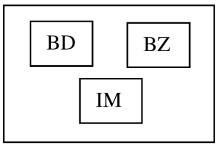
1 EXPERTNÍ SYSTÉM

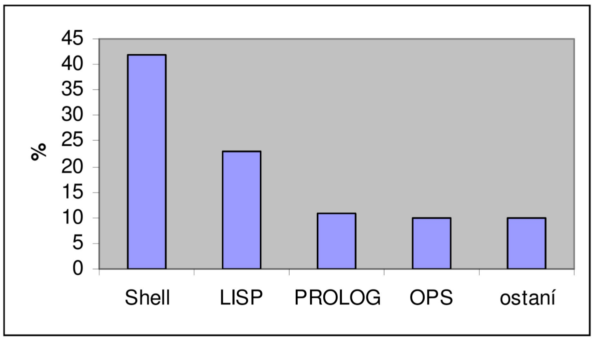
ES = počítačový program, který simuluje usuzování lidského experta (schopnost řízení)

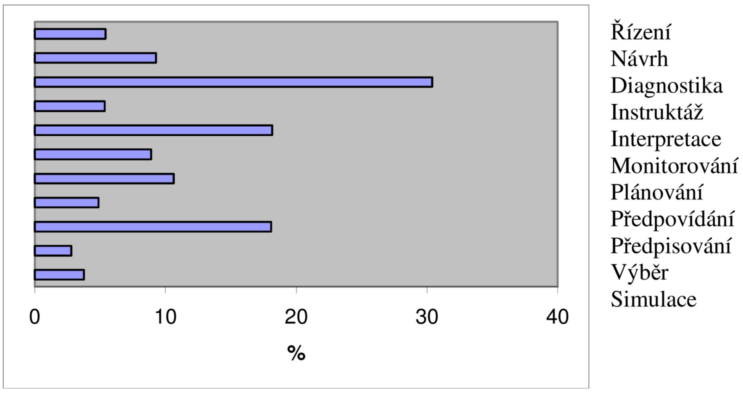


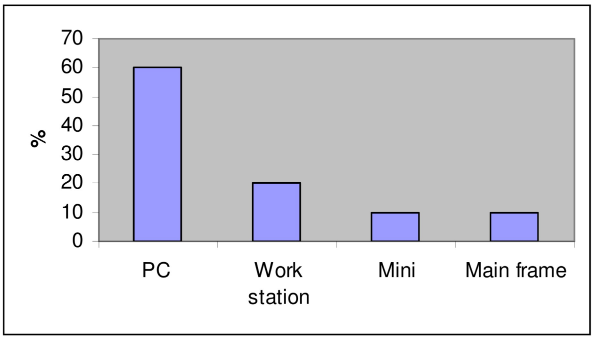
BD – báze dat
BZ – báze znalostí
IM – ~~informační~~ mechanismus (usuzovací)
Inferenční

Srovnání ES a člověka experta

	ES	Člověk
Dostupnost v čase	okamžitá	-
Dostupnost v místě	všude	-
Bezpečnost	↑ (kopie,…)	-
Smrtelnost	-	+
Výkonnost	konstantní	proměnlivá
Rychlost	konstantní ↑	proměnlivá
Cena	↑	↑↑







1.1 K ČEMU ES JSOU?

- náhrada experta – člověka
- pomoc expertovi – člověku

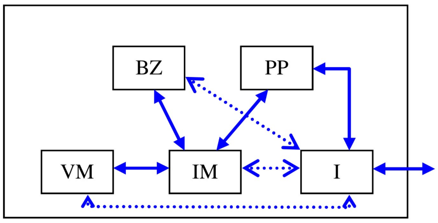
1.2 KDE JSOU ES NASAZOVÁNY?

1. obrázky nasazení v oborech
2. příklady:
   Drilling Advisor – Rádce ropných vrtů – Elf Aquitaine of France
   Cooker Advisor – Rádce ve sterilizačním procesu potravin
   R1/XCON – Konfigurace počítačů DEC
   Lending Advisor – finančnictví - úvěry

2 CHARAKTERISTIKY ES

2.1 STRUKTURA ES

IM = část ES, která srovnává fakta z PP (pracovní paměť) se znalostmi v BZ za účelem nalezení závěru řešeného problému



VM – vysvětlovací mechanismus
I - interface

BZ = část ES, která obsahuje oborové znalosti
PP = BD = část ES, která obsahuje konkrétní fakta (data) o dané řešené úloze a to jak vložená uživatelem (databáze), tak odvozená IM během práce ES

Znalost, informace, data, šum

metaznalosti
znalosti
informace
data
šum

Metaznalost = znalost o znalostech

Příklad: Mějme posloupnost 26 číslic
- 13907665847932520051564315 … <u>šum</u>
- když budeme vědět, že posloupnost obsahuje údaje, pak se jedná o data … <u>data</u>
- předpokládejme, že existuje předpis, jak tato data převést na informaci … <u>informace</u>
  Např.:
  1. rozdělíme posloupnost na: 13 90 76 … 15
  2. vynecháme každé číslo menší než 32: 90 76 65 … 43
  3. nahraď zbývající čísla ASCII znaky:
     ZLATO 438 + → stoupá
     cena
- Máme-li znalost
  IF ZLATO levnější než 500 a cena stoupá THEN nakupuj ZLATO
  pak informaci můžeme využít.

znalost = předpis, jak nakládat s informacemi

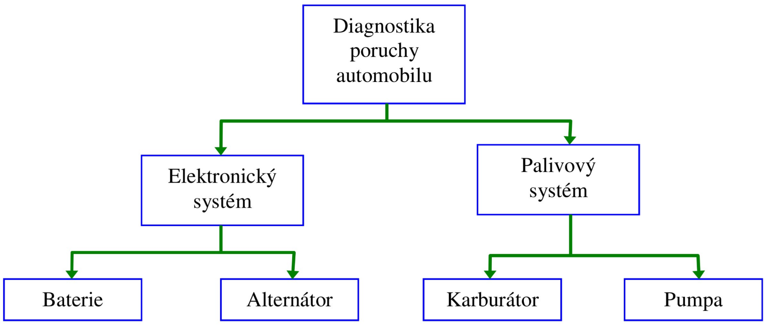
2.2 CHARAKTERISTIKY ES

1. separace znalostí od řízení (BZ x IM)
2. obsahuje expertní znalosti (BZ)
3. úzká oblast expertýzy
   ne příliš hluboká oblast expertýzy
   musí existovat člověk, který danou úlohu umí řešit
4. usuzování se symboly
   Příklad: IF osoba má horečku THEN osoba dostane aspirin
   Data: Jan má horečku
   Horečka (Jan) Horečka(x) → Dát(x, aspirin)
   x = Jan → Dát (Jan, aspirin)
5. heuristické usuzování
   Příklad: Lidé jsou v létě <u>málokdy</u> nachlazeni.
   Je-li podezření, že pacient má rakovinu, <u>nejdříve</u> se podívám na rodinnou anamnézu.
6. nepřesné usuzování
   zdroje nepřesnosti: - neurčitost dat BD (PP)
   - neurčitost znalostí BZ
   - neurčitost při práci IM IM
7. ES dělá chyby

Ad 6. Příklad:
Pacient je na ARO (JIP). Chceme diagnostikovat jeho stav.
- Nepřesná data: Chybí výsledky laboratorního testu nebo EKG, nebo jsou zatíženy chybou.
- Nepřesné znalosti: Chybí-li výsledky z laboratorního testu nebo EKG a pacient má bolest na hrudi, pak se <u>může</u> jednat o srdeční problém.

Dovede-li lidský expert vyřešit úlohu přibližně za 15 minut ⇔ složitost problematiky je vhodná pro nasazení ES.

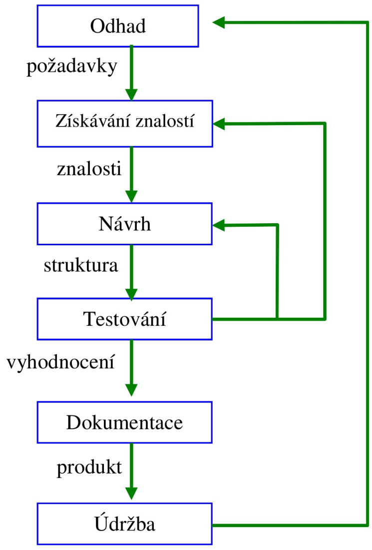
Složitost ⇒ rozklad



2.3 VÝVOJOVÉ FÁZE ES

(x vývoj konvenčního programu)

KONVENČNÍ PROGRAMY	EXPERTNÍ SYSTÉMY
Numerické hodnoty	Symboly
Algoritmus	Heuristika
Informace a řízení pohromadě	Znalosti oddělené od řízení
Těžké modifikovat	Lehké modifikovat
Přesná informace	Neurčitá informace
Povelový Interface	Přirozený dialog s vysvětlením
Dává konečný výsledek	Dává doporučení s vysvětlením
Optimální řešení	Přijatelné řešení



Iterativní proces:
- stavba prototypu
- testování
- modifikace

ES vzniká jako vedlejší produkt porozumění dané problematice.

Fáze vývoje

1. Odhad
   Zkoumá se proveditelnost, oprávněnost nasazení
   Definuje se okruh problému
   Definují se zdroje znalostí (člověk – expert, příručky, zprávy, …)
2. Získávání znalostí (Knowledge Acquisition)
   Jeden z hlavních problémů při tvorbě ES
   - pro porozumění problému
   - pro návrh ES

   Získávání znalostí = proces získávání, organizování a studování znalostí
   - schůzka s expertem
   - získání základní představy o metodě, kterou expert uplatňuje
3. Návrh
   a) zvolit nejlepší způsob reprezentace znalostí
   b) je definována základní struktura báze znalostí (BZ) a definována metoda práce se znalostmi (IM) [BZ + IM $\Rightarrow$ výběr SW]
   c) prototyp ES sloužící jako vstup do iteračního vývojového cyklu
4. Testování
   - spolupráce s expertem
   - spolupráce s uživatelem
5. Dokumentace
   - pro uživatele
   - pro vývojáře (tzv. znalostního inženýra):
     - slovník znalostí
     - procedury řešení
     - definice struktury ES a strategie IM
       (dekompozice na dílčí BZ, každá s nějakým mechanismem inference, jak jsou dílčí BZ řízeny, jak je řízen celek)

6. Údržba
    - ES se neustále učí

Všechny kroky 1 – 6 jsou provázány, nemusí být prováděny přesně podle uvedeného pořadí

2.4 LIDÉ ZAPOJENI V PROJEKTU ES

- oborový (domain) expert
- znalostní inženýr
- uživatel

2.4.1 Oborový expert

= osoba mající znalosti a dovednosti, která řeší problém mnohem lépe než ostatní lidé

- má expertní znalosti … hodnota experta = expert – neexpert
- má dovednosti jak vyřešit problém
- komunikační schopnost, např.: dekompilace znalostí

2.4.2 Znalostní inženýr

= člověk provádějící znalostní inženýrství a je za ně zodpovědný
(psycholog, diplomat, výzkumník, programátor)

- dovednosti
  umění, neboť nezná přesný postup jak vyvinout ES
- komunikační schopnosti
- schopnost naprogramovat ES pomocí vývojového SW
  schopnost vybrat vhodnou metodu reprezentace znalostí
  → vhodný SW:
  (systémy založené na pravidlech, systémy založené na rámcích)

2.4.3 Uživatel

Uživatel by měl:

- pomoci definovat specifikaci interface (interface znamená ≈ 1/2 rozpočtu projektu)
- pomoci při získávání znalostí (ze začátku více než expert)
- pomoci při vývoji software
  - definovat přístup k systému
  - vstup informace (menu, databáze, data informačního systému)
  - specifikovat vysvětlovací mechanismus
  - definovat formu výstupu (databáze,…)
  - specifikovat utility

2.5 VÝVOJ

Konvenční programy	Expertní systémy
- důraz na řešení	- důraz na problém
- programátor pracuje sám	- týmová práce
- sekvenční vývoj	- iterativní vývoj

Požadovaná kvalifikace:
Domain Expert (oborový expert):
- má expertní znalosti
- má dovednosti jak řešit problém
- je schopen předat znalosti
- je schopen věnovat čas
- není nepřátelský

Knowledge Engineer (znalostní inženýr):
- má dovednosti znalostního inženýrství
- dobré dorozumívací schopnosti
- umí přiřadit problému vhodný software
- umí programovat expertní systémy

End - User (uživatel):
- umí pomoci definovat specifikaci interface
- umí pomoci při získávání znalostí (knowledge acquisition)
- umí pomoci při vývoji software

Hlavní odpovědnosti znalostního inženýra:
- ohodnocení problému
- rozhovor s expertem
- identifikace koncepce expertem užívané
- organizace znalostí
- identifikace metody řešení problému expertem užívané
- výběr software
- „kódování" znalostí do systému
- testování systému
- revize systému
- integrace systému do pracovního prostředí
- údržba systému

3 REPREZENTACE ZNALOSTÍ

Znalost = abstraktní pojem představující individuální porozumění danému problému
≡ porozumění oblasti předmětu
(Pro nás znalost ≡ oborová znalost - např. část oblasti medicíny - např. infekční nemoce)
Obor = dobře vymezená oblast daného předmětu

Jak zachytit znalosti?
Reprezentace znalostí = metoda užívaná k zakódování znalostí do báze znalostí ES
Neexistuje jedna metoda, jak znalosti zakódovat.

3.1 TYPY ZNALOSTÍ

Psychologové vymysleli několik teorií, jak člověk řeší problémy a jak uchovává znalosti. Výzkumníci v UI přebrali tyto výsledky a vyvinuli metody, jak je nejlépe reprezentovat v počítači.
Neexistuje jedna nejlepší teorie, jak znalosti uchovávat, ale je jich několik. Protože znalostní inženýr musí umět vybrat nejlepší z nich „šitou" na daný problém, měl by znát všechny.

Typy znalostí
- procedurální – pravidla, strategie, agendy, procedury
- deklarativní – pojmy, objekty, fakty
- metaznalosti – znalosti o jiných typech znalostí a jak je užít
- heuristické znalosti (mělké) – znalosti „vycucané z palce“
- strukturální znalosti (celkový model ES) – soubory pravidel, vztahy mezi pojmy, vztahy mezi objekty

3.2 TECHNIKY REPREZENTACE ZNALOSTÍ

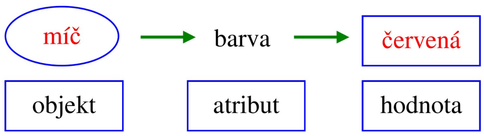
- O - A - V triplet (O - V - H triplet)
- pravidla
- sémantické sítě
- rámce
- logika

Každá má své výhody i nevýhody.

3.2.1 O - A - V

(Object - Attribute - Value = Objekt - Vlastnost - Hodnota)
Základním prvkem je <u>fakt</u> = forma deklarativní znalosti. V UI a ES se často hovoří o <u>výroku</u>.
Fakt = výrok = tvrzení, které může být pravdivé nebo nepravdivé
V pracovní paměti je každému faktu přiřazena hodnota TRUE nebo FALSE.
O-A-V je komplexnější pojem výroku, faktu.
Fakt užijeme při přiřazení hodnoty vlastnosti objektu.

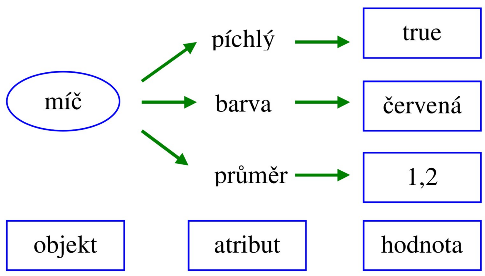
Příklad: Fakt „míč má červenou barvu“ vyjádříme pomocí O - A – V:
O - míč
A - barva
V - červená



Hodnota může být
- Boolean … true/false
- numerická … číslo
- textová … řetězec

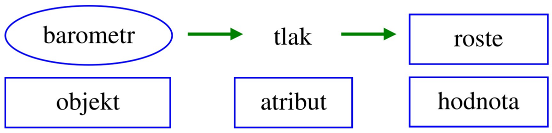
Objekt ale může mít několik vlastností:

Příklad: Objekt s několika vlastnostmi:



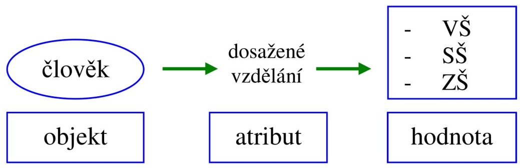
3.2.1.1 jednohodnotový a vícehodnotový fakt

Příklad: jednohodnotový fakt vyjádřený pomocí O-A-V:



Tlak může nabývat právě jedné hodnoty z množiny tlaků {roste, klesá, stojí}.

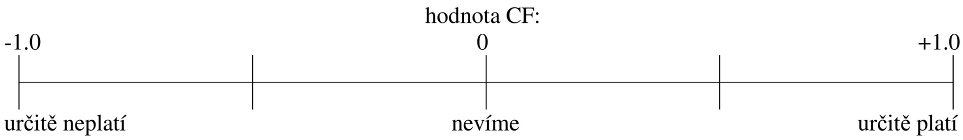
Příklad: vícehodnotový fakt vyjádřený pomocí O-A-V:



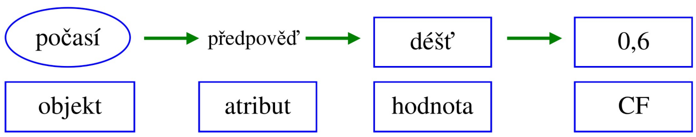
Dosažené vzdělání může nabývat několika hodnot z množiny vzdělání (základní, středoškolské, vysokoškolské, postgraduál)
Oba - jednohodnotový i vícehodnotový fakt vyjádřený pomocí O-A-V mají tuto podstatnou vlastnost: pokud přiřadíme hodnotu/hodnoty atributu, systém (program ES) automaticky přiřadí této hodnotě/těmto hodnotám hodnotu TRUE a všem ostatním hodnotám atributu hodnotu FALSE.

3.2.1.2 nepřesné fakty

Jistotu pravdivosti faktu vyjadřujeme pomocí činitele jistoty (CF = Certainty Factor)



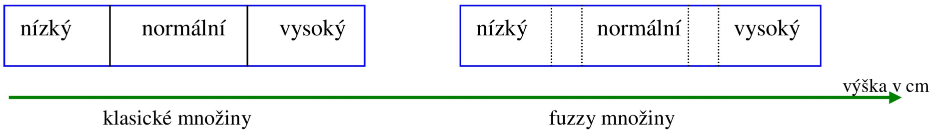
Příklad:



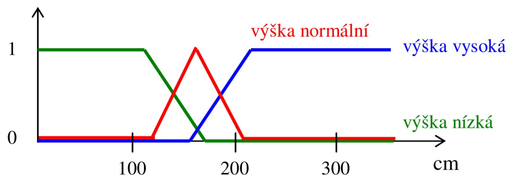
3.2.1.3 Fuzzy fakty

Člověk může říci: „Osoba X je vysoká." Kolik to ale je cm?
<u>Fuzzy logika:</u>
Umožňuje pracovat s metodami reprezentace nejednoznačného tvrzení i s metodami uvažování, které tato nejednoznačná tvrzení využívají. Používá pojmy fuzzy množiny:
Klasická teorie množin každému prvku univerza přiřadí buď prvek dané množiny (s absolutní určitostí) anebo prvek jejího doplňku (opět s absolutní určitostí).
Fuzzy (nezřetelná) množina však nemá přesné hranice a proto nelze prvek univerzální množiny jednoznačně přiřadit buď k množině anebo k jejímu doplňku.
Pro každou fuzzy množinu je proto zavedena tzv. <u>funkce příslušnosti</u>, která každému prvku univerza přiřadí váhu naší důvěry (tzv. <u>stupeň příslušnosti</u>) v to, že tento prvek je prvkem dané fuzzy množiny.

