badanie spełnialności (SAT):

$$\text{Czy istnieje interpretacja } I: \ \models_I \Psi$$

- weryfikacja tautologii:

$$\text{Czy dla każdej interpretacji } I: \ \models_I \Psi$$

Dwa alternatywne podejścia:

- analiza możliwych interprtacji — metoda zero-jedynkowa; problem — eksplozja kombinatoryczna[1],
- wnioskowanie logiczne — wywód — za pomoca reguł logicznych zachowujących logiczną konsekwencję.

Notacja: jeżeli formuła H jest wywodliwa (wyprowadzalna) ze zbioru Δ, to zapiszemu to jako:

$$\Delta \vdash H$$

Problemy konstrukcji systemów logicznych:

$$\Delta \vdash H \qquad \text{versus} \qquad \Delta \models H$$

[1] Redukcja: drzewa decyzyjne, grafy OBDD, tablice semantyczne

Metoda zero-jedynkowa: przykład: sprawdzanie tautologii

$$\phi = ((p \Rightarrow r) \wedge (q \Rightarrow r)) \Leftrightarrow ((p \vee q) \Rightarrow r).$$

Mamy (2^3) możliwych interpretacji.

p	q	r	$p \Rightarrow r$	$q \Rightarrow r$	$(p \Rightarrow r) \wedge (q \Rightarrow r)$	$(p \vee q) \Rightarrow r$	Φ
0	0	0	1	1	**1**	**1**	**1**
0	0	1	1	1	**1**	**1**	**1**
0	1	0	1	0	**0**	**0**	**1**
0	1	1	1	1	**1**	**1**	**1**
1	0	0	0	1	**0**	**0**	**1**
1	0	1	1	1	**1**	**1**	**1**
1	1	0	0	0	**0**	**0**	**1**
1	1	1	1	1	**1**	**1**	**1**

Inna możliwość — przekształcenia równoważne:

$$\phi \equiv ((\neg p \vee r) \wedge (\neg q \vee r)) \Leftrightarrow (\neg (p \vee q) \vee r).$$

$$\phi \equiv ((\neg p \wedge \neg q) \vee r) \Leftrightarrow (\neg (p \vee q) \vee r).$$

$$\phi \equiv (\neg (p \vee q) \vee r) \Leftrightarrow (\neg (p \vee q) \vee r).$$

Kładąc: $\psi = (\neg (p \vee q) \vee r)$ widzimy, że analizowana formuła jest postaci:

$$\phi \equiv \psi \Leftrightarrow \psi,$$

Metoda zero-jedynkowa: przykład: badanie logicznej konsekwencji

$$\frac{(p \Rightarrow q) \wedge (r \Rightarrow s)}{(p \vee r) \Rightarrow (q \vee s)}$$

Kładąc:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s)$$

oraz

$$\varphi = (p \vee r) \Rightarrow (q \vee s),$$

należy sprawdzić czy:

$$\phi \models \varphi. \tag{2}$$

p	q	r	s	$p \Rightarrow q$	$r \Rightarrow s$	$(p \Rightarrow q) \wedge (r \Rightarrow s)$	$p \vee r$	$q \vee s$	$(p \vee r) \Rightarrow (q \vee s)$
0	0	0	0	1	1	**1**	0	0	**1**
0	0	0	1	1	1	**1**	0	1	**1**
0	0	1	0	1	0	**0**	1	0	**0**
0	0	1	1	1	1	**1**	1	1	**1**
0	1	0	0	1	1	**1**	0	1	**1**
0	1	0	1	1	1	**1**	0	1	**1**
0	1	1	0	1	0	**0**	1	1	**1**
0	1	1	1	1	1	**1**	1	1	**1**
1	0	0	0	0	1	**0**	1	0	**0**
1	0	0	1	0	1	**0**	1	1	**1**
1	0	1	0	0	0	**0**	1	0	**0**
1	0	1	1	0	1	**0**	1	1	**1**
1	1	0	0	1	1	**1**	1	1	**1**
1	1	0	1	1	1	**1**	1	1	**1**
1	1	1	0	1	0	**0**	1	1	**1**
1	1	1	1	1	1	**1**	1	1	**1**

Z analizy kolumn 7 i 10 wynika, że zachodzi relacja logicznej konsekwencji (brak logicznej równoważności — 7, 10, 12 i 15).