Příklad:



funkce příslušnosti

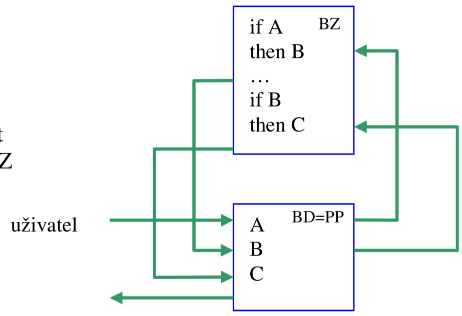


3.2.2 Pravidla

pravidlo = reprezentace znalostí udávající vztah mezi vstupní informací a výstupní informací, která se v případě, že vstupní informace je známá, stane také známou

IF
AND
AND
OR
THEN
AND
(ELSE) … nepoužívat

- pravidla jsou uložena v BZ



Co všechno může pravidlo dělat

- volání procedury
- volání programu

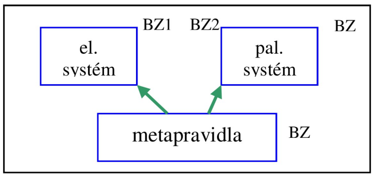
IF telefonovat THEN CALL program (jméno, telefonní číslo)

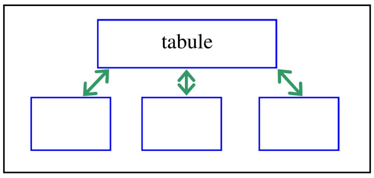
Typy pravidel

- vzájemný vztah … IF baterie je špatná THEN auto nestartuje
- doporučené … IF auto nestartuje THEN použij taxi
- příkaz … IF auto nestartuje THEN zavolej taxi službu
- strategie … IF máš použít taxi THEN zkontroluj, jestli máš peníze, cenu/km, taxametr
- heuristika

- pravidla mohou pracovat s proměnnými … IF X je starší jak 65 let THEN důchod
- neurčitá pravidla … IF inflace ↑ THEN téměř jistě úroková míra roste 0,8

- metapravidla
  = pravidlo o pravidlech
  IF auto nestartuje
  AND závada není v palivovém systému
  THEN použij pravidla o elektrickém systému

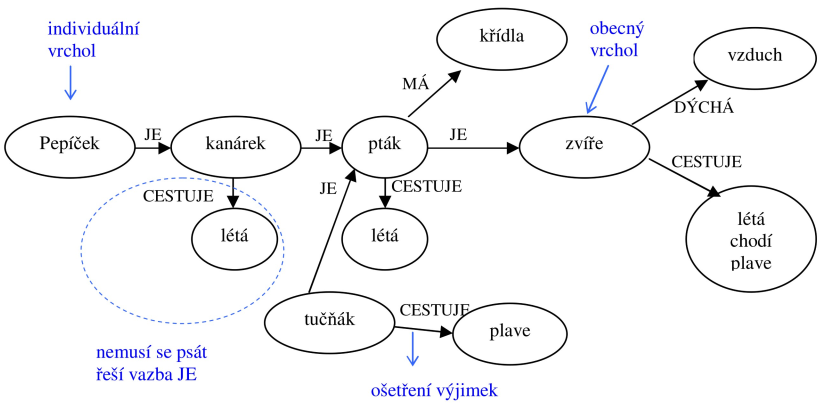




tabule = místo v PP (=BD), ve kterém si jednotlivé BZ vyměňují informace

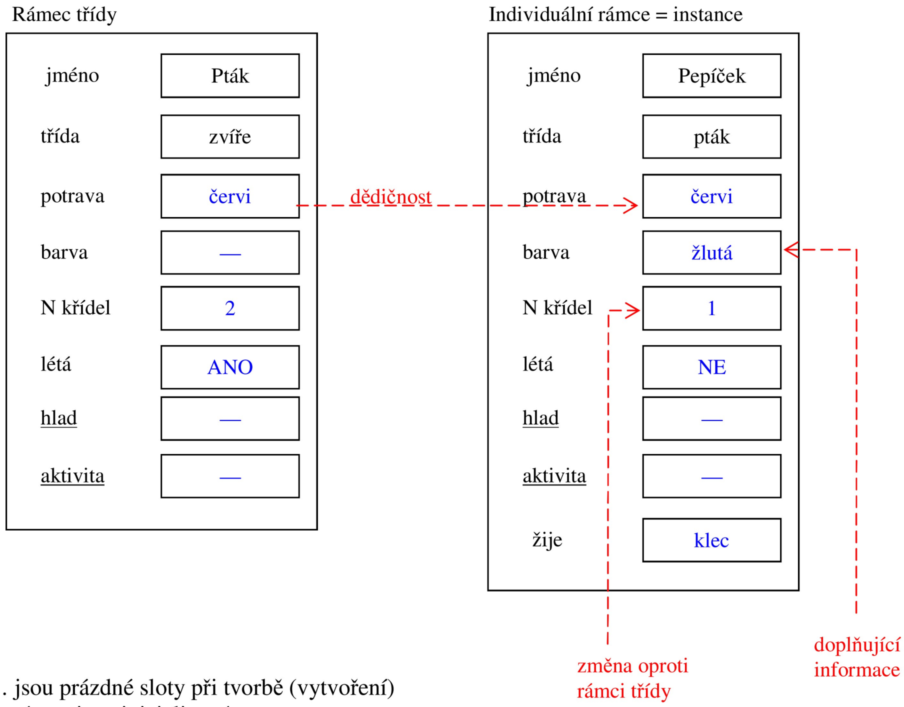
3.2.3 Sémantické sítě

sémantická síť = reprezentace znalostí vyjádřená formou grafu, kde vrcholy grafu určují objekty, vlastnosti, hodnoty a hrany určují vztahy mezi nimi



3.2.4 Rámce

Rámec = datová struktura, která zahrnuje stereotypní znalosti o daném objektu nebo pojmu



- … jsou prázdné sloty při tvorbě (vytvoření) rámce jsou inicializovány

- … hlad metody
… aktivita (procedury)

  Metoda_Hlad( ) … říká, co má pták dělat, když má hlad
  Metoda_Aktivita( ) … říká, co pták dělá

Faseta = vlastnost rámce, která:

1. může definovat obor hodnot pro každý datový typ (řetězec, číslo, boolean,...), např.: N křídel číslo $0 \leq N \leq 2$
2. pomůže zjistit hodnotu slotu, je-li neznámá - faseta IF_NEEDED
pomůže říci, co dělat, když se hodnota slotu změní - faseta IF_CHANGED

Příklad:

Faseta IF_NEEDED

V rámci Pepík (tj. individuální rámec) např. nemáme k dispozici hodnotu slotu létá (neznáme ji):
létá: $\boxed{—}$ (prázdný slot)
Chceme tuto hodnotu určit. Voláme fasetu IF_NEEDED.
Faseta volá metodu:

```
IF N křídel = 2
THEN létá := ANO
ELSE létá := NE
```

V Shellu se zapisuje:
IF Pepík : N_křídel = 2
THEN Pepík : létá := ANO
ELSE Pepík : létá := NE

lépe:
IF Self : N_křídel = 2
THEN Self : létá := ANO
ELSE Self : létá := NE

Faseta IF_CHANGED
V rámci Pepík se změní hodnota slotu hlad z NE na ANO.
Co se stane dál?
Vyvolá se faseta IF_CHANGED.
Faseta volá metodu:
IF Self : Hlad = TRUE
THEN Self : Aktivita = „jí“ + Self : Potrava
Pak se objeví ve slotu Aktivita :
aktivita | jí červy |

3.2.5 Logika predikátový počet

Příklad: Každý, kdo má rád pivo a pije pivo, má rád hospodu.
rád (X, pivo)
pije (X, pivo)
$\forall$X: rád (X, pivo) $\wedge$ pije (X, pivo) $\rightarrow$ rád (X, hospoda)

4 INFERENČNÍ MECHANISMUS

4.1 UVAŽOVÁNÍ ČLOVĚKA

Uvažování = proces práce se znalostmi, příchozími či vyvozenými fakty a strategiemi řešení, který se snaží dospět k závěru
(fakt = tvrzení
výrok = tvrzení TRUE, FALSE
axiom = TRUE výrok)

4.1.1 Dedukce

axiom (fakt): stojím v dešti (A)
implikace (pravidlo): každý, kdo stojí v dešti, zmokne
závěr (nový fakt): zmoknu (B)
Pravidlo modus ponens: Když platí A a platí A$\rightarrow$ B, pak platí B.

4.1.2 Indukce

Platí něco pro x_1, x_2, …, x_n zobecníme, že platí pro $\forall x_i$, pak platí i pro x_{n+1}, …

4.1.3 Abdukce

Když platí B
a platí A $\rightarrow$ B,
pak platí A.
Vím, že zmoknu.
Abdukcí usoudím, že: stojím v dešti.

4.1.4 Analogie

Člověk vytváří model na základě zkušenosti. Tento model pak používá k tomu, aby porozuměl situaci nebo objektu. Na základě analogie pak mezi dvěma situacemi či objekty člověk hledá stejné rysy nebo naopak odlišnosti.

TYGR		LEV
barva:	červená s pruhy	
žije kde:	Indie, Asie,…	
jí co:	maso	
počet nohou:	4	
patří do:	šelma	

4.1.5 Heuristika

Selský rozum opírající se spíše o dobrý úsudek (mínění) než o exaktní logiku může velmi efektivně řešit problém.

Např.: „Volný řemen větráku většinou způsobí podivný hluk.“

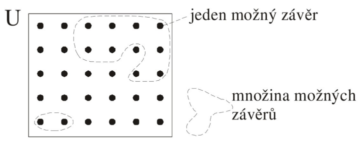
Tuto znalost si může vytvořit mechanik v autoservisu.

Je to znalost heuristická, vycucaná z prstu.

Užijeme-li heuristické znalosti k řízení hledání v BZ, nazýváme toto hledání heuristickým hledáním nebo nejlepší – první hledání (best-first search). Heuristické hledání nezaručuje řešení, jen garantuje, že hledání se bude ubírat „rozumným“ směrem. Používá se tam, kde chceme rychlé řešení.

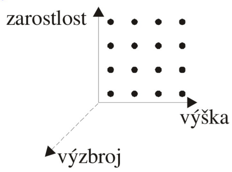
4.1.6 Nemonotónní usuzování

Jsou-li sbírané fakty (tvrzení) statické (neměnné), jedná se o monotónní usuzování.

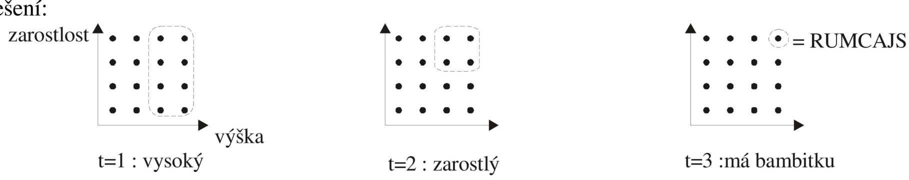


Stupeň vysvětlení: je tím větší, čím menší je množina možných závěrů.

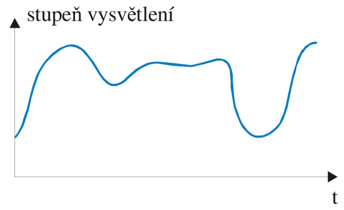
Příklad:



Řešení:



Není-li fakt statický v průběhu usuzování (může docházet k jeho změnám) ⇒ stupeň vysvětlení se může snížit ⇒ má nemonotónní průběh ⇒ nemonotónní usuzování



Příklad:
IF vítr fouká
THEN kolébka se houpá
Fakta: t = 1: vítr fouká ⇒ kolébka se houpá ⇒ …
t = 2: je vichřice ⇒ kolébka se houpá ⇒ …
t = 3: vichřice skončí ⇒ kolébka se nehoupá ⇒ …

Člověk uvažuje nemonotónně. Systém, který je schopen nemonotónního usuzování, se nazývá nemonotónní systém – příkladem je tzv. truth-maintenance system - viz. skripta [Kepka]

4.1.7 Generování a testování

4.1.8 Default usuzování

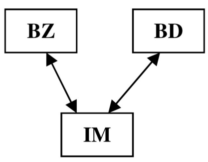
- nezná-li se předpoklad, bere se default hodnota (dědičnost)

4.1.9 Intuice

4.2 INFERENCE

= proces práce ES se znalostmi, vstupními a vyvozenými fakty používající definované strategie řešení, které vede k vyvození závěru / závěrů
= proces užívaný ES k vyvození nové informace z informace známé

báze znalostí (BZ) – dlouhodobá paměť báze dat (BD) – krátkodobá paměť



Otázky:
1. Jak prohledávat BZ?
2. Jaké otázky klást uživateli?
3. Jaké pravidlo vybrat?
4. Jak vyvozená (nová) informace ovlivní prohledávání BZ?

V ES založených na pravidlech se řeší DŘ (dopředné řetězení) a ZŘ (zpětné řetězení)
Původ postupu v logice – logickém uvažování

4.2.1 Pravidlo modus ponens

$\boxed{[A \wedge (A \Rightarrow B)] \;\Rightarrow\; B}$

Příklad:

R1:	IF (pacient~~o~~va t > 38°C)	E1
	THEN (pacient má horečku)	E2
R2:	IF (pacient má horečku)	E2
	THEN (vezme si 2 prášky léku)	E3

Platí ny~~n~~ní: (t > 38°C). Pak máme:

E1; E1 → E2; E2 → E3

t = 1: E1; E1 → E2; E2 → E3

t = 2: E1; E2; E1 → E2; E2 → E3

t = 3: E1; E2; E3; E1 → E2; E2 → E3

4.2.2 Rezoluce

Chceme-li odvodit (dokázat) hypotézu, je lépe použít rezoluce (nežli modus ponens)

$B \vee A \quad C \vee \bar{B} \quad \not\Leftarrow$

(A ∨ B) ∧ (¬B ∨ C) ⇒ A ∨ C

Příklad: $C \vee A$

Cílem je dokázat, že E3 je pravda, mám E1; E1 → E2; E2 → E3

Řešení:

1. připíši ¬E3
   E1; ¬E3; E1 → E2; E2 → E3
2. • ¬E1 ∨ E2
   • ¬E2 ∨ E3
   • E1
   • ¬E3

t = 1: • ¬E1 ∨ E2
• ¬E2 ∨ E3
• E1
• ¬E3 ∨ ∅
(¬E2 ∨ E3, ¬E3 ∨ ∅ } ¬E2 ∨ ∅)

t = 2: • ¬E1 ∨ E2
• ¬E2 ∨ ∅
• E1
• ¬E3
(¬E1 ∨ E2, ¬E2 ∨ ∅ } ¬E1 ∨ ∅)

t = 3: • ¬E1 ∨ ∅
• ¬E2
• E1 ∨ ∅
• ¬E3
(} ∅ ∨ ∅ = ∅ ⇒ platí E3 (důkaz sporem))
↑ rezolventa

$\bar{A} \Rightarrow B$

A	B	∨	?	$\bar{A}$	B	⇒
1	1	1	1	0	1	1
1	0	1	1	0	0	1
0	1	1	1	1	1	1
0	0	0	0	1	0	0

$\underline{(\bar{A} \Rightarrow B) \wedge (B \Rightarrow C)} \Rightarrow$

$A \vee C$

$\Rightarrow \underline{\bar{A} \Rightarrow C}$

$(\bar{B} \Rightarrow A) \wedge (\bar{C} \Rightarrow \bar{B}) \Rightarrow$

$\bar{C} \Rightarrow A$

$\bar{A} = D$

$(D \Rightarrow B) \wedge (B \Rightarrow C)$

$\Rightarrow$

$D \Rightarrow C$

$(\bar{C} \Rightarrow \bar{B}) \wedge (\bar{B} \Rightarrow A)$

$\Rightarrow$

$\bar{C} \Rightarrow A$

4.2.3 Nerezoluce

Na rozdíl od rezoluce si pamatuje, co je předpokladem a co závěrem.

4.3 DOPŘEDNÉ ŘETĚZENÍ (DŘ)

Nejdříve se seberou všechna data od uživatele (všechny informace), pak se usuzuje na závěr. Např. lékař sbírá informace o symptomech nemoci od pacienta. Dopřednému řetězení se proto říká hledání řízené daty. DŘ vychází z pravidla modus ponens.

DŘ = dopředné řetězení = inferenční strategie, která začíná množinou známých faktů, odvozuje nová fakta použitím pravidel, jejichž předpoklady odpovídají známým faktům, a pokračuje v tomto procesu, dokud není odvozený cílový stav nebo již neexistují pravidla, která mají předpoklad vyhovující faktům.

4.3.1 Proces DŘ probíhá takto:

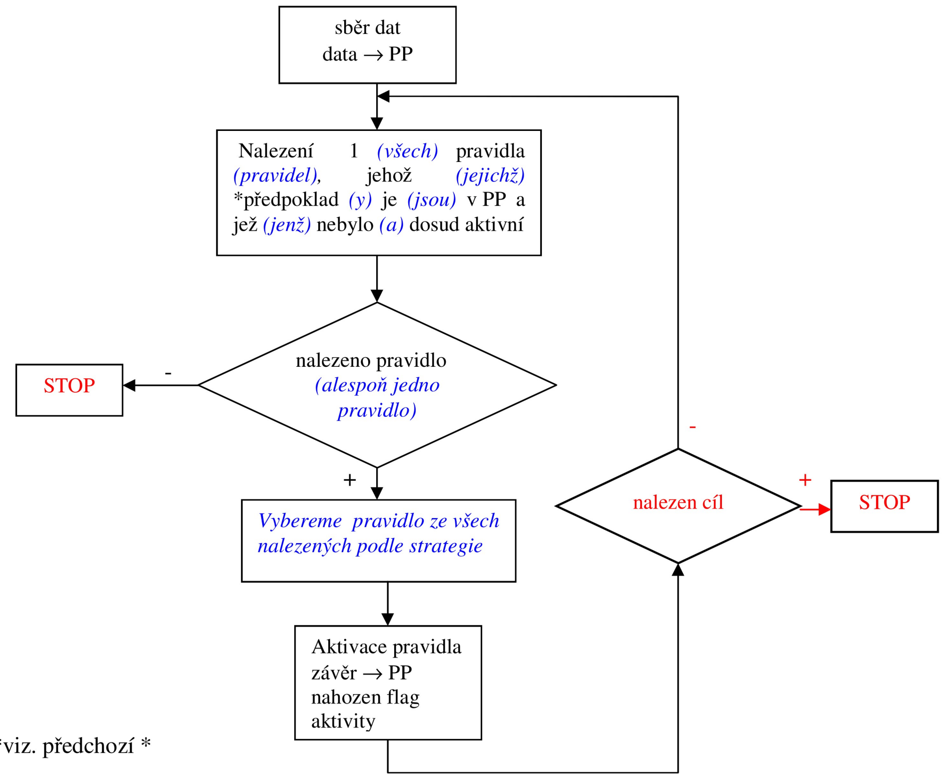
Inicializace: žádné pravidlo není aktivní ani nebylo aktivní (tj. flag aktivity je shozen u každého pravidla)

1. Informace od uživatele se zapíše do PP (pracovní paměti).
2. Prohledá se BZ, a to od začátku, a najde *(najdou)* se první *(všechna)* pravidlo *(a)*, jehož *(jejichž)* *předpoklad *(y)* vyhovuje *(í)* faktům v PP a jež *(jenž)* nebylo *(a)* dosud aktivní (tj. flag aktivity je shozen). Pokud se pravidlo nenajde, jde se na krok 4.
   Vybere se jedno pravidlo podle zvolení rozhodovací strategie.
3. Pravidlo se aktivuje, tj. jeho závěr se přidá do PP a označí se, že bylo aktivováno (tj. flag aktivity nahozen)
4. Pravidlo se při prohledávání BZ nenajde ⇒ proces končí

(IF E1
AND E2)
OR E3
*celkový předpoklad

THEN C

Dopředné řetězení:



*viz. předchozí *

4.3.2 Příklad na ZŘ a DŘ

Diagnóza streptokokové nákazy v krku

Pravidla v BZ:

Pr1 IF bolení v krku
AND očekáváme bakteriální infekci
THEN streptokoková nákaza …… je (goal) cíl 1

Pr2 IF t > 38°C
THEN pacient má horečku

Pr3 IF pacient nemocen ≥ 2 měsíce
AND pacient má horečku
THEN očekáváme bakteriální infekci

Pacient má tyto údaje:

t = 38,5°C ⇒ t > 38°C
nemocen 2 měsíce ⇒ pacientova nemoc ≥ 2 měsíce
bolení v krku

\

Legenda: P1 … dílčí předpoklad Pr1 … dílčí pravidlo

CYKLUS 1 PP

Pr1, P1 (pravidlo, předpoklad)
(bolení v krku) +

t > 38°C
nemoc ≥ 2 měsíce
bolení v krku

Pr1, P2
(očekáváme bakteriální infekci) –

—’’ —

•Pr2, P1 +
(t > 38 °C)
aktivace Pr2
(pacient má horečku) závěr do PP

—’’ —
pacient má horečku

Konec cyklu

CYKLUS 2

Pr1, P1 +

—’’ —
—’’ —

Pr1, P2
P1 nemohu aktivovat –

—’’ —
—’’ —

Pr3, P1
(nemoc ≥ 2 měsíce) +

—’’ —
—’’ —

•Pr3, P2
(pacient má horečku)
aktivace Pr3: závěr do PP
(očekáváme bakteriální infekci)

—’’ —
—’’ —
oček.bakter. infekci

Konec cyklu

CYKLUS 3 +

Pr1, P1

—’’ —
—’’ —
—’’ —

•Pr1, P2 +
aktivace Pr1:

(streptokoková nákaza) závěr do PP

—’’ —
—’’ —
—’’ —
streptokok. nákaza

Konec cyklu

CYKLUS 4

neexistuje žádné vhodné pravidlo ⇒ STOP

Dopředné řetězení vyvozuje ze vstupních dat cokoliv (hledání řízené daty) – i když je to zbytečné.
Vylepšení DŘ spočívá v testování PP po každé její změně, zda neobsahuje žádaný cíl. Obsahuje-li jej, proces končí.

Příklad:
V BZ budeme mít navíc tato pravidla:
1. IF pacient má horečku
   THEN nepůjde do kina
2. IF nepůjde do kina
   THEN půjde na seminář   cíl 4

Pořadí pravidel v BZ: 1, 2, 3, 4, 5 ⇒ dopadne to stejně v případě, že použijeme „vylepšení“
Pořadí pravidel v BZ: 1, 5, 4, 3, 2 ⇒ nedopadne to stejně, přibudou 2 cykly (i když aktivace 2, 4, 5, 3, 1, Ø použijeme „vylepšení“)

Z příkladu plyne, že záleží na pořadí pravidel v BZ.
Při špatném uspořádání pravidel:
1. v každém cyklu může být prohledáno velké množství pravidel
2. mohu odvozovat zbytečné cíle (závěry) (i při „vylepšení“)

⇓

Pořadí pravidel udává, jaká je efektivita procesu DŘ.

4.3.3 Řešení konfliktů

Příklad:
V BZ budou navíc tato pravidla:
3. IF pacient má horečku
   THEN zajde do lékárny   cíl 2
4. IF pacient má horečku
   THEN zajde   cíl 3

Je-li pořadí pravidel 1, 2, 3, 4, 5, 6, 7 ⇒ ? cíl : cíl 1
Je-li pořadí pravidel 1, 5, 4, 3, 2, 6, 7 ⇒ cíl 4
Je-li pořadí pravidel … jiné … ⇒ cíl 2, nebo 3

Z uvedeného příkladu vyplývá, že v určité fázi mohu aktivovat více pravidel, neboť mají splněné předpoklady, a závěry těchto pravidel mohou být v konfliktu.
Standardní postup je vybrat 1. pravidlo, ale pak je výsledek závislý na pořadí pravidel

řešení konfliktů = strategie určující N-best pravidel v případě, že více jak jedno pravidlo může být aktivní

Typy strategií:
1. vybrat 1. pravidlo podle pořadí
2. podle priority (každé pravidlo má přiřazenou prioritu)
3. podle specifičnosti pravidel

IF A	IF A	IF C	PP:
AND B	THEN Z2	THEN Z3	A
THEN Z1		↑	B
nejvíce specifické			C ↓

4. podle čerstvosti předpokladů v PP   nejčerstvější
5. neuvažovat již aktivovaná pravidla
6. výběr více pravidel, závěry do oddělených PP a úlohu řeším paralelně

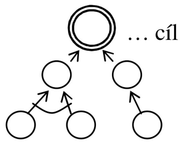
Pomocí strategií realizujeme řešení konfliktů, realizujeme heuristické hledání v inferenční síti (AND/OR stromech)

4.3.4 DŘ jako tříkrokový proces

DŘ je tříkrokový proces:
1. rozpoznej (∀ pravidla, jejichž předpoklady vyhovují údajům v PP)
2. rozhodni (výběr jednoho pravidla – podle strategie)
3. proveď (aktivace pravidla, závěr do PP)

4.4 ZPĚTNÉ ŘETĚZENÍ (ZŘ)

Použiji v případě, že se snažím ověřit hypotézu (hypotézy)

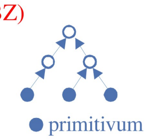

… cíl = hypotéza (platí či neplatí ?)