Twierdzenia o dedukcji

Twierdzenie 1 *Jeżeli $\Delta_1, \Delta_2, \ldots \Delta_n$ są formułami logicznymi (nazywanymi aksjomatami), formuła Ω (nazywana hipotezą lub konkluzją) jest ich logiczną konsekwencją wtw. gdy formuła $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \Rightarrow \Omega$ jest tautologią.*

Twierdzenie 2 *Jeżeli $\Delta_1, \Delta_2, \ldots \Delta_n$ są formułami logicznymi (nazywanymi aksjomatami), formuła Ω (nazywana hipotezą lub konkluzją) jest ich logiczną konsekwencją wtw. gdy formuła $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \wedge \neg\Omega$ jest sprzeczna.*

Problem dowodzenia twierdzeń ma postać: mając dane aksjomaty $\Delta_1, \Delta_2, \ldots \Delta_n$ uznane za prawdziwe wykazać prawdziwość hipotezy Ω. Tak więc należy wykazać, że:

$$\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \models \Omega$$

Metody dododzenia twierdzeń:

- sprawdzanie wszystkich możliwych interpretacji (wada: duża złożoność obliczeniowa),
- dowód wprost – korzystając z aksjomatów i reguł dowodzenia generujemy nowe formuły aż do uzyskania formuły Ω,
- dowodzenie tautologii – korzystając z Tw.1 dowodzimy, że formuła $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \Rightarrow \Omega$ jest tautologią,
- dowód nie wprost – to dowód twierdzenia przeciwstawnego, równoważnego danemu. Polega na dowodzeniu twierdzenia postaci $\neg\Omega \Rightarrow \neg(\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n)$.
- dowód przez sprowadzenie do sprzeczności; korzystają z Tw.2, polega na wykazaniu sprzeczności formuły: $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \wedge \neg\Omega$.

Metoda rezolucji

1. Problem:

$$\Delta \models H$$

2. Z twierdzenia o dedukcji (2) — należy wykazać, że

$$\Delta \cup \neg H$$

   jest niespełnialny.

3. Dokonać transformacji $\Delta \cup \neg H$ do postaci CNF.

4. Wykorzystując regułę rezolucji wyprowadzić zdanie puste - zawsze fałszywe.

Przykład:

1. Problem:

$$(p \Rightarrow q) \wedge (r \Rightarrow s) \models (p \vee r) \Rightarrow (q \vee s)$$

2. Z twierdzenia o dedukcji (2) — należy wykazać, że

$$[(p \Rightarrow q) \wedge (r \Rightarrow s)] \cup \neg[(p \vee r) \Rightarrow (q \vee s)]$$

   jest niespełnialny.

3. Dokonać transformacji do postaci CNF. Mamy:

$$\{\neg p \vee q, \neg r \vee s, p \vee r, \neg q, \neg s\}$$

4. Wykorzystując regułę rezolucji wyprowadzić zdanie puste - zawsze fałszywe.

Metoda rezolucji dualnej

1. Problem:

$$\Delta \models H$$

2. Z twierdzenia o dedukcji (1) — należy wykazać, że

$$\Delta \Rightarrow H$$

   jest tautologią.

3. Dokonać transformacji $\Delta \Rightarrow H$ do postaci DNF.

4. Wykorzystując regułę rezolucji dualnej wyprowadzić zdanie puste - zawsze zawsze prawdziwe.

Przykład:

1. Problem:

$$(p \Rightarrow q) \wedge (r \Rightarrow s) \models (p \vee r) \Rightarrow (q \vee s)$$

2. Z twierdzenia o dedukcji (1) — należy wykazać, że

$$[(p \Rightarrow q) \wedge (r \Rightarrow s)] \Rightarrow [(p \vee r) \Rightarrow (q \vee s)]$$

   jest tautologią.

3. Dokonać transformacji do postaci DNF. Mamy:

$$\{p \wedge \neg q, r \wedge \neg s, \neg p \wedge \neg r, q, s\}$$

4. Wykorzystując regułę rezolucji dualnej wyprowadzić zdanie puste - zawsze prawdziwe.

Metoda tablic semantycznych

Przypomnienie: atom, literał, literał pozytywny, literał negatywny, para literałów komplementarnych $\{p, \neg p\}$.

Formuła $p \wedge \neg p$ jest zawsze fałszywa. Formuła $p \vee \neg p$ jest zawsze prawdziwa.

Założenia metody tablic semantycznych:

- badamy spełnialność formuły,
- punktem startowym jest formuła w oryginalnej postaci! (nie sprowadzamy do CNF/DNF),
- analizując strukturę formuły systematycznie szukamy modelu — jego brak oznacza niespełnialność,
- do analizy tworzymy drzewo struktury (lub tablicę):
  - dla formuł koniunktywnych tworzymy (liniowo) zbiory literałów,
  - dla formuł dysjunktywnych tworzymy rozgałęzienia,
- wystąpienie pary literałów komplementarnych zamyka daną gałąź (falsyfikacja),
- brak takiej pary — dostarcza modelu (spełnialność),
- zamknięcie każdej gałęzi falsyfikacją oznacza brak modelu (niespełnialność formuły wyjściowej).

Przykład 1:

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

Przykład 2:

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

Przykłady

Przykład 1:

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

$$p, \neg q \vee \neg p$$

$$p, \neg q \qquad\qquad p, \neg p$$

Przykład 2:

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

$$p \vee q, \neg p \wedge \neg q$$

$$p \vee q, \neg p, \neg q$$

$$p, \neg p, \ \neg q \qquad\qquad q, \neg p, \neg q$$

Algorytm tablic semantycznych

Reguły przekształceń dla formuł koniunktywnych (typu α):

α	α_1	α_2
$\neg\neg A$	A	
$A_1 \wedge A_2$	A_1	A_2
$\neg(A_1 \vee A_2)$	$\neg A_1$	$\neg A_2$
$\neg(A_1 \Rightarrow A_2)$	A_1	$\neg A_2$
$A_1 \Leftrightarrow A_2$	$A_1 \Rightarrow A_2$	$A_2 \Rightarrow A_1$

Reguły przekształceń dla formuł dysjunktywnych (typu β):

β	β_1	β_2
$B_1 \vee B_2$	B_1	B_2
$\neg(B_1 \wedge B_2)$	$\neg B_1$	$\neg B_2$
$B_1 \Rightarrow B_2)$	$\neg B_1$	B_2
$\neg(B_1 \Leftrightarrow B_2)$	$\neg(B_1 \Rightarrow B_2)$	$\neg(B_2 \Rightarrow B_1)$

Algorytm tworzenia drzewa:

- Korzeń: formuła wyjściowa,
- U (dla liścia) zawiera same literały:
  - $p, \neg p \in U$ — stop/falsyfikacja; *else*
  - stop/zdefiniowano model,
- Dla formuły koniunktywnej $\alpha \in U$:

$$U' = (U - \{\alpha\}) \cup \{\alpha_1, \alpha_2\}$$

- Dla formuły dysjunktywnej $\beta \in U$ mamy rozgałęzienie:

$$U' = (U - \{\beta\}) \cup \{\beta_1\}$$

$$U'' = (U - \{\beta\}) \cup \{\beta_2\}$$

Przykład:

1. Problem:

$$(p \Rightarrow q) \wedge (r \Rightarrow s) \models (p \vee r) \Rightarrow (q \vee s)$$

2. Z twierdzenia o dedukcji (2) — należy wykazać, że

$$[(p \Rightarrow q) \wedge (r \Rightarrow s)] \cup \neg[(p \vee r) \Rightarrow (q \vee s)]$$

   jest niespełnialny.