= hledání řízené cílem, používá princip rezoluce

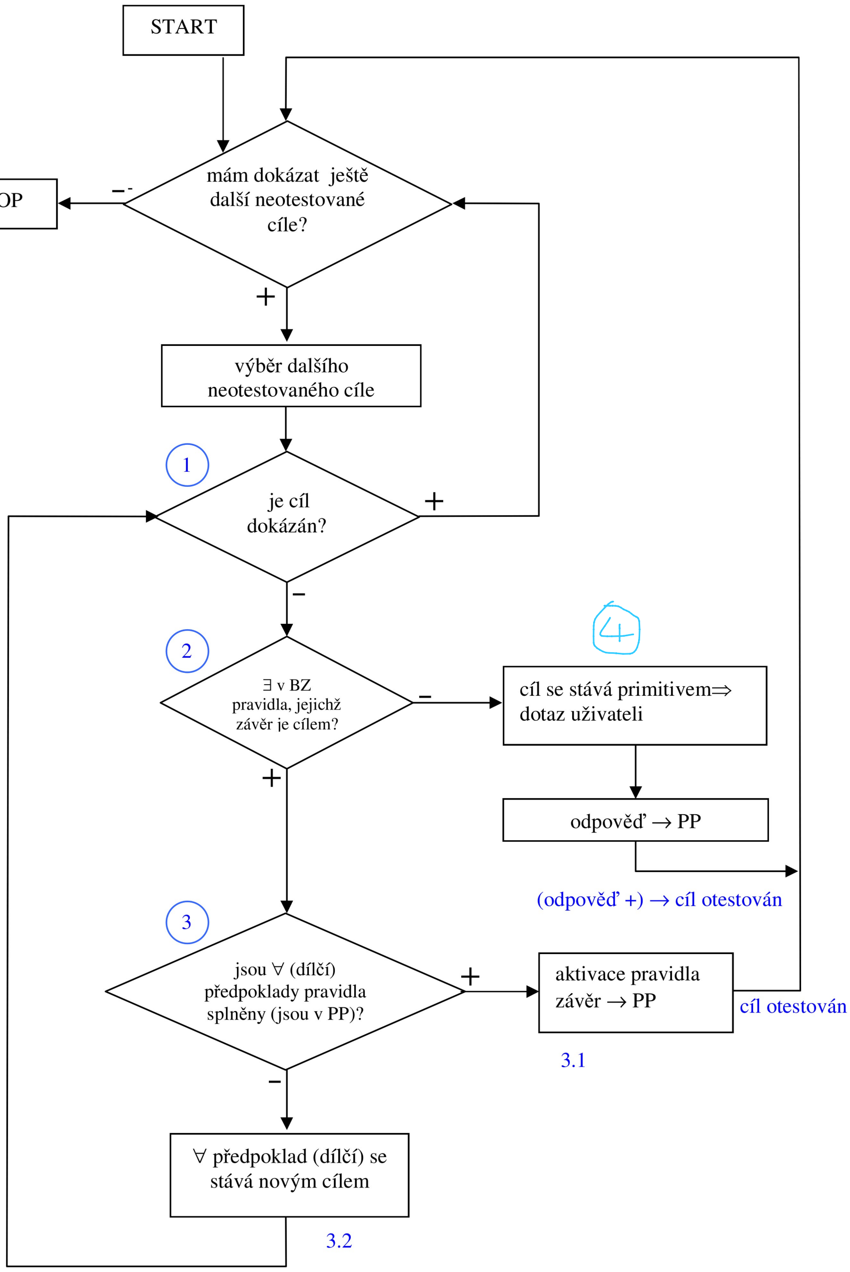
ZŘ = inferenční strategie, která se snaží dokázat hypotézu (cíl) sbíráním faktů (informací), které podporují dokazovanou hypotézu

4.4.1 Postup při ZŘ

1. Začneme důkazem cíle:
   → podíváme se, zda cíl již není dokázán v PP (může ho dokázat i jiná BZ)
   - když je dokázán, jdeme na dokazování dalšího cíle, neexistuje-li další cíl, pak končíme
2. hledáme pravidla (v BZ) taková, že jejich závěr je totožný s dokazovaným cílem (tzv. cílová pravidla)
3. najdou-li se taková pravidla, podíváme se na jejich předpoklad, zdali je v PP
   3.1 předpoklad je v PP
      - pravidla aktivujeme, jejich závěry jdou do PP a jdeme na krok 1
   3.2 předpoklad není v PP
      - předpoklad prohlásíme za nový cíl (podcíl) a jdeme na krok 1
4. nenajdou-li se pravidla (v BZ) – z kroku 2 taková, že jejich závěr je totožný s dokazovaným cílem (podcílem):
   cíl = závěr = primitivum = tvrzení, které není podporované žádným pravidlem (v BZ)
   - systém se zeptá na „závěr“ (= primitivum) uživatele
   Uživatel odpoví:
   „virtuální“
   - primitivum platí : aktivuji pravidlo (tj. zapíši primitivum do PP)
     dáme primitivum do PP: jdeme na krok 1 (budeme dokazovat další cíl)
   - primitivum neplatí:
     cíl nedokázán („negace primitiva“ → PP), jdeme na krok 1 (budeme dokazovat další cíl)



ZŘ


START
mám dokázat ještě další neotestované cíle?
STOP
–
+
výběr dalšího neotestovaného cíle
1
je cíl dokázán?
+
–
2
∃ v BZ pravidla, jejichž závěr je cílem?
–
+
4
cíl se stává primitivem⇒ dotaz uživateli
odpověď → PP
(odpověď +) → cíl otestován
3
jsou ∀ (dílčí) předpoklady pravidla splněny (jsou v PP)?
+
aktivace pravidla závěr → PP
cíl otestován
3.1
–
∀ předpoklad (dílčí) se stává novým cílem
3.2

Příklad: ***na ZŘ z odst. 4.3.2***

Diagnóza streptokokové nákazy v krku (zadání viz odst. 4.3.2)

0. cíl := streptokoková nákaza (Pr1, Z1) (pravidlo, závěr)
1. (Pr1, Z1) není v PP ⇒ najdu v BZ Pr1
   dokazuji: (Pr1, P1) – bolení v krku ? není v PP
   nový cíl := (Pr1, P1)
2. (Pr1, P1) není v PP ⇒ najdu v BZ Ø ⇒ (1, 2)
   cíl = primitivum ⇒ dotaz uživateli
   uživatel odpoví: bolení v krku (4)
3. cíl := streptokoková nákaza (Pr1, Z1)
   (Pr1, Z1) není v PP ⇒ najdu v BZ Pr1 (1,2)
   dokazuji: (Pr1, P1) → bolení v krku ? (3)
   (Pr1, P1) – bolení v krku je v PP
4. (Pr1, P2) není v PP (3)
   nový cíl := (Pr1, P2) – očekáváme bakteriální infekci (1)
5. (Pr1, P2) není v PP ⇒ najdu v BZ Pr3 (2)
   dokazuji: (Pr3, P1) – nemoc ≥ 2 měsíce? (3)
   (Pr3, P1) není v PP (3)
   nový cíl := (Pr3, P1) – nemoc ≥ 2 měsíce
6. (Pr3, P1) není v PP ⇒ najdu v BZ Ø (1, 2)
   cíl = primitivem ⇒ dotaz uživateli
   uživatel odpoví: nemoc ≥ 2 měsíce (4)
7. cíl := (Pr1, P2) – očekáváme bakteriální infekci
   (Pr1, P2) není v PP ⇒ najdu v BZ Pr3 (1, 2)
   dokazuji: (Pr3, P1) nemoc ≥ 2 měsíce
   (Pr3, P1) nemoc ≥ 2 měsíce je v PP (3)
8. (Pr3, P2) – pacient má horečku – není v PP (3)
   nový cíl := (Pr3, P2) – pacient má horečku
9. (Pr3, P2) není v PP ⇒ najdu v BZ Pr2 (1,2)
   Dokazuji: (Pr2, P1) – t > 38°C (3)
   (Pr2, P1) není v PP
   nový cíl := (Pr2, P1) – t > 38°C
10. (Pr2, P1) není v PP ⇒ najdu v BZ Ø (1,2)
    cíl = primitivum ⇒ dotaz uživateli
    uživatel odpoví: t > 38°C (4)
11. cíl := (Pr3, P2) – pacient má horečku
    (Pr3, P2) není v PP ⇒ najdu v BZ Pr2 (1,2)
    Dokazuji (Pr2, P1) – t > 38°C (3)
    (Pr2, P1) je v PP ⇒ *aktivace Pr2* (3.1)
12. cíl := očekáváme bakteriální infekci – (Pr1,P2)
    (Pr1, P2) není v PP ⇒ najdu v BZ Pr3 (1,2)
    (Pr3, P1) – pacient nemocen ≥ 2 měsíce – je v PP, (Pr3, P2)
    – pacient má horečku je v PP (3) ⇒ *aktivace Pr3*
13. cíl := streptokoková nákaza (Pr1, Z1)
    (Pr1, Z1) není v PP ⇒ najdu v BZ Pr1 (1,2)
    dokazuji (Pr1, P1) – bolení v krku,
    (Pr1, P2) – očekáváme bakteriální infekci – jsou v PP (3)
    ⇒ *aktivace Pr1*
14. Další cíl neexistuje ⇒ STOP

PP

+ bolení v krku

bolení v krku

+ —’’ — nemoc ≥ 2 měsíce

—’’ — nemoc ≥ 2 měsíce

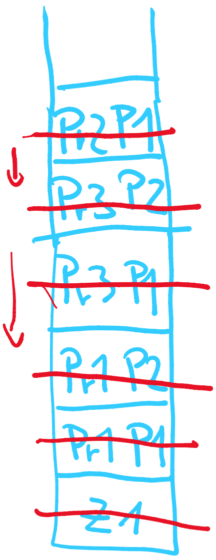
+ —’’ — t > 38 °C

—’’ — t > 38 °C

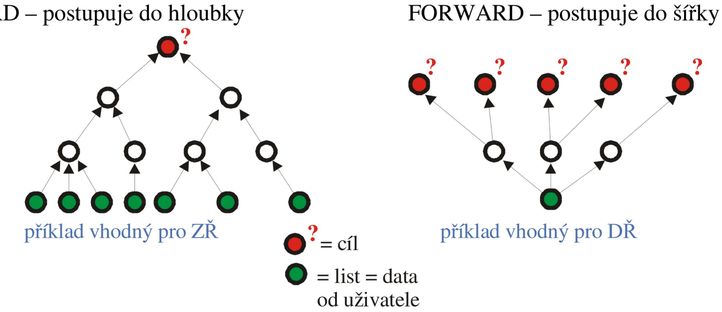
++ —’’ — pacient má horečku

++ —’’ — očekáváme bakt.infekci

++ —’’ — streptokoková nákaza



BACKWARD – postupuje do hloubky FORWARD – postupuje do šířky



Příklad:

P1: IF existují známky infekce v krku
AND organismus je streptokok
THEN streptokoková nákaza v krku ?

P2: IF krk je červený
THEN existují známky infekce v krku

P3: IF organismus je grampozitivní
AND tvar je kok
AND organismus je řetězovitý
THEN organismus je streptokok

Cíl 1: streptokoková nákaza v krku

Pacient: - krk je červený
- organismus je grampozitivní
- tvar je kok
- organismus je řetězovitý 

4.4.2 Agenda cílů

agenda cílů = řada cílů zpracovávaná v určitém pořadí

1. nejjednodušší je seznam cílů
   1. cíl A
   2. cíl B

   …
2. Hierarchický seznam (výhodou je umožnění částečného řešení)

Příklad:

- \+ 1. zvíře je pták
  1.1 zvíře je drozd + konec
  1.2 zvíře je pěnkava -
2. zvíře je savec
- \+ 2.1 zvíře je kůň
  2.2 zvíře je kráva
3. zvíře je plaz

……

3. Strategie (dynamický seznam)
   Po splnění každého jednotlivého cíle se definuje, co se má dělat dál (např. STOP, přejít na jiný cíl, atd.)

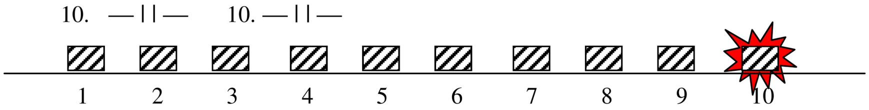
Příklad:

1. kup televizor ⇸ STOP
2. kup rádio ⇸ STOP
3. kup vysavač → STOP

4.4.3 Pořadí cílů dané uživatelem

Příklad:

10 modulů
1. diagnostika 1. modul - OK?
2. —||— 2. —||—
3. —||— 3. —||—
…
10. —||— 10. —||—



4.4.4 Pořadí cílů dané pravidly

Příklad:

1. cíl = zesilovač
2. cíl = chladič

} agenda ≡ IF P>10W THEN jdi na 2.cíl

metapravidlo - množina pravidel z BZ odvozující 1. cíl

cíl metapravidla můžeme přidat do seznamu cílů ⇒ vyšetřujeme nejprve cíle metapravidel

4.5 VÝHODY A NEVÝHODY DŘ A ZŘ

4.5.1 DŘ - výhody

1. výhodné při sběru informace s následným vyhodnocováním
2. z malého množství vstupních informací odvozuje velké množství nových faktů
   IF prší THEN tráva je mokrá
   IF tráva je mokrá THEN nesekat
   IF tráva je mokrá THEN nehrát fotbal
   IF prší THEN deštník
3. výhodné v úlohách plánování

4.5.2 DŘ - nevýhody

1. nepozná důležitost vstupních informací - vyvozuje úplně všechno
2. může klást otázky v nelogickém sledu
3. v případech, kde existuje málo hypotéz, které se mají zodpovědět, existuje velké množství vstupních dat

4.5.3 ZŘ – výhody

1. existuje málo hypotéz
2. otázky klade v logickém sledu
3. hledá jen to, co je potřebné pro splnění cíle
4. výhodné v diagnostice

4.5.4 ZŘ – nevýhody

1. postupuje slepě od cíle (kořene) dolů
2. když existuje velké množství hypotéz a málo vstupních dat

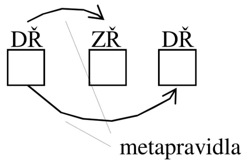
4.6 VÝBĚR MEZI DŘ A ZŘ

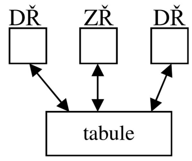
1. Jak úlohu řeší expert (např. málo hypotéz ⇒ ZŘ)
2. Jak je inference řešena jinde
3. Nic se neděje, i když se spleteme

4.7 KOMBINACE DŘ A ZŘ

- 2 způsoby

1. separované moduly





2. ZŘ s démony

démon = démonické pravidlo = pravidlo, které se aktivuje kdykoliv, když jsou splněny jeho předpoklady

Příklad:

Diagnostika tlakové nádoby → ZŘ

1. BZ obsahuje pravidla, která jsou využívána ZŘ
2. Do BZ přidáme tzv. démonická pravidla (_D)
   1D: IF tlak > 1000
   THEN stav ohrožení
   2D: IF stav ohrožení
   THEN poplach
   3D: IF poplach
   THEN siréna, telefon, evakuace
3. Proces ES po spuštění:
   Jakmile dojde k události „tlak > 1000“, aktivuje se pravidlo 1D ⇒ nová událost „stav ohrožení“ ⇒ aktivuje se 2D atd. (1D až 3D jsou řetězeny DŘ)

- práce v reálném čase
- řízení událostmi

4.8 HLEDÁNÍ VE STAVOVÉM PROSTORU

stavový prostor = strom nebo graf, kde uzly = stavy úlohy, hrany = vztahy (předpisy přechodu) mezi uzly

V ES založených na pravidlech jsou uzly = předpoklady nebo závěry pravidel a hrany = pravidla.

IM (inferenční mechanismus) říká, jak procházet graf (stavový prostor), abychom co nejrychleji (a nejlépe) dospěli k cíli (cílovému uzlu, cílové hypotéze).

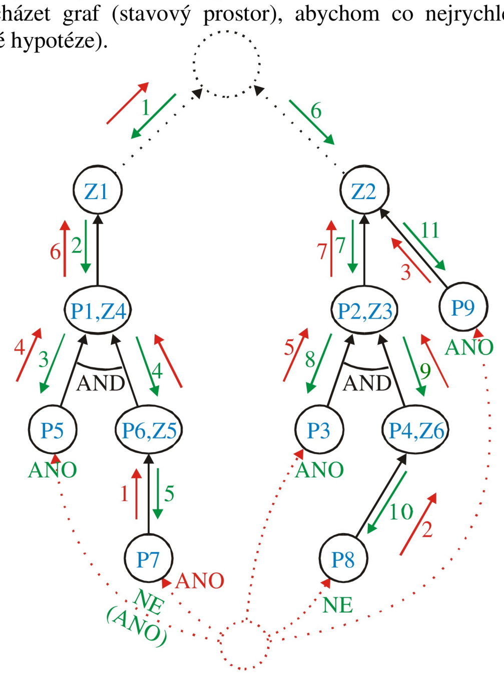
Příklad:

Servírovat bílé či červené?

1. IF objednáno maso P1
   THEN červené víno Z1
2. IF objednána ryba P2
   THEN bílé víno Z2
3. IF mají rybu P3
   AND máme dost peněz na rybu P4
   THEN objednáme rybu Z3
4. IF mají maso P5
   AND máme dost peněz na maso P6
   THEN objednáme maso Z4
5. IF máme ≥ 300 Kč P7
   THEN máme dost peněz na maso Z5
6. IF máme ≥ 150 Kč P8
   THEN máme dost peněz na rybu Z6
7. IF skutečně chceme bílé víno P9
   THEN bílé víno Z2

ZŘ … prohledávání do hloubky

DŘ … prohledávání do šířky



Inferenční síť (speciální AND/OR stromy)
(ukázka sleposti ZŘ ⇔ 11 kroků)

pozn: Pro demonstraci je BZ procházena až do konce!

4.8.1 Prohledávání stavového prostoru do hloubky

Viz. • v předchozím příkladu, používá ho ZŘ (od cíle k uzlům = primitivům)

prohledávání stavového prostoru do hloubky = technika hledání, která prohledává stavový prostor tak, že prohledává dříve vždy uzly s větší či stejnou hloubkou, než je hloubka uzlů ostatních

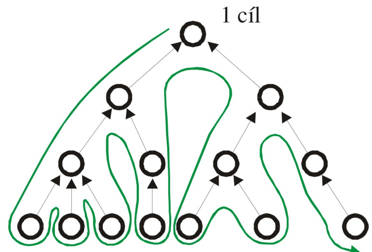
hloubka uzlu = počet hran cesty vedoucí od kořene k uzlu

Výhody:

1. zaručuje řešení – v případě konečného počtu stavů
2. rychlé v případech, kdy je výhodné použít ZŘ
3. zaostřené na cíl
4. interakce s uživatelem v logickém sledu

Nevýhody:

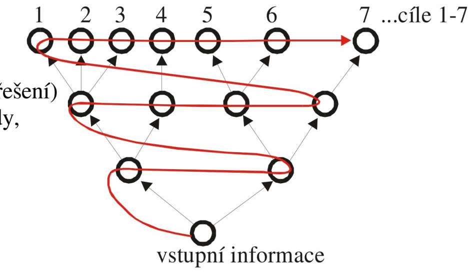
1. slepé
2. nevhodné pro rozsáhlý stavový prostor
3. nevhodné v případech, kdy je výhodné použít DŘ



4.8.2 Prohledávání stavového prostoru do šířky

Viz • v předchozím příkladu, používá ho DŘ

prohledávání stavového prostoru do šířky = technika hledání, která prohledává stavový prostor tak, že prohledává dříve vždy uzly s menší či stejnou hloubkou než je hloubka uzlů ostatních



Výhody:

1. zaručuje řešení
2. rychlé v případech, kdy je výhodné použít DŘ („mělké" řešení)
3. najde minimální cestu, potřebujeme-li ji (platí pouze tehdy, když cena cesty je rovna počtu hran!!)