3. *Dokonać transformacji do postaci CNF. Mamy:*

$$\{\neg p \vee q, \neg r \vee s, p \vee r, \neg q, \neg s\}$$

4. *Wykorzystując regułę rezolucji wyprowadzić zdanie puste - zawsze fałszywe.*

Problem: wykazać, że poniższa fromuła jest niespełnialna:

$$[(p \Rightarrow q) \wedge (r \Rightarrow s)] \cup \neg[(p \vee r) \Rightarrow (q \vee s)]$$

System Gentzena

Definicja 2 *System Gentzenowski jest systemem dowodzenia, w którym aksjomatami są zbiory formuł zawierające pary literałów komplementarnych (zdania prawdziwe), oraz dostępne są nastepujące reguły (schematy) dowodzenia:*

$$\frac{U \cup \{\alpha_1, \alpha_2\}}{U_1 \cup \{\alpha\}}$$

$$\frac{U_1 \cup \{\beta_1\}, \quad U_2 \cup \{\beta_2\}}{U_1 \cup U_2 \cup \{\beta\}}$$

α_1	α_2	α
A_1		$\neg\neg A_1$
$\neg A_1$	$\neg A_2$	$\neg(A_1 \wedge A_2)$
A_1	A_2	$A_1 \vee A_2$
$\neg A_1$	A_2	$A_1 \Rightarrow A_2$
$\neg(A_1 \Rightarrow A_2)$	$\neg(A_2 \Rightarrow A_1)$	$\neg(A_1 \Leftrightarrow A_2)$

β_1	β_2	β
B_1	B_2	$B_1 \wedge B_2$
$\neg B_1$	$\neg B_2$	$\neg(B_1 \vee B_2)$
B_1	$\neg B_2$	$\neg(B_1 \Rightarrow B_2)$
$B_1 \Rightarrow B_2$	$B_2 \Rightarrow B_1$	$B_1 \Leftrightarrow B_2$

Zbiór formuł postaci $\{\alpha_1, \alpha_2\}$ — stanowi dysjunkcję.

Dowody konstruktywne: Ważniejsze reguły wnioskowania (Fitch)

- AND Introduction (AI):

$$\frac{\phi_1, \ldots, \phi_n}{\phi_1 \wedge \ldots \wedge \phi_n}$$

- AND Elimination (AE):

$$\frac{\phi_1 \wedge \ldots \wedge \phi_n}{\phi_i}$$

- OR Introduction (OI):

$$\frac{\phi_i}{\phi_1 \vee \ldots \vee \phi_n}$$

- OR Elimination (OE):

$$\frac{\phi_1 \vee \ldots \vee \phi_n, \phi_1 \Rightarrow \psi, \ldots \phi_n \Rightarrow \psi}{\psi}$$

- Negation Introduction (NI):

$$\frac{\phi \Rightarrow \psi, \phi \Rightarrow \neg\psi}{\neg\phi}$$

- Negation Elimination (NE):

$$\frac{\neg\neg\phi}{\phi}$$

- Implication Introduction (II):

$$\frac{\phi \vdash \psi}{\phi \Rightarrow \psi}$$

- Implication Elimination (IE):

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

- Equivalence Introduction (EI),
- Equivalence Elimination (EE)