Nevýhody:

1. slepé
2. nevhodné pro rozsáhlý stavový prostor
3. nevhodné v případech, kdy je výhodné použít ZŘ
4. ↓ interakce s uživatelem

4.8.3 Prohledávání stavového prostoru heuristickým prohledáváním

heuristické prohledávání = první nejlepší prohledávání = technika hledání, která užívá znalostí o problému k vedení hledání

V ES se používá heuristik k:

1. pořadí cílů
2. pořadí předpokladů pravidel
3. definování metapravidel
4. stanovení priorit pravidel
5. stanovení CF (= certainty factor = činitel jistoty) pravidel

Výhody:

1. není slepé
2. rychlejší

Nevýhody:

- nemusí nalézt řešení (přestože řešení existuje)

4.9 MONOTÓNNÍ A / NEBO NEMONOTÓNNÍ INFERENCE

4.9.1 Monotónní

monotónní uvažování = metoda uvažování, která předpokládá, že jakmile nastal nějaký fakt, nemůže být během úlohy již změněn

Příklad:

IF T1 je špatný
THEN nahraď T1 novým (T1 je pořád špatný)

4.9.2 Nemonotónní

nemonotónní uvažování = metoda uvažování, která povoluje změny ve faktech a tím i všech závěrů z fakt doposud vyplynulých

Příklad:

IF prší
THEN vem si deštník
(pak přestane pršet)

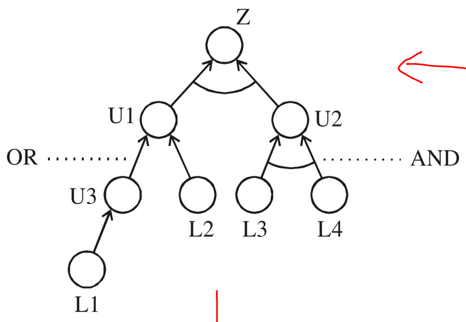
4.9.3 Monotónní a nemonotónní

Příklad:

IF vypínač zapnut
THEN světlo svítí

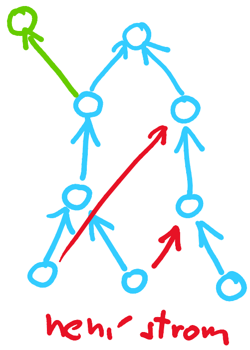
IF světlo svítí
THEN film zničen
(vypínač zapnu a pak vypnu)
(nemonotónní s nevratnými fakty)

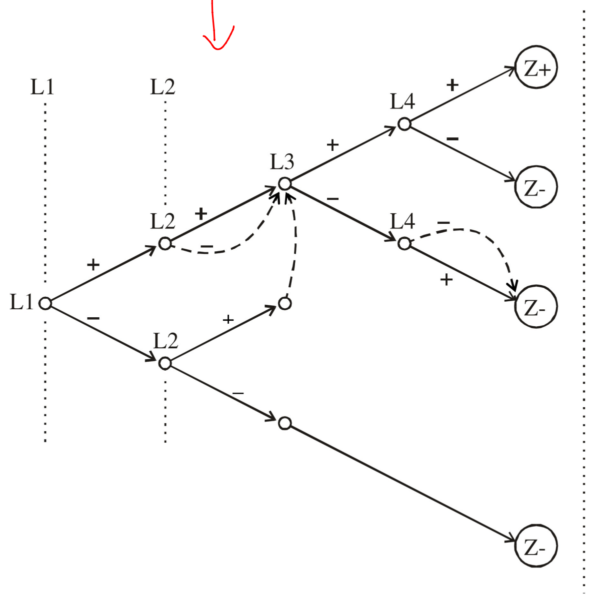
5 ROZHODOVACÍ STROMY, AND/OR STROMY A PRAVIDLA



$\Rightarrow$

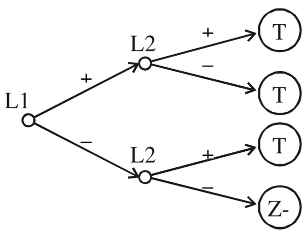
$L1 \Rightarrow U3$
$U3 \vee L2 \Rightarrow U1$
$L3 \wedge L4 \Rightarrow U2$
$U1 \wedge U2 \Rightarrow Z$

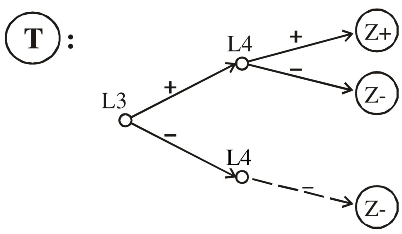


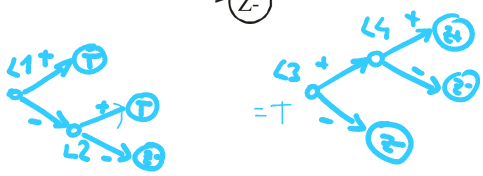


$16 = 2^4$

- záleží na pořadí L1, L2, L3, L4
- $\uparrow \Rightarrow$ více stromů







IF L1-	IF L1+	IF L1+	IF L1-	IF T	IF T	IF T
AND L2-	AND L2+	AND L2-	AND L2+	AND L3+	AND L3+	AND L3-
THEN Z-	THEN T	THEN T	THEN T	AND L4+	AND L4-	THEN Z-
				THEN Z+	THEN Z-	

6 MYCIN

(Shortlitte, E.H.: Computer - Based Medical Consultation, MYCIN, American Elserier, New York, 1976) meningitis, bakteriémie

- požadavek lidského experta
- úzce vymezená oblast
- neurčitost – výsledky testů nejsou ještě k dispozici, ale je třeba léčit
- včasnost diagnosy
- stanovení léčby (u lékaře bylo:
  - špatné dávkování v 66%
  - špatná kombinace léků v 62%)

6.1 HLAVNÍ RYSY

6.1.1 ZŘ, ≈ 500 pravidel, SW: INTERLISP:

Příklad:

```
      IF (AND (SAME CONTEXT GRAMNEG)
              (SAME CONTEXT MORPHROD)
              (SAME CONTEXT AIRAEROBIC))
    THEN (CONCLUDE CONTEXT CLASS
              ENTEROBACTERIACEAE TALLY 0.8)
```

6.1.2 BZ a IM oddělené

6.1.3 Metapravidla

IF absces v oblasti …
AND ∃ pravidla stanovující grampozitivitu
AND ∃ pravidla stanovující, zda-li se jedná o čeleď Enterobacteriaceae
THEN přednostně prohledej pravidla čeledi Enterobacteriaceae

6.1.4 Neurčitost

Činitel jistoty CF $\in \langle -1, 1 \rangle$

6.1.5 Zapamatování si minulého sezení (konzultace)

6.1.6 Přizpůsobení uživateli:

Příklad.:

ES se táže, o jakou infekci se jedná?
uživatel: Endurteritis
ES: Myslíte Endarteritis?

6.1.7 Alternativní léčba

- vyžádaná uživatelem (lékařem)

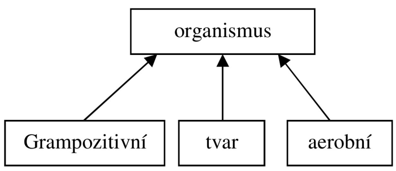
6.2 PŘÍSTUP K ŘEŠENÍ PROBLÉMU

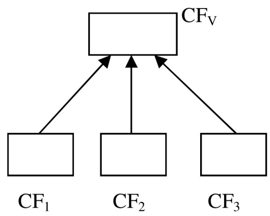
problém:
- stanovení diagnózy
- stanovení léčby

1. řeší se stanovení diagnózy
   1.1 určení infekce
   1.2 určení druhu organismu
2. řeší se stanovení léčby
   3 požadavky: - pokrytí organismů antibiotiky
   - co nejmenší počet antibiotik
   - bezpečnost léčby

6.2.1 Diagnóza

1. standardní posloupnost otázek
   1. jméno?
   2. věk?
   3. ………
2. Byly provedeny testy v laboratoři?
3. (Testy provedeny $\Rightarrow$ výsledky známy):
   - známá infekce (neptá se na symptomy nemoci)
4. (Testy nebyly):
   - určení infekce na základě symptomů nemoci
5. Infekce známá, pak stanovení druhu organismu



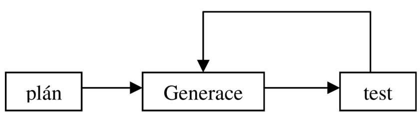


6. Výsledek diagnózy:
   1. organismus 1 … CF 1
   2. organismus 2 … CF 2
   ………

6.2.2 Léčba

- seznam organismů (A, B, C)
- seznam léků pro organismy (L1, L2, …)

léčba	A	B	C
1	L1	L2	L1
2	L3	L1	L2
3	…	…	…



- Generace : L1 .... předpis 1
  L1,L2 .... předpis 2
- Test – jestli eliminuje všechny organismy

Dále se provede redukce léků:

L1 třída 1 $\Big\}$ L1$\Leftrightarrow$ L2
L2 třída 1
…
L3 třída 2

The MYCIN system currently can treat the following infections: BACTEREMIA MENINGITIS CYSTITIS. Because the knowledge base is incomplete, all other infections will be treated like bacteremia.
Z
<http://www.computing.surrey.ac.uk/ai/PROFILE/mycin.html#Certainity%20Factors>

výsledek redukce: L1
- test bezpečnosti léčby: (+věk, pohlaví, alergie, …); ze všech předpisů vybere 1 lék

Výsledek léčby
1. AMPICILIN dávkování 1
2. PENICILIN dávkování 2

Lékař může změnit seznam doporučených léků ⇒ předloží alternativu systém:
systém otestuje alternativu:
1. test pokrytí organismů
2. ~~rudukce~~ redukce
3. test na bezpečnost

6.3 VYHODNOCENÍ

10 případů nemocných
9 „odborníků" – lidí + 1 ES MYCIN = 10

	% adekvátnost léčby	počet případů, ve kterých se nemoc nepokryla (léčbou)
MYCIN	70	0
přední odborník	70	0
Fak. nemocnice 1	50	1
Fak. nemocnice 2	50	1
vedoucí	50	1
Fak. nemocnice 4	50	0
Fak. nemocnice 3	40	0
Praktik	30	1
Fak. nemocnice 5	30	0
Student	10	3

6.4 PŘÍKLAD PRÁCE MYCINU

1. ES: Jméno pacienta? SPEJBL
2. Věk 60
3. Pohlaví M
4. Rasa (běloch,…) běloch
5. Byly nalezeny kultury pro testování v laboratoři? ANO
6. Z jakého místa byl sebrán vzorek? esf
7. V jakém čase vzorek pořízen? čas
8. Byl vzorek prozkoumán? ANO
9. Našly se kultury? ANO
10. Jste schopen z testů obdržet informaci o infekci? ANO
11. Jaká je infekce? Budartevitis
   - kdyby nebyly k dispozici laboratorní testy
      - neznala by se infekce
      - ES: Pacient trpí závratěmi?
        Bolí ho hlava?
        …
      ⇓
      infekce = _______
12. Zadej organismus 1: NEZNÁMÝ
13. Je organismus tyčinka nebo kok?
14. Je organismus Grampozitivní? ANO
15. Aerobicita organismu? fakultativní
16. Je infekce zjištěna během hospitalizace?

Proč?
…
NE
17. Jsou ještě jiné organismy, o kterých se chcete radit, přítomny? ANO
18. Zadejte organismus 2: SALMONELLA (0,4)
E. COL (0,3)

↓
ES: infekce 1 je endarteritis s bakteriemií

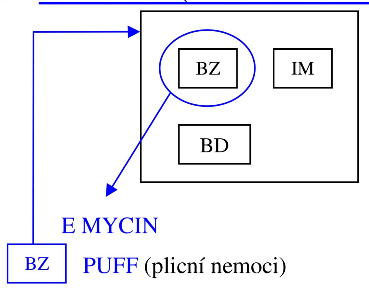
Položka 1: E. COL	0,71
Položka 2: SALMONELLA	0,60
Položka 3: PNEUMONIA	0,25

↓Otázky PROČ

19. Má SPEJBL alergickou reakci na antibiotika? NE
20. Preferovaná léčba je:
    Pro pokrytí položek 1,2,3:
    1. AMPICILLIN 2,5g
    2. CENTAMICIN 131mg po 10 dní

    Poznámka: Monitorovat koncentraci séra.
    …
21. Chcete vidět alternativní léčby: ANO
22. Jaký lék vypustit? AMP.
23. 2. léčba je:
    - pro pokrytí položek 2,3
      je …

6.5 E-MYCIN (EMPTY - MYCIN)



7 PŘÍKLADY ES ZALOŽENÝCH NA PRAVIDLECH

(MYCIN)

7.1 VÝHODY ES ZALOŽENÝCH NA PRAVIDLECH

1. přirozenost vyjádření znalostí
2. přirozená separace znalostí a IM
3. modularita znalostí
4. snadná rozšiřitelnost (přidání, modifikace, ubrání pravidel)
5. proporcionální růst inteligence
6. užívá relevantních znalostí k problému
7. snadná realizace vysvětlovacího mechanismu
8. snadný test konzistence systému:
   IF A THEN B
   IF A THEN B'
9. Neurčitost – snadno:

IF A THEN B CF
10. Práce s proměnnými
11. Metapravidla

7.2 NEVÝHODY ES ZALOŽENÝCH NA PRAVIDLECH

1. Vyžadují přesné srovnání (částí pravidel)
   A→B B→C
2. Neprůhledné vztahy mezi pravidly
   A→C B→D C→B
   (⇒ grafická interpretace)
3. mohou být pomalé

7.3 PŘÍKLAD DIAGNOSTICKÉHO SYSTÉMU V MEDICÍNĚ (PODOBNÝ MYCINU) (ZŘ)

1. stanovení diagnózy (6.1) omezená na meningitis
2. stanovení léčby (viz. 6.2)
3. stanovení alternativní léčby (viz. 6.3)

1.cíl: infekce je meningitis

INFEKCE MENINGITIS

R1: IF lékař ví, že se jedná o meningitis
OR nemoc má rysy meningitis
THEN infekce je meningitis
AND DISPLAY infekce je meningitis
ELSE DISPLAY není meningitis

RYSY MENINGITIS

R2: IF laboratorní testy ukazují na meningitis
OR klinické vyšetření ukazuje na meningitis
THEN nemoc má rysy meningitis

LABORATORNÍ TESTY

R3: IF test proveden
AND zjištěny kultury mikroor.
AND kultury jsou podobné meningitis
THEN laboratorní testy ukazují na meningitis

KULTURY JSOU PODOBNÉ MENINGITIS

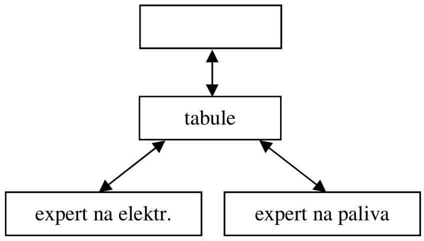
R4: IF organismus má tvar koku
AND organismus je grampositivní
THEN kultury jsou podobné meningitis

KLINICKÉ PŘÍZNAKY

R5: IF pacient má persistentní bolení hlavy
AND pacient má závratě
AND pacient je letargický
THEN klinické vyšetření ukazuje na meningitis

7.4 PŘÍKLAD ZŘ, TABULE (BLACKBOARD)

Diagnostika automobilu



{1.ES – šéf mechanik}

$BLACK1.PRL … začlení soubor tabule

- inicializace proměnných pro ∀ aktivaci dílčích ES (šéf, elektrikář, palivář)

{INIT}

REINIT	cena_baterie	1250
AND	cena_startéru	1850
AND	cena_paliva	50
…		
AND	cena_default	…

- inicializace proměnných na začátku spuštění programu

v tabuli

INIT	oblast_problému	IS	neznámá
AND	závada	IS	neznámá
AND	řešení	IS	neznámá
AND	počáteční informace	není k dispozici	

{Cíle}

1. Doporučení šéfa IS elektrický systém / palivový systém

1.1. Cena_určena

1.1.1. Vypsat výsledky

{Určení závady}

P1: EL. SYSTÉM

IF problém je patrně v el. systému
THEN Doporučení šéfa IS el. systém
AND Oblast problému IS el. systém
AND DISPLAY
AND CHAIN ELEKTRIKÁŘ předá se řízení elektrickému systému

P2: PAL. SYSTÉM

analogicky k P1
…
…CHAIN PALIVÁŘ

P3: STARTÉR_netočí

IF motor IS nenastartuje
AND startér IS netočí
THEN problém je patrně v el. systému

P4: STARTÉR_točí
IF motor IS nenastartuje
AND startér IS točí
AND motor se otáčí IS pomalu
THEN problém je patrně v el. systému
P5: Motor_běží špatně
IF motor IS nastartuje
AND motor běží nečistě
THEN problém je patrně v el. systému
P6: Motor_se točí normálně
IF motor IS nenastartuje
AND motor se otáčí IS normálně (při startování)
THEN problém je patrně v pal. systému
P7: Zrychlení je špatné
IF auto váhá při zrychlení
THEN problém je patrně v pal. systému

{Určení ceny}
P8: Cena paliva
IF Řešení IS přidat palivo
THEN cena_opravy = cena_paliva
AND cena_určena (cíl 1.1)
P9: cena…
IF řešení IS __________
THEN cena_opravy = cena__________
AND cena_určena
P12:
P13: IF Řešení IS neznámé
THEN cena_opravy = default
AND cena_určena
Výsledky______AND DISPLAY NENALEZENO ŘEŠENÍ
P14: IF cena_určena
THEN vypsat výsledky (cíl 1.1.1)
AND DISPLAY VÝSLEDKY …

Pořadí
P8-P13

{2.ES na palivo}
$BLACK1.PRL
{cíle}
1. Určení závady v palivovém systému
1.1. Předat výsledky šéfovi
P1: Není palivo
IF motor IS nenastartuje
AND startér IS točí
AND paliv.ukazatel IS nehýbe se
THEN Určení závady v pal. systému
AND Závada IS není palivo
AND Řešení IS přidat palivo
P2: karburátor …
IF …
AND …
AND …
P5: THEN Určení závady v pal. systému
AND Závada IS __________
AND Řešení IS __________
P6: IF CHAIN ŠÉF

THEN předat výsledky šéfovi

Obdobně:
{3.ES na el. systém}
$BLACK1.PRL
{cíle}
1. Určení závady v el. systému
1.1. Předat výsledky šéfovi
P1: IF …
AND …
AND …
THEN Určení závady v el. systému
P5: AND Závada IS _________
AND Řešení IS _________
P6: IF CHAIN ŠÉF
THEN předat výsledky šéfovi

8 METODY PRÁCE S NEURČITOSTÍ V ES

Druhy neurčitosti

E→C

BD 1. v datech od uživatele P(E|e) … e – víra uživatele v platnost E

$e \to E$
uživatel má v hlavě

BZ 2. v BZ (pravidlech) od experta
IF E
THEN C (CF = 0,8)

P(C|E) ~~P(C|E,e)~~ ?

IM 3. v nekompatibilitě pravidel

IF E1	IF E1 AND E2
THEN C	THEN C
IF E1	IF E1
THEN C CF1	THEN $\overline{C}$ CF2

Neexistuje jednotná teorie, jak pracovat s neurčitostí v ES. Existuje tedy několik přístupů k tomuto problému. Každý z přístupů vede k obecně odlišným závěrům (např. jiné „váhy" závěrů).
4. Neurčitost závěru je ovlivněna přístupem k práci s neurčitostí, který je použit.
metaneurčitost

Druhy přístupů
1. Bayesův přístup a jeho modifikace (PROSPECTOR)
2. Teorie určitosti (Certainty Theory)
3. Fuzzy logika
4. Dempster – Shaferova teorie

8.1 BAYESŮV PŘÍSTUP K NEPŘESNÉMU USUZOVÁNÍ

8.1.1 Opakování

$E \to H$
P(E) … apriorní pravděpodobnost jevu E
P(H) … apriorní pravděpodobnost jevu H
P(H|E) … aposteriorní pravděpodobnost závěru (hypotézy) H
P(E|H) … podmíněná pravděpodobnost jevu E za podmínky, že nastal jev H

$$\boxed{P(H \mid E) = \frac{P(E \mid H)P(H)}{P(E)}}$$

$$P(H,E) = P(E,H) = P(E|H)P(H) = P(H|E)P(E)$$

$= P(E)\cdot P(H)$

↑ pokud E a H jsou nezávislé

$$P(E)=\sum_i P(E\mid H_i)P(H_i)=\sum_i p(E,H_i)=P(E)$$

$$H_i = \left\langle \begin{array}{ll} H & i=0 \\ \overline{H} & i=1 \end{array}\right.$$

$$\boxed{P(E)=P(E\mid H)P(H)+P(E\mid \overline{H})P(\overline{H})}$$

Příklad:

Pacient s bolestmi na hrudi:

má: — NS (nemocné srdce): + (ano, má NS)

- (ne nemá NS)

\ EKG + (ano, EKG je pozitivní)

- (ne, EKG je negativní)

Za předpokladu, že EKG je + (+EKG) se ptáme, zda pacient má skutečně NS, tj. jaká je pravděpodobnost P(+NS | +EKG) = ?

Přitom máme statisticky zjištěno, že:

P(+NS) … apriorní pravděpodobnost jevu, že pacient má nemocné srdce = 0,1

P(+EKG | +NS) … podmíněná pravděpodobnost jevu, že EKG je pozitivní u pacienta s NS = 0,9

P(-EKG | -NS) … = 0,95

Řešení:

$$P(+NS|+EKG)=\frac{P(+EKG\mid +NS)P(+NS)}{P(+EKG)}=$$

$$=\frac{P(+EKG\mid +NS)P(+NS)}{P(+EKG\mid +NS)P(+NS)+P(+EKG\mid -NS)P(-NS)}=$$

$$=\frac{P(+EKG\mid +NS)P(+NS)}{P(+EKG\mid +NS)P(+NS)+[1-P(-EKG\mid -NS)][1-P(+NS)]}$$

$$=\frac{0{,}9\cdot 0{,}1}{0{,}9\cdot 0{,}1+[1-0{,}95]\cdot[1-0{,}1]}\doteq 0{,}67\doteq\frac{2}{3}$$

Odpověď:

2/3 lidí s bolestmi na hrudi, kterým se naměřilo pozitivní EKG, má ve skutečnosti nemocné srdce. (Tzn., že 1/3 lidí s bolestmi na hrudi a pozitivním EKG má srdce zdravé.)

8.1.2 Bayesův teorém v ES

8.1.2.1 Šance

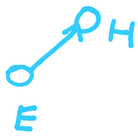
Šance 2/3 : 1/3 = 2 : 1 odpovídá tedy pravděpodobnosti ≈ 0,67 ≈ 2/3.

Šance, že má pacient nemocné srdce za předpokladu, že EKG je pozitivní, je tedy 2.

Šance jevu H - apriorní O(H) ↔ P(H)

- aposteriorní O(H|E) (za podmínky E) ↔ P(H|E) apriorní šance: $\boxed{O(H) = \frac{P(H)}{P(\overline{H})} = \frac{P(H)}{1 - P(H)}}$

aposteriorní šance: $\boxed{O(H \mid E) = \frac{P(H \mid E)}{P(\overline{H} \mid E)} = \frac{P(H \mid E)}{1 - P(H \mid E)}}$



$O \in \langle 0, \infty)$

Věta:

$\boxed{P(H) = \frac{O(H)}{1 + O(H)}}$

Důkaz: $O = \frac{P}{1-P} = \frac{1}{1/P - 1}$

$$\frac{1}{O} = \frac{1}{P} - 1$$

$$P = \frac{1}{1 + 1/O} = \frac{O}{1 + O}$$

Příklad:

Spočítejte apriorní a aposteriorní šanci k P(+NS) a P(+NS|+EKG).

O(+NS) = $\frac{1}{9}$

O(+NS|+EKG) = $\frac{2}{1}$ = 2

8.1.2.2 Míra postačitelnosti a míra nezbytnosti

Míra postačitelnosti (likelihood of sufficiency) LS:

$\boxed{O(H \mid E) = LS \cdot O(H)}$

Míra nezbytnosti (likelihood of neccesity) LN:

$\boxed{O(H \mid \overline{E}) = LN \cdot O(H)}$

$$\boxed{LS = \frac{O(H \mid E)}{O(H)}}$$

$$\boxed{LN = \frac{O(H \mid \overline{E})}{O(H)}}$$

Když E $\rightarrow$ H s velkou pravděpodobností, pak LS$\uparrow$ (LS $\rightarrow \infty$)

Když $\overline{\text{E}} \rightarrow$ H s malou pravděpodobností, pak LN$\downarrow$ (LN $\rightarrow$0)

Tj. je-li LS$\uparrow^{\infty}$, je O(H|E) $\uparrow$ bez ohledu na O(H), E je tedy postačitelné pro H

je-li LN$\downarrow_0$, je O(H|$\overline{\text{E}}$) $\downarrow$ bez ohledu na O(H), stačí tedy, aby platilo $\overline{\text{E}}$, a pak H neplatí. Aby tedy H mohla platit, musí platit ($\overline{\overline{\text{E}}}$) => musí platit E. E je tedy nezbytné pro to, aby H mohla platit.

Věta:

$$\boxed{LS = \frac{P(E \mid H)}{P(E \mid \overline{H})} \qquad LN = \frac{P(\overline{E} \mid H)}{P(\overline{E} \mid \overline{H})}}$$

Důkaz:

$$LS = \frac{O(H \mid E)}{O(H)} = \frac{\dfrac{P(H \mid E)}{P(\overline{H} \mid E)}}{\dfrac{P(H)}{P(\overline{H})}} = \frac{P(H \mid E)}{P(H)} \frac{P(\overline{H})}{P(\overline{H} \mid E)} = \frac{P(E \mid H)P(H)}{P(E)P(H)} \frac{P(\overline{H})P(E)}{P(E \mid \overline{H})P(\overline{H})} = \frac{P(E \mid H)}{P(E \mid \overline{H})}$$

LN … obdobně záměnou E za $\overline{\text{E}}$

Tabulka:

E → H

$O(H \mid E) = LS \cdot O(H)$

$O(H \mid \overline{E}) = LN \cdot O(H)$

! zapamatuj si !

LS		LN	
0	$\overline{E}$ je nezbytné pro H	∞	$\overline{E}$ je postačitelné pro H
1	*E* ani $\overline{E}$ nemá žádný vliv na H	1	*E* ani $\overline{E}$ nemá žádný vliv na H
∞	*E* je postačitelné pro H	0	*E* je nezbytné pro H

Platí:

LS > 1 ⇔ LN < 1

LS = 1 ⇔ LN = 1

LS < 1 ⇔ LN > 1

$= P(E|H)$

Neboť platí:

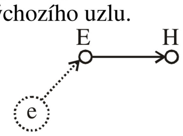
$$LN = \frac{1 - LS \cdot P(E \mid \overline{H})}{1 - P(E \mid \overline{H})} = \frac{P(\overline{E}|H)}{P(\overline{E}|\overline{H})} \qquad P(E|\overline{H}) \cdot LS = \frac{P(E|H)}{\cancel{P(E|\overline{H})}} \cdot \cancel{P(E|\overline{H})}$$

Důkaz plyne z předchozí věty

Toto však platí jen teoreticky. Praktickým požadavkem totiž je, že expert ke každému pravidlu zadává též LS i LN a apriorní pravděpodobnosti, které ale nemusí splňovat uvedený vztah (např. zadá-li LN = 2, LS = 1).

8.1.3 Použití Bayesova teorému v PROSPECTORu

V inferenční síti jsou ke každému přechodu z uzlu do uzlu (tj. ke každému pravidlu) přiřazeny vztahy pro výpočet pravděpodobnosti faktu odpovídajícího uzlu, do kterého se přejde. Přitom se použije pravděpodobnost faktu odpovídajícího výchozího uzlu.

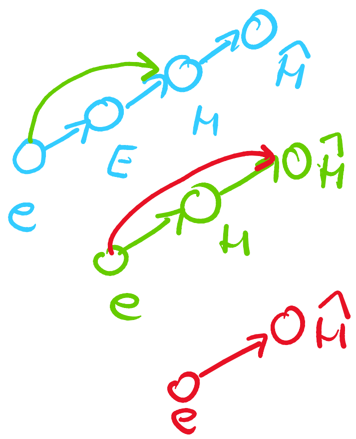


$$P(H \mid e) = ? = f\left[P(H \mid E), P(H \mid \overline{E}), P(E \mid e)\right]$$

e e

musí být dáno ke každému pravidlu již při konstrukci BZ expertem

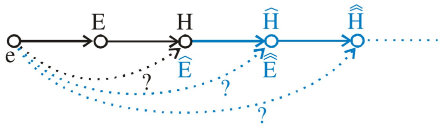
= vstup do vztahu, vyplývá buď z předchozího pravidla anebo je zadáno uživatelem (v případě listu), pak e představuje uživatelovu indicii (víru v to, že platí E)



(8.1.1): $P(H \mid e) = P(H \mid E, e)P(E \mid e) + P(H \mid \overline{E}, e)P(\overline{E} \mid e)$

1. aproximace PROSPECTORu $\doteq \underbrace{P(H \mid E)}_{\text{expert}}\underbrace{P(E \mid e)}_{\text{předchozí pravidlo či uživatel}} + \underbrace{P(H \mid \overline{E})}_{\text{expert}}\underbrace{P(\overline{E} \mid e)}_{\text{předchozí pravidlo či uživatel}}$

V inferenční síti pak dochází k postupné propagaci



$P(H \mid e) = f(P(E \mid e))$

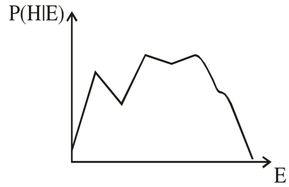
$P(\hat{H} \mid e) = \hat{f}(P(\hat{E} \mid e)) = \hat{f}(f(P(E \mid e)))$

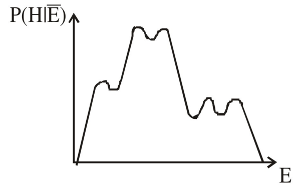
$P(\hat{\hat{H}} \mid e) = \hat{\hat{f}}(P(\hat{\hat{E}} \mid e)) = \hat{\hat{f}}(\hat{f}(P(\hat{E} \mid e))) = \hat{\hat{f}}(\hat{f}(f(P(E \mid e))))$

…

Protože je pro experta obtížné určit $P(H \mid E)$ a $P(H \mid \overline{E})$ pro všechny případy E, tj.

Obecné rozdělení:





Speciálně pro jev E nabývající pouze 2 hodnot (hod korunou) by měl expert zadat 2 čísla $\underline{P(H \mid E)}$ a $\underline{P(H \mid \overline{E})}$. Expert ale zadává místo $P(H \mid E)$ a $P(H \mid \overline{E})$ LS a LN (a protože ty nestačí na určení $P(H \mid E)$ a $P(H \mid \overline{E})$ (je potřeba znalosti P(H)), též zadává P(E) a P(H)). (P(E) z důvodu zřetězení pravidel) Určení $P(H \mid E)$ a $P(H \mid \overline{E})$ z LS, LN pak provedeme následovně:

Věta:

$$\boxed{P(H \mid E) = \frac{O(H \mid E)}{1 + O(H \mid E)}}$$

Pak:

$$P(H \mid E) = \frac{LS \cdot O(H)}{1 + LS \cdot O(H)} = \frac{LS \cdot P(H)/[1 - P(H)]}{1 + LS \cdot P(H)/[1 - P(H)]} \qquad O(H \mid E) = LS \cdot O(H)$$

$$P(H \mid \overline{E}) = \frac{LN \cdot O(H)}{1 + LN \cdot O(H)} = \frac{LN \cdot P(H)/[1 - P(H)]}{1 + LN \cdot P(H)/[1 - P(H)]} \qquad O(H \mid \overline{E}) = LN \cdot O(H)$$

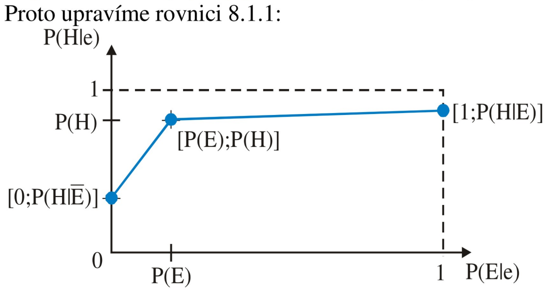
Zadávání P(E) pak plyne z faktu, že vlivem řetězení zadáním P(H) zároveň zadáváme P(Ê) dalšího pravidla, atd.

Problém je, že nahrazujeme P(H|E) → LS, LN a ty jsou navzájem obecně nekonzistentní (⇐ expert zadává např. LS = 1, LN < 1)

Pro případ P(H|E) na LS, LN a obráceně potřebujeme ještě znát P(H), P(E) – zadává expert

Expert zadává: P(H), P(E), LS, LN ⇒ 3 body na přímce f(H|e) = f(P(E|e)) ⇒ f obecně není přímka

Proto upravíme rovnici 8.1.1:

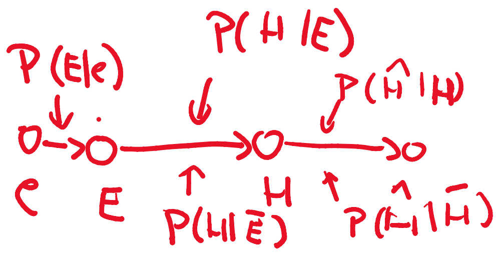


←plyne z 1.aproximace prospectoru

$P(H|E) \quad P(E|e) \quad P(H|\bar{E})\, P(\bar{E}|e)$

$$P(H|e) = P(H,E|e) + P(H,\bar{E}|e)$$

$$= P(H|E,e)\cdot P(E|e) + P(H|\bar{E},e)\cdot P(\bar{E}|e)$$



H, e jsou podmíněně nezávislé při zadaném E

$$P(A,B) = P(A|B)\cdot P(B) = P(B|A)\cdot P(A)$$

$$= P(A)\cdot P(B)$$

A, B nezávislé

$$P(A,B|C) = P(A|B,C)\cdot P(B|C) = P(B|A,C)\cdot P(A|C)$$

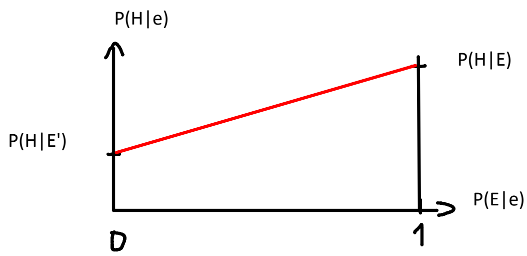
$$= P(A|C)\cdot P(B|C)$$

A, B podmíněně nezávislé při daném C

$$\underbrace{P(H|e)}_{y} = \underbrace{P(H|E)}_{b}\, \underbrace{P(E|e)}_{x} + \underbrace{P(H|E')}_{a}\, \underbrace{P(E'|e)}_{(1-x)}$$

$$y = bx + a - ax$$

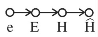
$$y = a + (b-a)x$$



… 2. aproximace PROSPECTORu pramenící z přeurčenosti údajů o neurčitosti zadávaných expertem

8.1.1 → 8.1.2:

(8.1.2): rovnice propagace



Pro $P(E|e) \in \langle 0; P(E) \rangle$

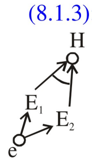
$$\underbrace{P(H|e)}_{y} = \underbrace{P(H|\bar{E})}_{=a} + \frac{\overset{=x}{P(E|e)}}{P(E)}\underbrace{\left[P(H) - P(H|\bar{E})\right]}_{=b}$$

Pro $P(E|e) \in \langle P(E); 1 \rangle$

$$\underbrace{P(H|e)}_{y} = \underbrace{P(H)}_{=a} + \frac{\overset{=x-x_0}{P(E|e) - P(E)}}{1 - P(E)}\underbrace{\left[\underbrace{P(H|E)}_{\text{expert}} - P(H)\right]}_{=b}$$

1. heuristika PROSPECTORu:

(8.1.3): Pravidla s logicky vázanými předpoklady:



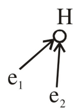
IF E1 AND E2 … THEN H

IF E1 OR E2 … THEN H

$$P(E|e) = \begin{matrix}\min \\ \max\end{matrix}\{P(E_i|e)\} \quad \text{pro} \quad \begin{matrix}\text{AND} \\ \text{OR}\end{matrix}$$

2. heuristika PROSPECTORu:

(8.1.4): Kombinace pravidel

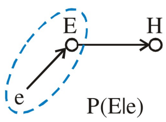


$O(H|e_1, e_2, \ldots, e_n) \leftrightarrow P(H|e_1, e_2, \ldots, e_n)$

$$O(H|e_1,e_2,\ldots,e_n) \stackrel{\blacktriangle}{=} \left(\prod_{i=1}^{n} LE_i\right) O(H), \text{ kde } LE_i \stackrel{\blacktriangle}{=} \frac{O(H|e_i)}{O(H)} \text{ je efektivní poměr}$$

$$LE_i = \frac{O(H|e_i)}{O(H)} = \frac{P(e_i|H)}{P(e_i|\bar{H})}$$

Důkaz:



$$O(H|e_i) = \frac{P(H|e_i)}{P(\bar{H}|e_i)}$$

$$O(H) = \frac{P(H)}{P(\bar{H})}$$

$$LE_i = \frac{P(H|e_i)P(\bar{H})}{P(\bar{H}|e_i)P(H)} \cdot \frac{P(e_i)}{P(e_i)} = \frac{P(H,e_i)}{P(H)}\frac{P(\bar{H})}{P(\bar{H},e_i)} = \frac{P(e_i|H)}{P(e_i|\bar{H})}$$

(8.1.5): Míra určitosti:

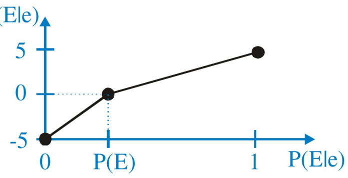
slouží k zadávání P(E|e) od uživatele (pro uživatele) v přijatelné formě

Pro $P(E|e) > P(E)$:

$$C(E|e) = 5\frac{P(E|e) - P(E)}{1 - P(E)}$$

Pro $P(E|e) \le P(E)$:

$$C(E|e) = 5\frac{P(E|e) - P(E)}{P(E)}$$



Místo P(E|e) se udává C(E|e) ∈ ⟨-5, 5⟩

P(E|e) = ? f(C(E|e))

(8.1.6): Pro C(E|e) > 0 :

$$P(E\mid e)=C(E\mid e)\frac{1}{5}\left(1-P(E)\right)+P(E)$$

Pro C(E|e) ≤ 0 :

$$P(E\mid e)=C(E\mid e)\frac{1}{5}P(E)+P(E)$$

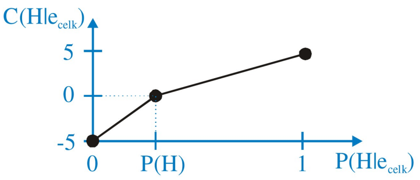
$e_{celk}=e_1\wedge e_2\wedge\ldots\wedge e_n$

Pro $P(H|e_{celk}) > P(H)$:

$$C(H\mid e_{celk})=5\frac{P(H\mid e_{celk})-P(H)}{1-P(H)}$$

Pro $P(H|e_{celk}) \le P(H)$:

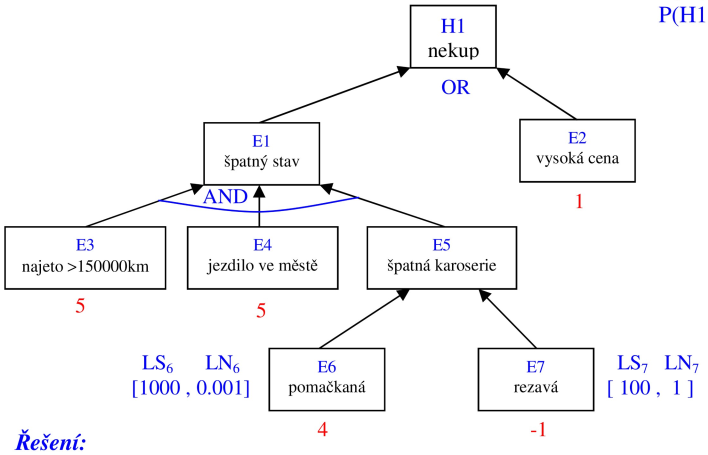
$$C(H\mid e_{celk})=5\frac{P(H\mid e_{celk})-P(H)}{P(H)}$$



P(E)=P(H) $P(H\mid e_{celk})=\frac{O(H\mid e_{celk})}{1+O(H\mid e_{celk})}=\frac{O(H\mid e_1,e_2,\ldots,e_n)}{1+O(H\mid e_1,e_2,\ldots,e_n)}$

Příklad:

P(Ei) = 0.1, pro ∀i
P(H1) = 0.1



Řešení:

(ZŘ): $H1,\ C(H1 \mid e_{celkové})$

$e_{celkové} = \{e_1, e_2\} = \{e_3, e_4, e_5, e_2\} = \{e_3, e_4, e_6, e_7, e_2\}$

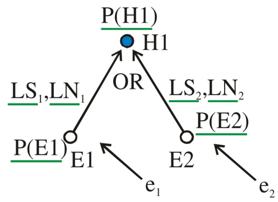
Co jsou to e_i ? , $i = 1,2,\ldots,7$

$C(E_i \mid e_i)$ …míra určitosti tvrzení E_i , za předpokladu, že platí indicie e_i

Např:

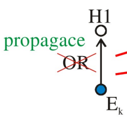
$C(E2 \mid e_2) = 1$, protože např. $\underbrace{\text{auto stojí 105 000,- Kč}}_{e_2}$ a je to Peugeot 205 $\Rightarrow E2 = \uparrow$ cena

1. E1 OR E2 $\rightarrow$ H1, $LS_1 = \infty$, $\cancel{LN_1 = 0}$
$LS_2 = \infty$, $\cancel{LN_2 = 0}$



(1) $C(H1 \mid e_1, e_2) = ?$ $\Rightarrow$ dle 8.1.5 $P(H1 \mid e_1, e_2) = ?$ $\uparrow P(H1)$

(2) $P(H1 \mid e_1, e_2) = f\big(P(E1 \mid e_1), P(E2 \mid e_2), \ldots\big)$



dle 8.1.2 $= f\Big(P(E_k \mid e_k), P(H1), P(E_k), P(H_1 \mid E_k), P(H_1 \mid \overline{E_k})\Big)$

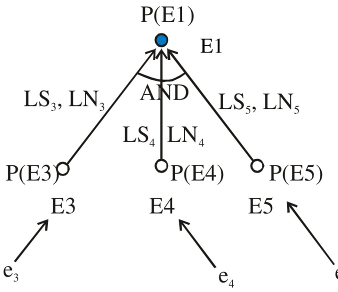
dle 8.1.3 $P(E_k \mid e_k) = \max\big(P(E1 \mid e_1), P(E2 \mid e_2)\big)$
(3)
$k \in \{1,2\}$

(4) $$P(H_1 \mid E_k) = \frac{O(H_1 \mid E_k)}{1 + O(H_1 \mid E_k)} = \frac{LS_k \cdot O(H_1)}{1 + LS_k \cdot O(H_1)}$$

$$= \frac{LS_k \cdot \frac{P(H_1)}{1-P(H_1)}}{1+LS_k \cdot \frac{P(H_1)}{1-P(H_1)}} \quad k \in \{1,2\}$$

$$(5)\quad P(H_1 \mid \overline{E_k}) = \frac{LN_k \cdot \frac{P(H_1)}{1-P(H_1)}}{1+LN_k \cdot \frac{P(H_1)}{1-P(H_1)}} \quad k \in \{1,2\}$$

$P(E1 \mid e_1) = ?$ $P(E2 \mid e_2) = ?$

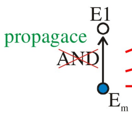


2. E3 AND E4 AND E5 $\rightarrow$ E1, $LS_3 = \infty, \cancel{LN_3 = 0}$

$LS_4 = \infty, \cancel{LN_4} = 0$

$LS_5 = \infty, \cancel{LN_5 = 0}$

$$(6)\quad P(E1 \mid e_1) = f\big(P(E3 \mid e_3), P(E4 \mid e_4), P(E5 \mid e_5)\ldots\big)$$



dle 8.1.2 $= f\big(P(E_m \mid e_m), P(E1), P(E_m), P(E_1 \mid E_m), P(E_1 \mid \overline{E_m})\big)$

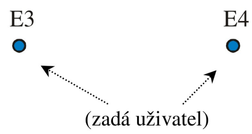
$m \in \{3,4,5\}$

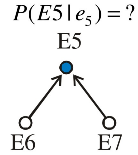
dle 8.1.3 $(7)\quad P(E_m \mid e_m) = \min\big(P(E3 \mid e_3), P(E4 \mid e_4), P(E5 \mid e_5)\big)$

$$(8)\quad P(E1 \mid E_m) = \frac{LS_m \cdot \frac{P(E1)}{1-P(E1)}}{1+LS_m \frac{P(E1)}{1-P(E1)}}$$

$$(9)\quad P(E1 \mid \overline{E_m}) = \frac{LN_m \cdot \frac{P(E1)}{1-P(E1)}}{1+LN_m \frac{P(E1)}{1-P(E1)}}$$

$P(E3 \mid e_3) = ?$ $P(E4 \mid e_4) = ?$ $P(E5 \mid e_5) = ?$





3. E6 → E5, $LS_6 = 1000$, $LN_6 = 0{,}001$

E7 → E5, $LS_7 = 100$, $LN_7 = 1$

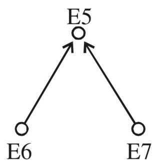
(10) $P(E5 \mid \underbrace{e_5}_{e_6,e_7}) = f\left(P(E5 \mid e_6), P(E5 \mid e_7), \dots\right)$

Rov. 8.1.4.
(komb. pravidel)
používá O, ne P

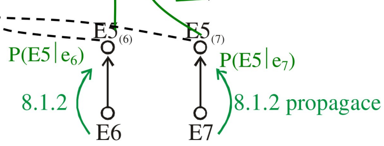
$$= \frac{O(E5 \mid e_6, e_7)}{1 + O(E5 \mid e_6, e_7)}, \text{ kde}$$

dle 8.1.4 (11) $O(E5 \mid e_6, e_7) = \dfrac{O(E5 \mid e_6)}{O(E5)} \cdot \dfrac{O(E5 \mid e_7)}{O(E5)} \cdot O(E5)$, kde

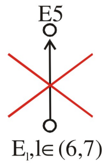
$$O(E5 \mid e_6) = \frac{P(E5 \mid e_6)}{1 - P(E5 \mid e_6)}, \quad O(E5) = \frac{P(E5)}{1 - P(E5)}, \quad O(E5 \mid e_7) = \frac{P(E5 \mid e_7)}{1 - P(E5 \mid e_7)}$$



kombinace!