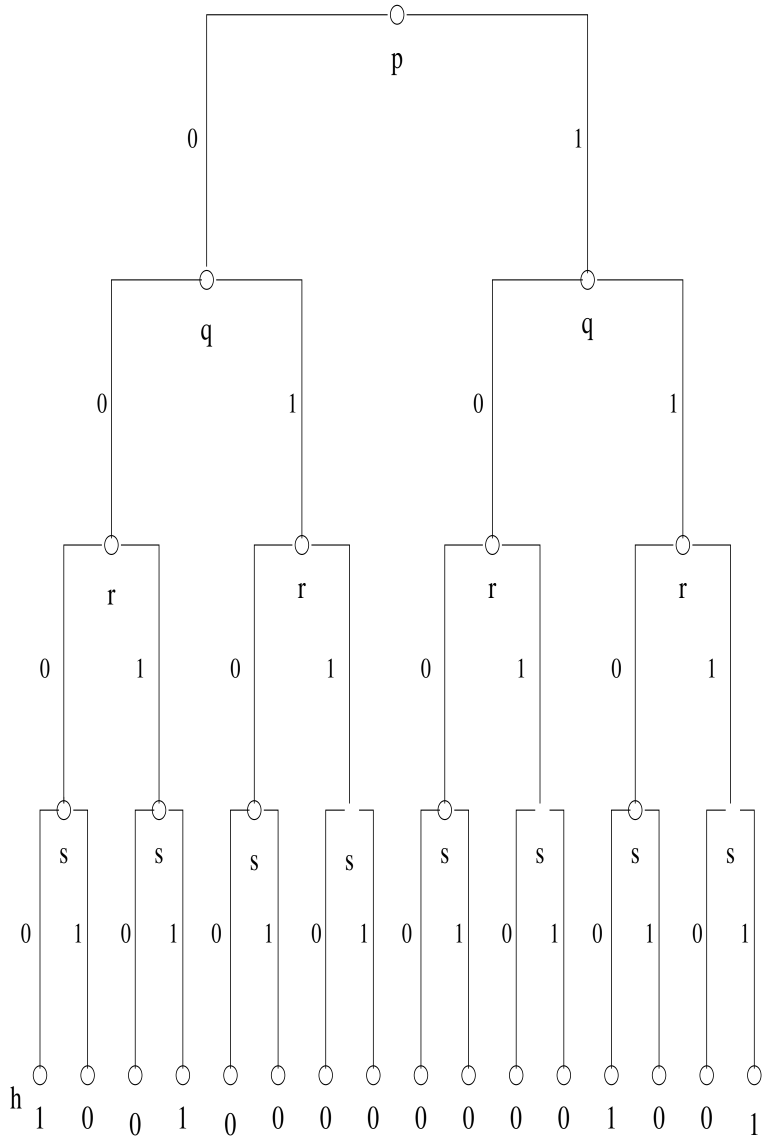
Ewaluacja formuły — metoda zero-jedynkowa

Badamy spełnialność formuły:

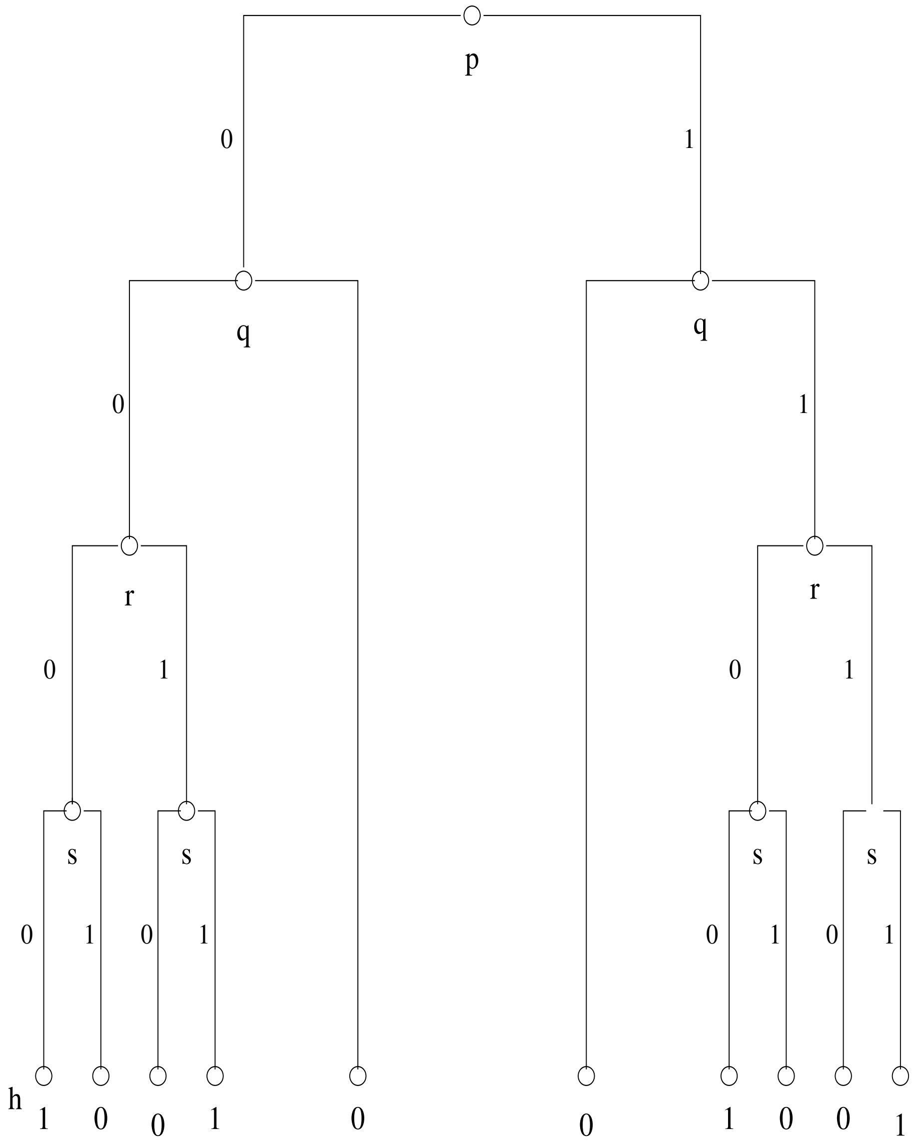
$$h \equiv (p \Leftrightarrow q) \wedge (r \Leftrightarrow s) \tag{3}$$