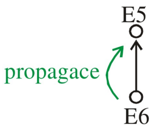


Nelze u kombinace
(jde jen u OR, AND)



(12) $P(E5 \mid e_6) = f\left(P(E6 \mid e_6), P(E5), P(E6), P(E5 \mid E6), P(E5 \mid \overline{E6})\right)$

Dle 8.1.2



(13) $$P(E5 \mid E6) = \frac{\dfrac{LS_6 \cdot P(E5)}{1 - P(E5)}}{1 + \dfrac{LS_6 \cdot P(E5)}{1 - P(E5)}}$$

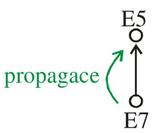
(14) $$P(E5 \mid \overline{E6}) = \frac{\dfrac{LN_6 \cdot P(E5)}{1 - P(E5)}}{1 + \dfrac{LN_6 \cdot P(E5)}{1 - P(E5)}}$$

$P(E6 \mid e_6) = ?$

E6

(15) $P(E5 \mid e_7) = f\left(P(E7 \mid e_7), P(E5), P(E7), P(E5 \mid E7), P(E5 \mid \overline{E7})\right)$

Dle 8.1.2



(16) $$P(E5 \mid E7) = \frac{\dfrac{LS_7 \cdot P(E5)}{1 - P(E5)}}{1 + \dfrac{LS_7 \cdot P(E5)}{1 - P(E5)}}$$

(17) $$P(E5 \mid \overline{E7}) = \frac{\dfrac{LN_7 \cdot P(E5)}{1 - P(E5)}}{1 + \dfrac{LN_7 \cdot P(E5)}{1 - P(E5)}}$$

$P(E7 \mid e_7) = ?$

E7

Zbývá určit:

$$\begin{array}{ccccc}
P(E2 \mid e_2), & P(E3 \mid e_3), & P(E4 \mid e_4), & P(E6 \mid e_6), & P(E7 \mid e_7) \\
\uparrow \text{Dle 8.1.6} & \uparrow & \uparrow & \uparrow & \uparrow \\
C(E2 \mid e_2), & C(E3 \mid e_3), & C(E4 \mid e_4), & C(E6 \mid e_6), & C(E7 \mid e_7) \\
P(E2) & P(E3) & P(E4) & P(E6) & P(E7)
\end{array}$$

Numerický výpočet:

(Od listů ke kořenu)

$$C(Ei|e_i)>0 \left\{ \begin{array}{lccccc}
P(E2 \mid e_2) = & C(E2 \mid e_2) & \cdot \frac{1}{5}\big(1 - P(E2)\big) & + P(E2) & = & 0{,}28 \\
 & 1 & 0{,}9 & 0{,}1 & & \\
P(E3 \mid e_3) = & 5 & 0{,}9 & 0{,}1 & = & 1 \\
P(E4 \mid e_4) = & 5 & 0{,}9 & 0{,}1 & = & 1 \\
P(E6 \mid e_6) = & 4 & 0{,}9 & 0{,}1 & = & 0{,}82
\end{array} \right.$$

$$C(E7|e_7)<0 \left\{ \begin{array}{lccccc}
P(E7 \mid e_7) = & C(E7 \mid e_7) & \cdot \frac{1}{5}\big(P(E7)\big) & + P(E7) & = & 0{,}08 \\
 & -1 & 0{,}1 & 0{,}1 & &
\end{array} \right.$$

(18) $P(E5 \mid E_7)$ nepočítám, neboť $\underset{0{,}08}{P(E7 \mid e_7)} < \underset{0{,}1}{P(E_7)} \Rightarrow$ nepotřebuji LS_7

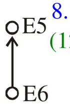
(17) $$P(E5 \mid \overline{E7}) = \frac{1 \cdot \frac{0{,}1}{0{,}9}}{1 + 1 \cdot \frac{0{,}1}{0{,}9}} = 0{,}1$$

8.1.2:

(15) $$P(E5 \mid e_7) = P(E5 \mid \overline{E7}) + \frac{P(E7 \mid e_7)}{P(E7)} \cdot \underbrace{\big(P(E5) - P(E5 \mid \overline{E7})\big)}_{0} = 0{,}1$$

(13) $$P(E5 \mid E6) = \frac{1000 \cdot \frac{0{,}1}{0{,}9}}{1 + 1000 \cdot \frac{0{,}1}{0{,}9}} = \frac{1000}{1009} = 0{,}9911$$

(14) $P(E5 \mid \overline{E6})$ nepočítám, neboť $P(E6 \mid e_6) > P(E6) \Rightarrow$ nepotřebuji LN_6



8.1.2:

(12) $$P(E5 \mid e_6) = P(E5) + \frac{P(E6 \mid e_6) - P(E6)}{1 - P(E6)} \cdot \big(P(E5 \mid E6) - P(E5)\big) = 0{,}1 + \frac{0{,}82 - 0{,}1}{1 - 0{,}1} \cdot (0{,}9911 - 0{,}1)$$

$$= 0{,}8129$$

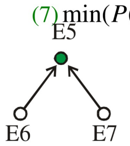
(11) $$O(E5 \mid e_6, e_7) = \frac{0{,}8129}{1 - 0{,}8129} \cdot \frac{0{,}1}{1 - 0{,}1} \cdot \frac{1 - 0{,}1}{0{,}1} \doteq 4{,}3447$$

(10) $$P(E5 \mid e_5) \doteq \frac{4{,}3447}{1 + 4{,}3447} \doteq 0{,}8129$$

(7) $\min(P(E3 \mid e_3), P(E4 \mid e_4), P(E5 \mid e_5)) = 0{,}8129 = P(Em \mid e_m)$

$\downarrow$

m=5

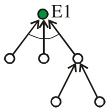


$$P(E1\,|\,E5)=\frac{LS\cdot\frac{0,1}{0,9}}{1+LS\cdot\frac{0,1}{0,9}}="\frac{\infty}{\infty}"=1 \qquad \text{(zde totiž LS} \to \infty)$$

(8) $P(E1\,|\,\overline{E5})=\dfrac{0}{1+0}=0$, ale nemusíme počítat, $P(E5\,|\,e_5)>P(E5)$

(6) $P(E1\,|\,e_1)=P(E5\,|\,e_5)=0{,}8129$

U deterministických pravidel se stejnou apriorní pravděpodobností se „přenese“ aposteriorní pravděpodobnost předpokladu do závěru.



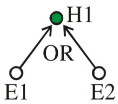
(3) $\max\big(P(E1\,|\,e_1),P(E2\,|\,e_2)\big)=P(E1\,|\,e_1)=0{,}8129$
$k=1$

(5) $P(H1\,|\,\overline{E1})=0$

(4) $P(H1\,|\,E1)=1$

(2) $P(H1\,|\,e_1,e_2)=P(E_k\,|\,e_k)=P(E1\,|\,e_1)=0{,}8129$

U deterministických pravidel se stejnou apriorní pravděpodobností se pravděpodobnost předpokladu „přenese“ do závěru.



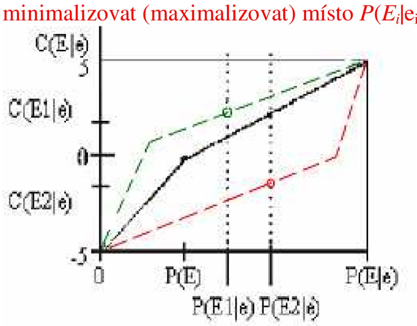
(1) $C(H1\,|\,e1,e2)=5\cdot\dfrac{P(H1\,|\,e_1)-P(H1)}{1-P(H1)}=5\cdot\dfrac{0{,}8127-0{,}1}{0{,}9}=\underline{3{,}96}$

$P(H1\,|\,e_1)>P(H1)$

Odpověď: Nekup (auto) s mírou určitosti 3,96 ≈ 4.

Ušetřit počítání lze nejen u deterministických pravidel se stejnou apriorní pravděpodobností předpokladu a závěru, ale dále též i:

Jsou-li apriorní pravděpodobnosti dílčích předpokladů pravidla s logickými spojkami stejné, lze minimalizovat (maximalizovat) místo $P(E_i|e_i)$ $C(Ei|e_i)$ a naopak.



příklad různých $P(E1)$ $P(E2)$, kdy tvrzení neplatí
$C(E1\,|\,e)>C(E2\,|\,e)\wedge P(E1\,|\,e)<P(E2\,|\,e)$

8.1.4 Zhodnocení Bayesova přístupu

Nevýhody:

1. Musí se znát předem apriorní pravděpodobnosti $P(E_i)$ (musí zadat expert)
2. Musí platit $P(H\,|\,E)+P(\overline{H}\,|\,E)=1$
3. Předpokládá se nezávislost předpokladů v pravidle tvaru:
   IF E1
   AND E2
   AND E3
   THEN H
   Proto je třeba snažit se o co nejmenší počet AND v pravidle.
4. Když uživatel zadá P(E|e) ⇒ musí se přepočítat síť O(E|e) ⇒ výpočetní náročnost

8.2 TEORIE URČITOSTI

(MYCIN, EMYCIN)

MB … míra důvěry – relativní přírůstek pravděpodobnosti H uplatněním pravidla
MD … míra nedůvěry – relativní úbytek pravděpodobnosti H
CF … činitel jistoty, faktor jistoty ∈ ⟨-1; +1⟩ = CF

8.2.1

$$\mathrm{MB(H,E)} = \begin{cases} = 1 & \mathrm{P(H)} = 1 \\ = \dfrac{\mathrm{P(H \mid E)} - \mathrm{P(H)}}{1 - \mathrm{P(H)}} & \mathrm{P(H \mid E)} > \mathrm{P(H)} \\ = 0 & \mathrm{P(H \mid E)} \leq \mathrm{P(H)} < 1 \end{cases}$$

$$\mathrm{MD(H,E)} = \begin{cases} = 1 & P(H) = 0 \\ = \dfrac{\mathrm{P(H)} - \mathrm{P(H \mid E)}}{\mathrm{P(H)} - 0} & P(H \mid E) < P(H) \\ = 0 & P(H \mid E) \geq P(H) > 0 \end{cases}$$

$$\mathrm{CF(H,E) = MB(H,E) - MD(H,E)}$$

↑ MYCIN

$$CF(H,E) = \frac{MB(H,E) - MD(H,E)}{1 - \min(MB, MD)}$$

↑ EMYCIN

Příklady:

A. E zcela podporuje H (E je postačitelné pro H): P(H|E) = 1
   $\mathrm{P(H \mid E) = 1} \Rightarrow$ MB(H,E) = 1, MD(H,E) = 0, CF(H,E) = 1

B. E zcela vyvrací H (E je postačitelné pro $\overline{\mathrm{H}}$); $\mathrm{P(\overline{H} \mid E) = 1}$
   $\mathrm{P(\overline{H} \mid E) = 1}$, tj. P(H|E) = 0 $\Rightarrow$ MB(H,E) = 0, MD(H,E) = 1, CF(H,E) = -1

C. E nijak neovlivňuje H; P(H) ∈ (0,1) : P(H|E) = P(H)
   P(H|E) = P(H) $\Rightarrow$ MB(H,E) = 0, MD(H,E) = 0, CF(H,E) = 0

D. E podporuje H; P(H) < P(H|E) < 1
   P(H) < P(H|E) < 1 $\Rightarrow$ $\mathrm{MB} = \dfrac{\mathrm{P(H \mid E)} - \mathrm{P(H)}}{1 - \mathrm{P(H)}}$; MD(H,E) = 0, CF(H,E) = MB(H,E)

E. E vyvrací H; 0 < P(H|E) < P(H)
   0 < P(H|E) < P(H) $\Rightarrow$ MB(H,E) = 0; $\mathrm{MD} = \dfrac{\mathrm{P(H)} - \mathrm{P(H \mid E)}}{\mathrm{P(H)}}$, CF(H,E) = - MD(H,E)

F. Více E_i podporuje a jedna E_j vyvrací H

$E_1, E_2, \ldots, E_n$, $n\uparrow$ $\Rightarrow MB(H, E_1, E_2, \ldots, E_n) \rightarrow 0{,}9999$

$\Rightarrow MD(H, E_{n+1}) = 0{,}8$ (např.)

$\Rightarrow$ dle MYCIN CF = MB – MD = 0,1999 přestože podporujících E_i >> vyvracejících E_i

$\Rightarrow$ dle EMYCIN CF = 0,995

Pravděpodobnost pro a proti

V pravděpodobnosti P platí $P(H \mid E) + P(\overline{H} \mid E) = 1$. Jak je tomu u CF?

$CF(\overline{H}, E) = MB(\overline{H}, E) - MD(\overline{H}, E)$ MYCIN

$CF(\overline{H}, E) = 0 - \dfrac{P(\overline{H}) - P(\overline{H} \mid E)}{P(\overline{H})}$, je-li $P(\overline{H} \mid E) < P(\overline{H})$ $\Leftrightarrow$ 1-P(H|E) < 1-P(H)

$\Leftrightarrow$ P(H|E) > P(H)

$CF(H, E) = MB(H, E) - MD(H, E)$

$CF(H, E) = \dfrac{P(H \mid E) - P(H)}{1 - P(H)}$, je-li P(H|E) > P(H)

$$CF(\overline{H}, E) + CF(H, E) = \frac{-1 + P(H) + 1 - P(H \mid E) + P(H \mid E) - P(H)}{1 - P(H)} = 0$$

$\Rightarrow CF(\overline{H}, E) = -CF(H, E)$

Je-li $P(\overline{H} \mid E) \geq P(\overline{H}) \Rightarrow$ analogicky

Tedy $\underline{CF(\overline{H}, E) = -CF(H, E)}$ (vždy)

Podmíněný činitel jistoty CF(H,E):

Příklad:

$P(H_1) = 0{,}7$ > $P(H_2) = 0{,}2$

$P(H_1|E) = 0{,}9$ > $P(H_2|E) = 0{,}8$

$$CF(H_1, E) = \frac{0{,}9 - 0{,}7}{1 - 0{,}7} = \frac{0{,}2}{0{,}3} = \frac{2}{3} = \qquad CF(H_2, E) = \frac{0{,}8 - 0{,}2}{1 - 0{,}2} = \frac{0{,}6}{0{,}8} = \frac{3}{4}$$

= 0,666 < = 0,750

(není to jako u pravděpodobnosti)

8.2.2 Propagace

$e \rightarrow E \rightarrow H$ CF(H,e) = CF(H,E).CF(E,e)

8.2.3 Logické spojky

CF(H,E1 AND E2) = min(CF(H,E1),CF(H,E2))

OR max

$CF(H, \overline{E}) = -CF(H, E)$ / heuristika – neplatí přesně z def. CF

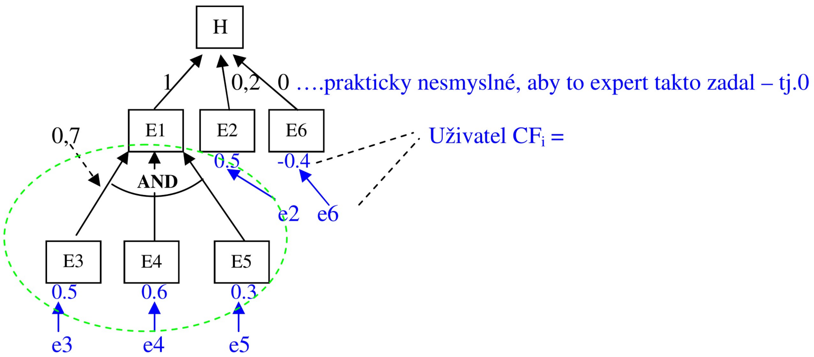
8.2.4 Kombinace pravidel

$$CF_{COMBINE}(CF_1, CF_2) = \begin{cases} CF_1 + CF_2 - CF_1.CF_2 & CF_1 > 0, CF_2 > 0 \\ \dfrac{CF_1 + CF_2}{1 - \min(|CF_1|, |CF_2|)} & CF_1.CF_2 < 0 \\ CF_1 + CF_2 + CF_1.CF_2 & CF_1 < 0, CF_2 < 0 \end{cases}$$

Platí komutativnost, tj. ($CF_{COMBINE}(CF_1,CF_2) = CF_{COMBINE}(CF_2,CF_1)$)

Vztahy 8.2.2, 8.2.3, 8.2.4 platí pro CF definovaný pro EMYCIN.

Příklad:



Řešení:

uživatel: $CF(E_3,e_3) = 0,5$ $\quad$ expert: $CF(E_1, E_3, E_4, E_5) = 0,7$

$CF(E_4,e_4) = 0,6$ $\quad$ $CF(H,E_1) = 1$

$CF(E_5,e_5) = 0,3$ $\quad$ $CF(H,E_2) = 0,2$

$CF(E_2,e_2) = 0,5$ $\quad$ $CF(H,E_6) = 0$

$CF(E_6,e_6) = -0,4$

$CF(H,e_3, e_4, e_5, e_2, e_6) = ?$

$\rightarrow CF(H,e_3, e_4, e_5, e_2, e_6) = CF_{COMBNIE}(CF(H, e_3, e_4, e_5), CF(H,e_2), CF(H,e_6))$

$\Rightarrow CF(H,e_3, e_4, e_5) = CF(H,E_1)\ CF(E_1, e_3, e_4, e_5)$

$\uparrow$ propagace

$\downarrow$ propagace 8.2.2

$CF(E_1, e_3, e_4, e_5) = CF(E_1, E_3, E_4, E_5)\ CF(E_3, E_4, E_5, e3, e_4, e_5)$

logicky v.p. 8.2.3 $\rightarrow$ $= CF(E_1, E_3, E_4, E_5) \min\{CF(E_3,e_3), CF(E_4,e_4), CF(E_5,e_5)\}$

$= 0,7 * 0,3 = 0,21$

$CF(H,e_3, e_4, e_5) = 1*0,21 = 0,21$

$\Rightarrow CF(H,e_2) = CF(H,E_2)* CF(E_2,e_2) = 0,2*0,5 = 0,1$

$\uparrow\downarrow$ propagace

$\Rightarrow CF(H,e_6) = CF(H,E_6)* CF(E_6,e_2) = 0*(-0,4) = 0$ (-0,4 nemá smysl zadávat)

$\rightarrow CF_{COMBINE}(A,B,C) = CF_{COMBINE}\ (CF_{COMBINE}(A,B),C)$:

$CF_{COMBINE}(A,B)= CF_{COMBINE}(0.21,0.1)=0,289$; $\quad CF_{COMBINE}(0.289,0)=0,289$

$CF(H,e_3, e_4, e_5, e_2, e_6) = 0,289$

Poznámky k teorii určitosti

1. Více e_i podporujících H; např. bude pršet
   $CF_{COMBINE}(CF_1,CF_2) = 0,8 + 0,8 - 0,8*0,8 = 0,96$ $\quad CF_1 = 0,8;\ CF_2 = 0,8$
   $CF_{COMBINE}(CF_1,CF_2,\ldots,CF_N) \rightarrow 1\ \ (N\rightarrow\infty)$
   1 je proti: $CF_{N+1} = -0,8$, pak $CF_{COMBINE}(CF_1,CF_2,\ldots,CF_N) = (1-0,8)/(1-\underline{0,8}) \rightarrow 1$
   (proto je tu min)
2. Heuristické hledání
   Například: R1: IF A AND B THEN C1 CF = 0,8 $\quad$ R2: IF D THEN C2 CF = 0,7
   Prioritu má pravidlo s vyšším $|CF|$, tj. R1
   Kdyby CF(C1)< CF(C2), pak můžeme brát R2

3. Řízené hledání pomocí CF
   Např. IF CF (Problém je v elektrickém systému) < 0,5
   THEN GOAL = Problém je v palivovém systému
4. Prořezávání při hledání
   Např.: pravidla s |CF(H|e)| < 0,2 nejsou dále uvažována
   ↑ obecně threshold
   MYCIN - uvažuje 0,2
5. V případě hlubokého řetězení CF(H,e) klesá s hloubkou ⇒ problém při prořezávání (MYCIN má ale malou hloubku)
6. Problém, kdy jednu hypotézu vyvozuje mnoho předpokladů.
   Např. všichni věří, že bude pršet s 0,8 CF, ale systém uvažuje hodnotu $CF_{COMBINE} = 1$
   ⇒ vyhněme se použití mnoha pravidel vyvozujících tu samou hypotézu
7. Problém pravidel s konjugovanými předpoklady
   Příklad: IF Obloha je zatažená
   AND Vítr sílí
   AND Teplota klesá
   THEN Bude pršet
   CF 0,9

   CF (Obloha je zatažená) = 1
   CF(Vítr sílí) = 1 ?
   CF(Teplota klesá) = 0,1
   CF(Bude pršet) = ? = 0,09

   Závěr: Houby vím o tom, jestli bude pršet či nikoliv.