$RuleNo$	p	q	r	s	h
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	1

Drzewo binarne — uniwersalne narzędzie analizy formuł



Drzewo zredukowane



Ordered Binary Decision Diagrams (OBDD)

Notacja:

$$p \longrightarrow h_0, h_1$$

oznacza:

$$\textit{if } p \textit{ then } h_0 \textit{ else } h_1.$$

Definicja 3 *Reguła ekspansji Shannona*

$$\phi \equiv p \longrightarrow \phi\{p/1\}, \phi\{p/0\},$$

Przykład:

$$p \wedge q \equiv p \longrightarrow q, 0,$$

$$p \vee q \equiv p \longrightarrow 1, q$$

$$\neg p \equiv p \longrightarrow 0, 1.$$

Redukcja formuły

$$\phi = (p \Leftrightarrow q) \wedge (r \Leftrightarrow s).$$

$$\phi \equiv p \longrightarrow \phi_1, \phi_0 \tag{4}$$

$$\phi_1 \equiv q \longrightarrow \phi_{11}, 0 \tag{5}$$

$$\phi_0 \equiv q \longrightarrow 0, \phi_{00} \tag{6}$$

$$\phi_{11} \equiv r \longrightarrow \phi_{111}, \phi_{110} \tag{7}$$

$$\phi_{00} \equiv r \longrightarrow \phi_{001}, \phi_{000} \tag{8}$$

$$\phi_{111} \equiv s \longrightarrow 1, 0 \tag{9}$$

$$\phi_{110} \equiv s \longrightarrow 0, 1 \tag{10}$$

$$\phi_{001} \equiv s \longrightarrow 1, 0 \tag{11}$$

$$\phi_{000} \equiv s \longrightarrow 0, 1 \tag{12}$$

(13)