Lépe:

R1: IF Obloha je zatažená
THEN Bude pršet
CF 0,7

R2: IF Vítr sílí
THEN Bude pršet
CF 0,6

R3: IF Teplota klesá
THEN Bude pršet
CF 0,6

CF(Bude pršet) = ?

$CF_1=1*0,7=0,7$
$CF_2=1*0,6=0,6$
$CF_3=0,1*0,6=0,06$

$CF_{COMBINE}(CF_1 ,CF_2)= 0,7 + 0,6 – 0,7 * 0,6 = 0,88 = CF_{12}$
$CF_{COMBINE}(CF_{12} ,CF_3)= 0,88 + 0,06 – 0,88 * 0,06 = 0,89 = CF$
↑to je lepší, ale problém 6. zůstává

8.3 FUZZY LOGIKA

V klasické teorii množin se zavádí tzv. charakteristická funkce:

$$\mu_A(x) = \begin{cases} 1 & \text{jestliže} \quad x \in A \\ 0 & \text{jestliže} \quad x \notin A \end{cases}$$

Jedná se o zobrazení univerza X do množiny {0,1}: $\mu_A(x)$: X → {0,1}.

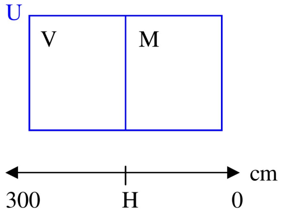
Prvek tedy do množiny patří nebo nepatří. V teorii fuzzy množin může oproti tomu prvek do množiny patřit částečně. Stupeň příslušnosti prvku do fuzzy množiny je vyjadřován pomocí zobecněné charakteristické funkce nazývané **funkce příslušnosti**.

$\mu_A(x): X \to \langle 0,1 \rangle$ (do intervalu!)

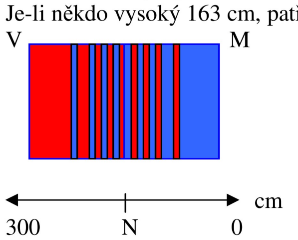
Stupeň příslušnosti prvku v množině se tedy pohybuje v rozmezí $0 \leq \mu_A \leq 1$, kde 0 znamená úplnou nepříslušnost a 1 úplnou příslušnost prvku do fuzzy množiny A.

Např.:

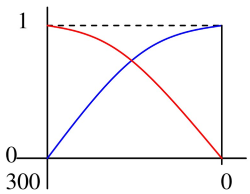
V klasické teorii množin mějme množinu vysokých lidí V a její doplněk malých lidí M = $\overline{V}$ =U-V



H… hodnota výšky v cm, dělící populaci na množiny V a M, např. 165 cm.
Je-li někdo vysoký 163 cm, patří do množiny malých; je-li vysoký 166cm, patří do množiny vysokých.



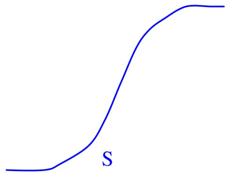
Fuzzy množiny V, M



- funkce příslušnosti (subjektivní hodnocení) výšky do fuzzy množiny vysoký $\mu_V(x)$
- funkce příslušnosti výšky do fuzzy množiny malý $\mu_M(x)$

Poznámka: V teorii se často používá tzv. S-funkce (Π fce)

$$S(x, \alpha, \beta, \gamma) = \begin{cases} 0 & \dots\ x \le \alpha \\ \frac{1}{2}\left(\frac{x-\alpha}{\beta-\alpha}\right)^2 & \dots\ \alpha \le x \le \beta \\ 1-\frac{1}{2}\left(\frac{x-\gamma}{\gamma-\beta}\right)^2 & \dots\ \beta \le x \le \gamma \\ 1 & \dots\ x \ge \gamma \end{cases}$$



Π analogicky

Fuzzy logika := odvětví logiky, které užívá stupně příslušnosti do množiny <0,1> místo striktní ANO/NE příslušnosti

Lingvistická proměnná := pojem užívaný v našem přirozeném jazyce k popisu věcí (objektů a jejich vlastností), které mají vágní (rozptýlené, neurčité) hodnoty

Fuzzy množina := množina charakterizovaná funkcí příslušnosti $\mu_A(x)$, která přiřazuje každému prvku X (z univerza) hodnotu z intervalu <0,1>.
(Stupeň příslušnosti)

Reprezentace fuzzy množin

= Jak vyjádřit funkci příslušnosti $\mu_A(x)$: pomocí S-funkce nebo v případě diskrétního souboru prvků pomocí konečné množiny prvků, kde každému prvku přiřadíme hodnotu funkce příslušnosti fuzzy množiny $A = (a_1, a_2, \dots, a_n)$, kde $a_i = \mu_A(x_i)$.
Píšeme pro přehlednost ve tvaru: $A = (a_1/x_1, a_2/x_2, \dots, a_n/x_n)$!takto
např.: vysoký = (0/160; 0,2/175; 0,5/185; 0,9/190; 1/200)

Prvky fuzzy množiny, pro které platí $\mu_A(x) > 0$, tvoří tzv. podporu (support) fuzzy množiny.
α-řez = množina, jejíž prvky mají $\mu > \alpha$ (α je zvolená konstanta).
Konečná fuzzy podmnožina o N prvcích je obvykle reprezentována jako sjednocení fuzzy množin o jednom prvku μ_i/x_i:

$$A = \sum_{i=1}^{N} \mu_i/x_i = \bigcup_{i=1}^{N} \mu_i/x_i$$

Pro spojitou funkci: $A = \int_x \mu_A(x_i)/x_i$

Operátory:

- koncentrace (hodně) = CON(A): $\mu_{CON(A)}(X) = [\mu_A(x)]^2$
- roztažení (trochu) = DIL(A): $\mu_{DIL(A)}(X) = \sqrt{[\mu_A(x)]}$
- intenzifikace (vskutku) = INT(A): $\mu_{INT(A)}(X) = \begin{cases} 2[\mu_A(x)]^2 & 0 \le \mu_A(x) \le \frac{1}{2} \\ 1-2[1-\mu_A(x)]^2 & \frac{1}{2} \le \mu_A(x) \le 1 \end{cases}$
- mohutnost (hodně hodně) = POW(A): $\mu_{POW(A)}(X) = [\mu_A(x)]^n$
- normalizace = NORM(A): $\mu_{NORM(A)}(X) = \mu_A(x)/\max\{\mu_A(y)/y \in X\}$

- průnik: $A \cap B$: pro $\forall x \in X$

$$\mu_{A \cap B} = \min\{\mu_A(x), \mu_B(x)\}$$
$$= \mu_A(x) \wedge \mu_B(x)$$
$$= \mu_A(x) \cap \mu_B(x)$$

  např.: vysoký = (0/160; 0,2/175; 0,5/185; 0,9/190; 1/200)
  nízký = (1/160; 0,5/175; 0,2/185; 0,1/190; 0/200)
  $\mu_{\text{vysoký} \wedge \text{nízký}}(x)$ = (0/160; 0,2/175; 0,2/185; 0,1/190; 0/200)

- sjednocení: $A \cup B$: pro $\forall x \in X$

$$\mu_{A \cup B} = \max\{\mu_A(x), \mu_B(x)\}$$
$$= \mu_A(x) \vee \mu_B(x)$$
$$= \mu_A(x) \cup \mu_B(x)$$

- doplněk A‘: $\mu_{A'}(x) = 1 - \mu_A(x)$
- rovnost: $A = B$: $\mu_A(x) = \mu_B(x)$ pro $\forall x \in X$
- inkluze: $A \subseteq B$: $\mu_A(x) \leq \mu_B(x)$ pro $\forall x \in X$
  $A \subset B$: $\mu_A(x) \leq \mu_B(x)$ a zároveň alespoň pro jedno x: $\mu_A(x) < \mu_B(x)$

Pro fuzzy množiny <u>platí</u> komutativnost, asociativnost; <u>neplatí</u> zákon protikladu a vyloučení třetího – tedy může platit: $A \cup A' \neq U$
$A \wedge A' \neq \emptyset$

Odvození
Př1: A: vysoký
B: ne hodně vysoký: $\mu_B(x) = 1 - (\mu_A(x))^2$
Př2: A: vysoký
B: nízký
C: ne velmi vysoký a ne velmi nízký: $\mu_C(x) = [1 - (\mu_A(x))^2] \wedge [1 - (\mu_B(x))^2]$

Fuzzy inference

Fuzzy výrok: X je A, kde A je fuzzy množina
(A: vysoká; X: rychlost)
Y je B, kde B je fuzzy množina
(B: nízký; Y: tlak)

fuzzy pravidlo: IF X je A THEN Y je B
Expertní systém uchovává pravidla:
IF A THEN B jako asociaci (A,B) v matici M
M zobrazuje fuzzy množinu A do fuzzy množiny B

Např.: IF výška je vysoká THEN váha je těžká
A B
X výška A – fuzzy množina vysoká výška
Y váha B – fuzzy množina těžká váha
$A = (a_1/x_1, a_2/x_2, \ldots, a_n/x_n)$
$B = (b_1/y_1, b_2/y_2, \ldots, b_m/y_m)$
M =: $A \circ M = B$

Úkolem je určit M z A a B. Budeme-li M znát, budeme umět určit B^+ z $A^+ \cap$ M.

Obecně: M: $\begin{bmatrix} a_1 \rightarrow b_1 & a_1 \rightarrow b_2 & \ldots\ldots\ldots \\ a_2 \rightarrow b_1 & \ldots\ldots\ldots\ldots & \\ \ldots\ldots\ldots & \ldots\ldots\ldots\ldots & \ldots\ldots\ldots \end{bmatrix}$

Vzniklo několik operátorů → 2 nejpoužívanější jsou Max-min a Max-product inference.

Max-min inference:

- M : $m_{ij} := \min(a_i, b_j)$

Max-product inference:

- M : $m_{ij} := a_i \bullet b_j$

pak $B^+ : b_j^+ = \max_{(1 \le i \le n)} \{\underbrace{\min(a_i^+, m_{ij})}_{\text{operátor } A^+ \circ M}\}$

Příklad: IF teplota je normální THEN rychlost je střední (A… teplota normální, B… rychlost střední)
A: (0/100; 0,5/125; 1/150; 0,5/175; 0/200)
B: (0/10; 0,6/20; 1/30; 0,6/40; 0/50)
IF A THEN B

$$M = m_{ij} = \min(a_i, b_j) \quad = \begin{bmatrix} \min(0;0) & \min(0;0,6) & \min(0;1) & \min(0;0,6) & \min(0;0) \\ \min(0,5;0) & \min(0,5;0,6) & \min(0,5;1) & \min(0,5;0,6) & \min(0,5;0) \\ \min(1;0) & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \end{bmatrix}$$

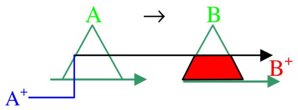
$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0,5 & 0,5 & 0,5 & 0 \\ 0 & 0,6 & 1 & 0,6 & 0 \\ 0 & 0,5 & 0,5 & 0,5 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Máme-li nyní A^+ = (0/100; 0,5/125; 0/150; 0/175; 0/200)
Pak $B^+ = b_j^+ = \max\{\min(a_i^+, m_{ij})\}$ $\quad 1 \le i \le n$
$b_1^+ = \max[\min(0;0), \min(0,5;0), \min(0;0), \min(0,0), \min(0;0)]$
$b_2^+ = \max[\min(0;0), \min(0,5;0,5), \min(0;0,6), \min(0;0,5), \min(0;0)]$
…
B^+ = (0/10, 0,5/20, 0,5/30, 0,5/40; 0/50)

odpověď:
V případě, že A^+ =(0/100; 0,5/125; 0/150; 0/175; 0/200) ,
je B^+ = (0/10, 0,5/20, 0,5/30, 0,5/40; 0/50)
V reálných situacích hodnota A^+ je tvořena <u>jedním</u> číslem (např. změření t = 135°C = x_{SINGLE})
Pro tento případ platí: $\boxed{B^+ = \mu_{A^+}(x_{SINGLE}) \wedge \mu_B(y)}$ $\quad \forall$y: $\mu_{B^+}(y) = min(\mu_{A^+}(x_{SINGLE}), \mu_B(y))$

Příklad: $\mu_{A^+} = 0{,}5$

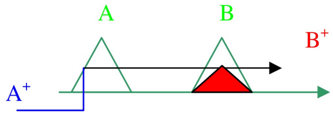
$A^+ = [\min(0{,}5;0), \min(0{,}5;0{,}6), \min(0{,}5;1), \min(0{,}5;0{,}6), \min(0{,}5;0)] = (0; 0{,}5; 0{,}5; 0{,}5; 0)$



M = m_{ij} = …

Max product
single value

$\boxed{B^+ = \mu_{A^+}(x_{SINGLE}) \bullet \mu_B(y)}$ $\quad \forall y: \ \mu_{B^+}(y) = (\mu_{A^+}(x_{SINGLE}) \bullet \mu_B(y))$



Několik předpokladů v pravidle:

IF A AND B THEN C

M = ?

Řešení: $A \circ M_{AC} = C \quad B \circ M_{BC} = C$ (místo M uvažujeme M_{AC} a M_{BC})

$A^+ \circ M_{AC} = C_{A^+}$

$B^+ \circ M_{BC} = C_{B^+}$

Pak při zadaném A^+, B^+ odvodíme C^+:

$A^+ \wedge B^+ \to C^+$

$\boxed{C^+ = [A^+ \circ M_{AC}] \wedge [B^+ \circ M_{BC}] = C_{A^+} \wedge C_{B^+}}$ $\quad A^+ \wedge B^+ \to C^+$

IF A OR B THEN C

Řešení:

- při zadaném A^+, B^+ získáme C^+:

$A^+ \vee B^+ \to C^+$

$\boxed{C^+ = [A^+ \circ M_{AC}] \vee [B^+ \circ M_{BC}] = C_{A^+} \vee C_{B^+}}$ $\quad A^+ \vee B^+ \to C^+$

Zjednodušení pro SINGLE VALUE A^+, B^+:

$$\left(\mu_{A^+}(x_{SINGLE}) \overset{\wedge\to}{\bullet} \mu_C\right) \overset{\wedge}{\vee} \left(\mu_{B^+}(x_{SINGLE}) \overset{\wedge\to}{\bullet} \mu_C\right) = \substack{\min \\ \max}\left(\mu_{A^+}(x_{SINGLE}), \mu_{B^+}(x_{SINGLE})\right) \overset{\wedge\to}{\bullet} \mu_C \qquad \text{pro} \quad A^+ \overset{\wedge}{\vee} B^+ \ \to \ C^+$$

Max-min

Max-product

$$\substack{\min \\ \max}\left[\substack{\min \\ \bullet}(a,c), \substack{\min \\ \bullet}(b,c)\right] = \substack{\min \\ \bullet}\left[\substack{\min \\ \max}(a,b), c\right]$$

Několik pravidel:
IF A_1 THEN B_1
IF A_2 THEN $B_2 \Rightarrow M_1, M_2, \ldots, M_N$
IF …
…
IF A_N THEN B_N

Pak při zadaných $A_1^+, A_2^+, \ldots, A_N^+$

1. určíme: $B_1^+, B_2^+, \ldots, B_N^+$
2. $\boxed{B^+ = B_1^+ \cup B_2^+ \cup \ldots \cup B_N^+ = \max(B_1^+, B_2^+, \ldots, B_N^+)}$

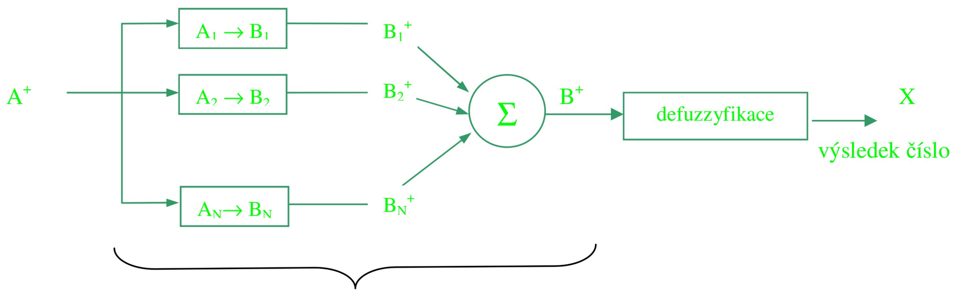
Defuzzyfikace

Výsledkem je fuzzy množina např.:
normální rychlost = (a_1/x_1, a_2/x_2, …, a_n/x_n). My ale chceme vědět, jaká je rychlost, tj. číslo. Procesu určení čísla z vektoru reprezentujícího fuzzy množinu, se říké defuzzyfikace.

Nejvíce populární metodou je metoda fuzzy centroidu:

$$\boxed{x = \frac{\sum_{j=1}^{N} x_j \mu_A(x_j)}{\sum_{j=1}^{N} \mu_A(x_j)}}$$

Architektura systému s fuzzy pravidly:



Takovýchto bloků může být více s delšími větvemi pravidel.

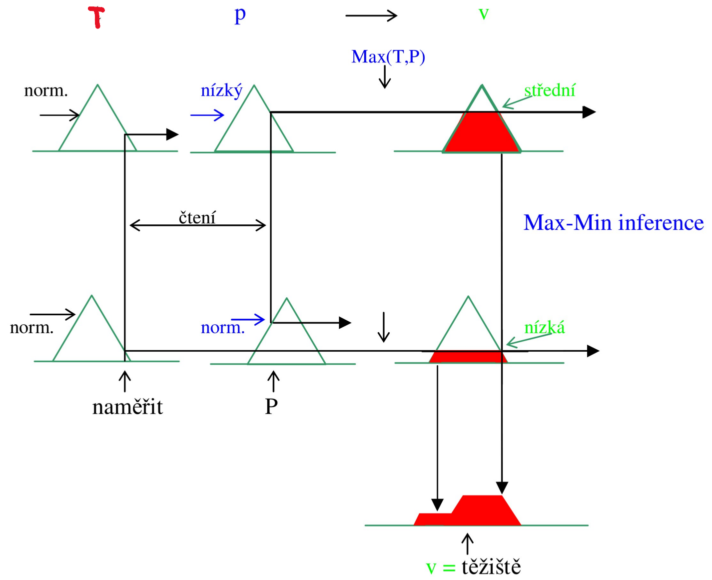
Příklad:

IF teplota je normální
OR tlak je nízký
THEN rychlost je střední

IF teplota je normální
AND tlak je normální
THEN rychlost je nízká

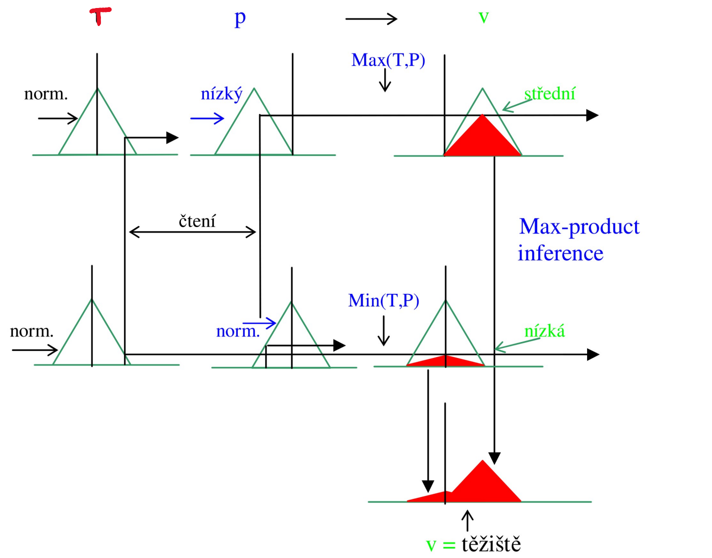
IF T = normální
OR p = ↓
THEN v = střední

IF T = normální
AND p = normální
THEN v = ↓



IF T = normální
OR p = ↓
THEN v = střední

IF T = normální
AND p = normální
THEN v = ↓



8.4 DEMPSTER – SHAFEROVA TEORIE

Dempster – Shaferova teorie je založena na myšlence modelování neurčitosti místo jedním číslem určitým intervalem „pravděpodobnosti“. Celá teorie je vybudována na velmi dobrém teoretickém základě a bylo prokázáno, že faktory jistoty jsou speciálním případem Dempster – Shaferovy teorie.
Uvažujme množinu Θ vzájemně nezávislých prvků. Např.:

Θ = {dopravní letadlo, bombardér, stíhačka}

Θ = {červená, zelená, modrá, oranžová, žlutá}

$\Theta = \{\Theta_1, \Theta_2, \dots, \Theta_N\}$

Odpověď na otázku: „Jaká jsou vojenská letadla?“
potom reprezentuje podmnožina Θ: $\{\Theta_2, \Theta_3\}$ = {bombarder, stíhačka}={B,S}
Obdobně na otázku: „Jaká jsou civilní letadla?“
získáme odpověď: $\{\Theta_1\}$ = {dopravní letadla}={D}

Každá podmnožina množiny Θ tedy představuje možnou odpověď na nějakou otázku z oblasti, o kterou se zajímáme. Za předpokladu, že prvky množiny Θ jsou nezávislé a množina Θ „pokrývá oblast“ našeho zájmu, potom na otázku z příslušné oblasti existuje pouze jedna správná odpověď.

Množina Θ o celkem N prvcích má celkem 2^N podmnožin, které odpovídají všem možným odpovědím. Množinu všech podmnožin množiny Θ označme P(Θ), což je tzv. potenční množina.