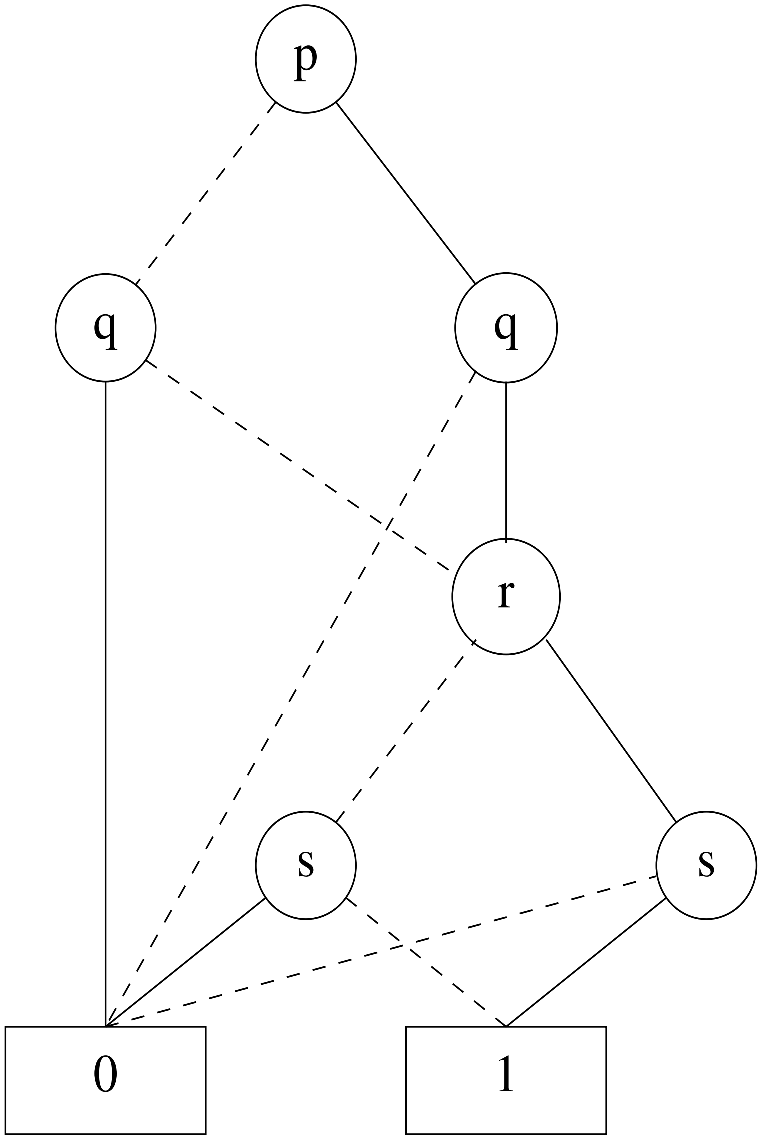
Redukcja po wykryciu powtarzających się drzew (podgrafów)

$$\begin{aligned}
\phi &\equiv p \longrightarrow \phi_1, \phi_0 && (14)\\
\phi_1 &\equiv q \longrightarrow \phi_{11}, 0 && (15)\\
\phi_0 &\equiv q \longrightarrow 0, \phi_{00} && (16)\\
\phi_{11} &\equiv r \longrightarrow \phi_{111}, \phi_{110} && (17)\\
\phi_{00} &\equiv r \longrightarrow \phi_{001}, \phi_{000} && (18)\\
\phi_{111} &\equiv s \longrightarrow 1, 0 && (19)\\
\phi_{110} &\equiv s \longrightarrow 0, 1 && (20)\\
\phi_{001} &\equiv s \longrightarrow 1, 0 && (21)\\
\phi_{000} &\equiv s \longrightarrow 0, 1 && (22)\\
& && (23)
\end{aligned}$$

przyjmuje postać:

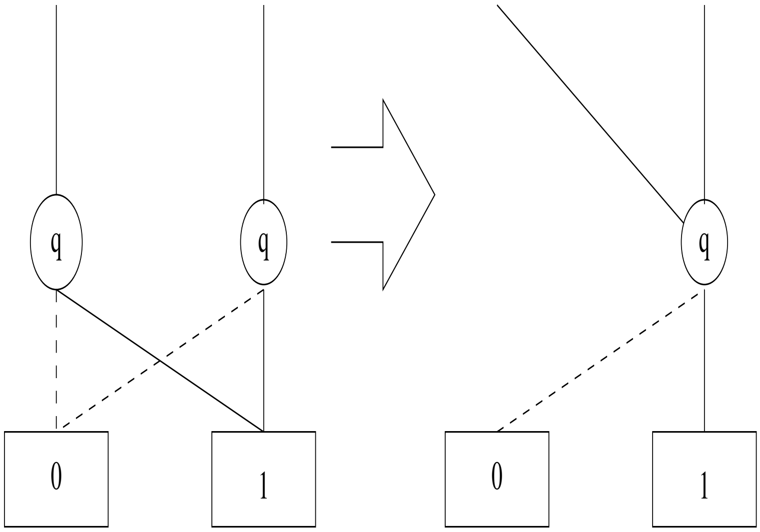
$$\begin{aligned}
\phi &\equiv p \longrightarrow \phi_1, \phi_0 && (24)\\
\phi_1 &\equiv q \longrightarrow \phi_{11}, 0 && (25)\\
\phi_0 &\equiv q \longrightarrow 0, \phi_{11} && (26)\\
\phi_{11} &\equiv r \longrightarrow \phi_{111}, \phi_{110} && (27)\\
\phi_{111} &\equiv s \longrightarrow 1, 0 && (28)\\
\phi_{110} &\equiv s \longrightarrow 0, 1 && (29)\\
& && (30)
\end{aligned}$$

Zredukowany OBDD

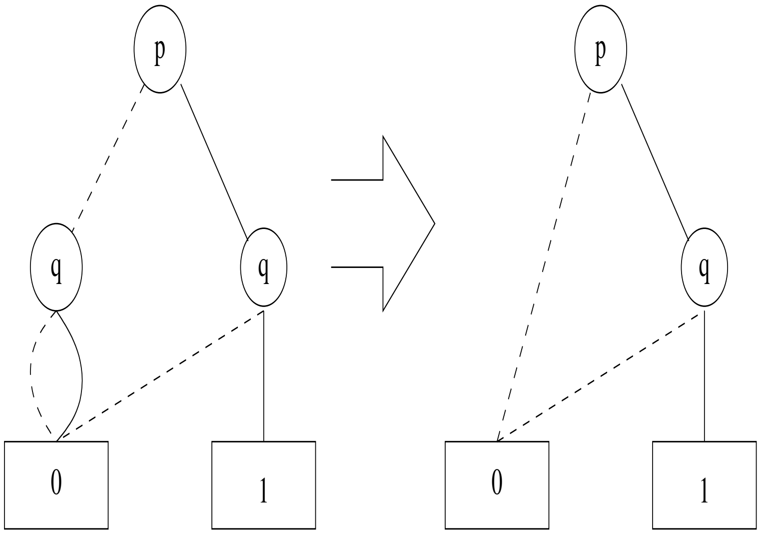


Metody redukcji

Metoda redukcji: sklejanie



Metoda redukcji: eliminacja nieistotnego węzła



Literatura

1. Mordechai Ben-Ari: Mathematical Logic for Computer Science (Logika matematyczna w informatyce). Springer-Verlag, London, 2001 (WN-T, Warszawa, 2005).
2. Kenneth A. Ross i Charles R. B. Wright: Discrete Mathematics (Matematyka dyskretna). WN PWN, 2013.
3. Antoni Ligęza: Logical Foundations for Rule-Based Systems. Springer-Verlag, Berlin, 2006. Wydawnictwo AGH, Kraków, 2005.
4. Michael R. Genesereth, Nils J. Nilsson: Logical Foundations of Artificial Intelligence. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1987.
5. Zbigniew Huzar: Elementy logiki dla informatyków. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2007.
6. Stuart Russell, Peter Norvig: Artificial Intelligence. A Modern Approach. Pearson, 2010.
7. Marek Wójcik: Zasada rezolucji. Metoda automatycznego wnioskowania. PWN, Warszawa, 1991.
8. C. L. Chang and R. C. T. Lee: Symbolic Logic and Mechanical Theorem Proving. Academic Press, 1973.
9. Ronald J. Brachman and Hector J. Levesque: Knowledge Representation and Reasoning. Morgan Kaufmann, 2004.
10. Frank van Harmelen, Vladimir Lifschitz, Bruce Porter (Eds.): *Handbook of Knowledge Representation*. Elsevier B.V., Amsterdam, 2008. `http://ii.fmph.uniba.sk/~sefranek/kri/handbook/`

Zasoby sieciowe. Kurs w Stanford

Kurs logiki on-line Stanford:

`https://www.coursera.org/course/intrologic`

1. Wikipedia-pl: `http://pl.wikipedia.org/wiki/Logika_matematyczna`
2. Wikipedia-en: `http://en.wikipedia.org/wiki/Logic`
3. AI-Lab-Prolog: `http://ai.ia.agh.edu.pl/wiki/pl:prolog:prolog_lab`
4. EIS-KRR: `http://ai.ia.agh.edu.pl/wiki/pl:dydaktyka:krr:start`
5. ALI-home: `home.agh.edu.pl/~ligeza`
6. David Poole and Allen Mackworth: Artificial Intelligence. Foundations of Computational Agents. `http://artint.info/`
7. Ulf Nilsson and Jan Maluszynski: Logic, Programming and Prolog. `http://www.ida.liu.se/~ulfni/lpp/`