V Dempster – Shaferově teorii se s mírou důvěry pracuje analogicky jako s hmotností fyzikálního objektu. Nenulová váha je přiřazována pouze těm prvkům v množině P(Θ), kterým je vyjadřována důvěra. Jakákoliv jiná důvěra, která není přiřazena určitému prvku z P(Θ), je vzhledem k tomuto prvku neutrální, tzn. že se nepovažuje za nedůvěru.

Uvažujme například, že vojenské zařízení na identifikaci přátelského či nepřátelského letadla (IPN) vyšle k blížícímu se letounu radiovou zprávu. V případě, že se jedná o přátelský letoun, měl by tento letoun sdělit svůj identifikační kód. Letoun, který neodpoví, je považován za nepřátelský bombardér či stíhačku. Problém je však v tom, že se může jednat i o přátelský letoun, který přesto nemusí odpovědět. Důvodem může být např. chybné (poškozené) přijímací zařízení, porucha ve vyslání signálu IPN aj.

Předpokládejme například, že jestliže IPN nezíská od blížícího se letounu identifikační kód, pak se s důvěrou 0,7 jedná o nepřátelský bombardér či stíhačku.

Podmnožině {B, S} = {bombardér, stíhačka} množiny Θ je tedy přiřazena váha:

$m_1(\{B, S\}) = 0{,}7$

Zbytek důvěry je ponechán v množině Θ, jako neutrální důvěra, tj.

$m_1(\Theta) = 1 - 0{,}7 = 0{,}3$

Hlavní rozdíl mezi Dempster – Shaferovou teorií a teorií pravděpodobnsoti je v tom, že zatímco v teorii pravděpodobnosti by platilo:

P(nepřítel) = 0,7

P(přítel) = 0,3;

tak v Dempster – Shaferově teorii platí, že cíl je nepřátelský minimálně s důvěrou 0,7 a zbývající důvěra 0,3 je v určitém nám dosud neznámém poměru rozdělena mezi dodatečnou důvěru (že se jedná o nepřítele) a v nedůvěru.

Takto zavedená váha má mnohem větší volnost než pravděpodobnost, jak je patrné z tabulky:

Dempster – Shaferova teorie	Teorie pravděpodobnosti
m(Θ) nemusí být rovno 1	$\Sigma_i P_i = 1$
je-li $X \subseteq Y$, nemusí být $m(X) \le m(Y)$	$P(X) \le P(Y)$
nevyžaduje vztah mezi m(X) a m(X‘)	P(X) + P(X‘) = 1

Každá váha může být formálně vyjádřena jako funkce, která zobrazuje každý prvek z P(Θ) do intervalu ⟨0,1⟩:

$$m: P(\Theta) \rightarrow \langle 0,1 \rangle.$$

Přitom se dále definuje $m(\emptyset) = 0$

a současně se vyžaduje, aby součet všech vah přiřazených všem možným podmnožinám množiny Θ, tj. součet všech vah prvků množiny P(Θ) byl roven jedné:

$$\sum_{X \in P(\Theta)} m_1(X) = 1. \qquad (1.11)$$

V našem případě skutečně platí:

$$\sum_{X \in P(\Theta)} m_1(X) = m(\{B,S\}) + m(\Theta) = 0,7 + 0,3 = 1.$$

Předpokládejme nyní, že nějaké druhé zařízení pracující na odlišném principu identifikuje cíl jako nepřátelský bombardér s vahou 0,9, tj.

$m_2(\{B\}) = 0,9$ a $m_2(\Theta) = 0,1$

Tyto váhy mohou být kombinovány s předchozími vahami:

$m_1(\{B,S\}) = 0,7$ a $m_1(\Theta) = 0,3$

pomocí Dempsterova pravidla ve tvaru:

$$\underline{m_v(Z) = m_1 \oplus m_2(Z) = \sum_{X \cap Y = Z} m_1(X) m_2(Y)}, \qquad (1.12)$$

kde součet probíhá přes všechny prvky (množiny), jejichž průnikem je Z. Při využití tohoto pravidla se předpokládá, že kombinované důkazy (poznatky) jsou vzájemně nezávislé.

Aplikací Dempsterova pravidla v našem případě obdržíme:

$m_v(\{B\}) = m_1 \oplus m_2(\{B\}) = m_1(\{B,S\})\, m_2(\{B\}) + m_1(\Theta) m_2(\{B\}) =$
$= 0,7 * 0,9 + 0,3 * 0,9 = 0,63 + 0,27 = 0,9$ bombardér

$m_v(\{B,S\}) = m_1 \oplus m_2(\{B,S\}) = m_1(\{B,S\}) m_2(\{\Theta\}) =$
$= 0,7 * 0,1 = 0,07$ bombardér nebo stíhačka

$m_v(\{\Theta\}) = m_1 \oplus m_2(\Theta) = m_1(\Theta)\, m_2(\Theta)$
$= 0,3 * 0,1 = 0,03$ zbývající důvěra a nedůvěra

$m_v(\{B\})$ reprezentuje výslednou váhu (míru důvěry), že cílem je právě nepřátelský bombardér a nic jiného. Avšak váhy $m_v(\{B,S\})$ a $m_v(\{\Theta\})$ obsahují ještě další informaci. Protože množiny {B,S} a Θ zahrnují také bombardér, může určitá část jim odpovídajících vah ještě také přispět ke zvýšení důvěry v bombardér, nejvíce však o 0,1 neboť součet:

$m_v(\{B,S\}) + m_v(\{\Theta\}) = 0,07 + 0,03 = 0,1.$

Míra důvěry v bombardér leží tedy v rozmezí od 0,9 do 1, tj. od minimální důvěry až po maximální přijatelnou důvěru.

Označme minimální jistou důvěru *Bel* a maximální ještě možnou důvěru *Pls*. Zřejmě $0 \le Bel \le Pls \le 1$.

Interpretace některých intervalů $\langle Bel, Pls\rangle$ je uvedena v tabulce:

$\langle Bel, Pls\rangle$	Význam
$\langle 1, 1\rangle$	Absolutní pravdivost
$\langle 0, 0\rangle$	Absolutní nepravdivost
$\langle 0, 1\rangle$	Absolutní nejistota
$\langle Bel, 1\rangle$, $0 < Bel < 1$	Tendence – podpora
$\langle 0, Pls\rangle$, $0 \le Pls < 1$	Tendence – odmítnutí
$\langle Bel, Pls\rangle$, $0 < Bel \le Pls < 1$	Podpora i odmítnutí

Minimální jistá důvěra či podpora *Bel* je celková důvěra v množinu a její podmnožiny a je určena jako součet vah přiřazených této množině a jejím podmnožinám:

$$Bel(X) = \sum_{Y \subseteq X} m(Y)\,. \tag{1.13}$$

V našem případě je hodnota minimální jisté důvěry (podpory) *Bel* pro množinu {B,S} získaná na základě poznatku získaného od IPN:

$$Bel_1(\{B,S\}) = m_1(\{B,S\}) + m_1(\{B\}) + m_1(\{S\}) = 0{,}7 + 0 + 0 = 0{,}7.$$

Mezi hodnotou minimální jisté důvěry *Bel* a vahou je rozdíl v tom, že váha reprezentuje důvěru přiřazenou právě množině a nikoliv i jejím podmnožinám. Platí mezi nimi následující vztah:

$$m(X) = \sum_{Y \subseteq X} (-1)^{|X-Y|} Bel(Y),$$

kde $|X\text{-}Y|$ udává kardinalitu, tj. počet prvků v množině X – Y.

Protože je hodnota *Bel* určena pomocí vah vztahem (1.13), může být i kombinace dvou podpor pro bombardér (funkcí důvěry) vyjádřena s pomocí vztahu (1.13):

$$Bel_1 \oplus Bel_2(\{B\}) \overset{1.13}{=} Bel_v(\{B\}) \overset{1.1\,3}{=} m_v(\{B\}) + m_v(\{\emptyset\}) = \overbrace{m_1 \oplus m_2(\{B\}) + \underbrace{m_1 \oplus m_2(\emptyset)}_{0}}^{0{,}7 * 0{,}9 + 0{,}3 * 0{,}9}$$
$$= 0{,}90 + 0 = 0{,}90$$

neboť váha prázdné množiny je (definována) rovna nule.

Celková podpora pro podmnožinu (bombardér, stíhačka) má již více podmnožin než v předchozím případě a

$$\underline{Bel_1 \oplus Bel_2(\{B,S\})} \overset{1.13}{=} Bel_v(\{B,S\}) \overset{1.13}{=:} m_v(\{B,S\}) + m_v(\{B\}) + m_v(\{S\}) + m_v(\{\emptyset\}) =$$

$$m_1 \oplus m_2(\{B,S\}) \overset{1.12}{+} m_1 \oplus m_2(B) + m_1 \oplus m_2(\{S\}) = 0{,}07 + 0{,}90 + 0 = 0{,}97 \tag{1.14}$$

Celková podpora pro celou množinu Θ

$$\Theta = \{\text{dopravní letadlo, stíhačka, bombardér}\} = \{D,S,B\}$$

je určena výpočtem:

$$\begin{aligned} Bel_1 \oplus Bel_2(\Theta) = Bel_v(\{\Theta\}) &= m_v(\{\Theta\}) + m_v(\{B,S\}) + m_v(\{B\}) \\ &= m_1 \oplus m_2(\Theta) + m_1 \oplus m_2(\{B,S\}) + m_1 \oplus m_2(B) \\ &= 0{,}03 + 0{,}07 + 0{,}90 \\ &= 1 \end{aligned}$$

Množina Θ sice obsahuje podmnožiny {}, {D,B}, {D}, {D,S}, {S}, ale ty však mají nulovou váhu. Podmnožina $Bel\{\Theta\} = 1$ je splněna v každém případě, tj. součet všech vah přiřazených množině Θ a všem jejím podmnožinám musí být vždy podle (1.11) roven jedné.

Interval míry důvěry pro množinu Q označme

$EI(Q) = \langle Bel, Pls \rangle$

Tento interval lze také vyjádřit:

$EI(Q) = \langle Bel(Q), 1- Bel(Q‘) \rangle$.

Je-li v našem příkladě Q = {B}, potom Q’ = {D,S} a

$Bel(\{D,S\}) = m_1 \oplus m_2(\{D,S\}) + m_1 \oplus m_2(\{D\}) + m_1 \oplus m_2(\{S\}) = 0 + 0 + 0 = 0$

a skutečně platí

$EI(\{B\}) = \langle 0{,}9;\ 1\text{-}0 \rangle = \langle 0{,}9;\ 1 \rangle$.

Obdobně pro Q = {B,S} je Q‘ = {D} a potom zřejmě:

$Bel(\{D\}) = 0$

$Bel(\{B,S\}) = 0{,}97$

a tedy

$EI(\{B,S\}) = \langle 0{,}97;\ 1\text{-}0 \rangle = \langle 0{,}97;\ 1 \rangle$

$EI(\{D\}) = \langle 0;\ 0{,}03 \rangle$.

Přípustnost *Pls*(X) tudíž představuje míru, do které zjištěné poznatky nestačí k odmítnutí X.

Zřejmě: $Pls(X) = 1 - Bel(X‘) = 1 - \sum_{Y' \subseteq X'} m(Y')$.

Pokračujeme nyní dále v našem příkladu a předpokládejme, že třetí zařízení poskytne s předchozími výsledky konfliktní poznatek, že se jedná o dopravní letadlo s mírou důvěry 0,95, tj.:

$m_3(\{D\}) = 0{,}95$ a $m_3(\Theta) = 0{,}05$

Snadno odvodíme, že:

$m_1 \oplus m_2 \oplus m_3(\{D\}) = 0{,}0285$

$m_1 \oplus m_2 \oplus m_3(\{B\}) = 0{,}045$

$m_1 \oplus m_2 \oplus m_3(\{B,S\}) = 0{,}0035$

$m_1 \oplus m_2 \oplus m_3(\Theta) = 0{,}0015$

$m_1 \oplus m_2 \oplus m_3(\varnothing) = 0$ (podle definice)

Ostatní váhy pro zbývající podmnožiny jsou nulové. V tomto případě však zjistíme, že součet vah je menší než jedna, neboť:

$$\sum_{X \in P(\Theta)} m_1 \oplus m_2 \oplus m_3(X) = 0{,}0285 + 0{,}045 + 0{,}0035 + 0{,}0015 = 0{,}0785.$$

Řešení tohoto problému je založeno na normalizaci prvků s nenulovou vahou, a to vydělením každé váhy rozdílem

$1 - K$,

kde K je definováno pro libovolné množiny X a Y vztahem:

$$K = \sum_{X \cap Y = prázdná\ \ mn.} m_i(X) m_j(Y) \qquad i \neq j$$

V našem případě (uvedeny jsou pouze kombinace dávající nenulový příspěvek):

$K = m_1 \oplus m_2(\{B\}) m_3(\{D\}) + m_1 \oplus m_2(\{B,S\}) m_3(\{D\}) =$

$= 0{,}9 * 0{,}95 + 0{,}07 * 0{,}95 = 0{,}9215$

$1 - K = 1 - 0{,}9215 = 0{,}0785$

Po vydělení získáme tyto normalizované hodnoty:

$m_1 \oplus m_2 \oplus m_3(\{D\}) \doteq 0{,}363$

$m_1 \oplus m_2 \oplus m_3(\{B\}) \doteq 0{,}573$

$m_1 \oplus m_2 \oplus m_3(\{B,S\}) \doteq 0{,}045$

$m_1 \oplus m_2 \oplus m_3(\Theta) \doteq 0{,}019$

Jeden důkaz ve prospěch {D} = {dopravní letadlo} značně změnil důvěru v {B} = {bombardér}, jak se očekávalo.

Zřejmě:

Bel({B}‘) = *Bel*({D,S})

= $m_1 \oplus m_2 \oplus m_3(\{D,S\}) + m_1 \oplus m_2 \oplus m_3(\{D\}) + m_1 \oplus m_2 \oplus m_3(\{S\})$ =

= 0 + 0,363 + 0 = 0,363

a interval míry důvěry pro bombardér

EI({B}) = ⟨*Bel*({B}); 1 - *Bel*({B}‘)⟩ = ⟨0,573; 0,637⟩.

Díky třetímu poznatku, konfliktnímu s předchozími, se podstatně snížila podpora i přijatelnost bombardéru (jistá i maximálně možná důvěra).

Obecná forma Dempsterova pravidla má tedy tvar:

$$m_1 \oplus m_2(Z) = \frac{\sum_{X \cap Y = Z} m_1(X) m_2(\overset{Y}{\cancel{X}})}{1 - K}, \qquad (1.15)$$

kde $K = \sum_{X \cap Y = prázdná\ \ mn.} m_1(X) m_2(Y)$.

Hodnota K přitom naznačuje množství konfliktu v důkazu. Zatímco K = 0 odpovídá úplné kompatibilitě; K = 1 odpovídá úplnému rozporu.

Určitým problémem Dempster-Shaferovy teorie je normalizace, která může vést až k výsledkům, které mohou být v rozporu s naším očekáváním. Příčina tkví v tom, že normalizací je ignorována důvěra, že uvažovaný objekt vůbec neexistuje.

Příklad:

Předpokládejme dva lékaře A a B, kteří mají následující důvěru v pacientovu nemoc:

m_A {zápal mozkových blan} = 0,99

m_A {mozkový otok} = 0,01

m_B {otřes mozku} = 0,99

m_B {mozkový otok} = 0,01

Oba dva lékaři sice přisuzují vhodně extrémně nízkou šanci 0,01 mozkovému otoku, ale značně se liší v názoru na hlavní podstatu problému. Aplikací Dempsterova pravidla však dospějeme k důvěře v mozkový otok.

Podle (1.15) totiž:

$$m_1 \oplus m_2(Z) = \frac{0{,}01 * 0{,}01}{1 - (0{,}99 * 0{,}99 + 0{,}99 * 0{,}01 + 0{,}01 * 0{,}99)} = 1 \qquad (1.16)$$

(Z … mozkový otok)

Tento výsledek odporuje naší intuici a navíc vůbec nezávisí na ostatních vahách pro zápal mozkových blan i otřes mozku.

9 INDUKČNÍ SYSTÉMY

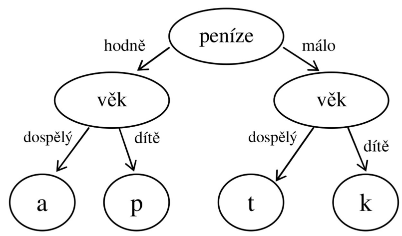
Indukční znalostní systém = systém, který z množiny příkladů (trénovacích dat) vyvodí množinu znalostí o datech

Indukční systém s pravidly = systém, který z množiny příkladů (trénovacích dat) vyvodí množinu pravidel platných pro data

Příklad:

Jaký dárek?

Peníze/fakta	Věk / fakta	Dárek/Výsledek
hodně	dospělý	auto
hodně	dítě	počítač
málo	dospělý	topinkovač
málo	dítě	kalkulačka

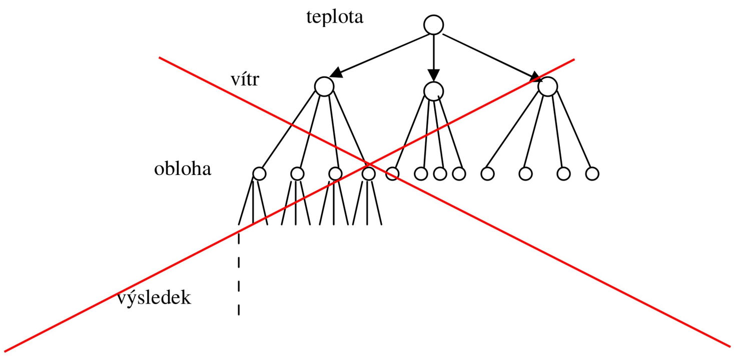


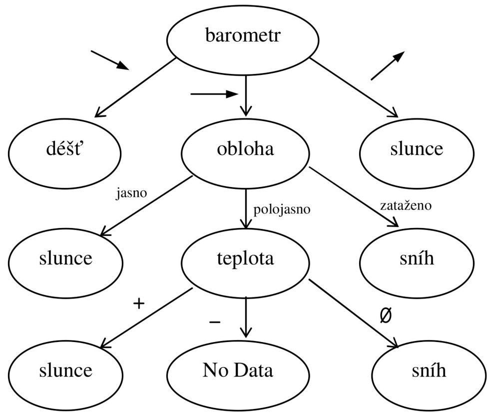
IF máme hodně peněz
AND dárek je pro dospělého
THEN kupujeme auto
………

Příklad:

Předpověď počasí

teplota	vítr	obloha	barometr	výsledek
+	Z	zataženo	↘	déšť
–	*	zataženo	→	sníh
+	V	zataženo	↗	slunce
+	*	polojasno	→	slunce
*	*	jasno	→	slunce
+	J	jasno	↘	déšť
0	S	polojasno	→	sníh





Problém odvození množiny pravidel je převeden na problém konstrukce rozhodovacího stromu (tj. chceme, aby rozhodovací strom měl co nejmenší větvení ⇒ malá množina pravidel)

CLS algoritmus

C … množina příkladů
W_i…třídy výsledků

1. Když všechny příklady jsou ze třídy W_i, pak vytvoříme uzel W_i a konec.
   Jinak vybereme (užitím heuristiky) faktor F_j (vítr) s hodnotami H_{j1}, H_{j2}, …, H_{jNj} a vytvoříme uzel s F_j. (tzv. hodnotící funkce)
2. Rozdělíme množinu příkladů C do podtříd C_{j1}, C_{j2}, …, C_{jNj} podle F_j a jeho hodnot.
3. Aplikujeme algoritmus rekurzivně na každou C_{jk}.

ID3 algoritmus

1. Vybereme náhodně podmnožinu z C (tzv. okno)
2. Aplikuje CLS algoritmus
3. Projde všechny příklady z {C – (okno)} a najde výjimky
4. Když existují výjimky, pak vybere některé z nich a přidá je do okna a jde na krok 2.
   Jinak konec a strom je hotov.

Hodnotící funkce může mít tvar např. entropie:

$$H = -\sum_{i=1}^{N} p_i \log_2 p_i$$

N … počet výsledných tříd W_i (např. výsledek: déšť, sníh, slunce; faktor: barometr)

Příklad:

$P_1 = P(\text{výsledek} = \text{déšť} \mid \text{barometr} \searrow) = 1$
$P_2 = P(\text{výsledek} = \text{sníh} \mid \text{barometr} \searrow) = 0$
$P_3 = P(\text{výsledek} = \text{slunce} \mid \text{barometr} \searrow) = 0$

$H_{\searrow} = -1*0 - 0*\log 0 - 0*\log 0 = 0$

$H_{\nearrow}$ … spočítám i pro barometr $\nearrow$
$H_{\rightarrow}$ … spočítám i pro barometr $\rightarrow$

$$H_v = H_{\text{výsledné}} = \sum_{i\in\{\nearrow,\searrow,\rightarrow\}}^{n} H_i P(\text{barometr } i)$$

n … počet různých hodnot faktoru

n=3, $\{\searrow, \nearrow, \rightarrow\}$

Spočítám H_v i pro vítr, teplotu, oblohu a vyberu faktor, pro který H_v nabývá nejmenší hodnoty.

N=3, {déšť, sníh, slunce}

1.barometr ($\searrow, \nearrow, \rightarrow$)

1.1 $\searrow$
$P_1=P(\text{déšť} \mid \searrow) = 1$
$P_2=P(\text{sníh} \mid \searrow) = 0$
$P_3=P(\text{slunce} \mid \searrow) = 0$
$H_{\searrow} = 1*0 + 0*0 + 0*0 = 0$

1.2 $\nearrow$
$P_1=P(\text{déšť} \mid \nearrow) = 0$
$P_2=P(\text{sníh} \mid \nearrow) = 0$
$P_3=P(\text{slunce} \mid \nearrow) = 1$
$H_{\nearrow} = 0$

1.3 $\rightarrow$
$P_1=P(\text{déšť} \mid \rightarrow) = 0$
$P_2=P(\text{sníh} \mid \rightarrow) = 1/2$
$P_3=P(\text{slunce} \mid \rightarrow) = 1/2$
$H_{\rightarrow} = 0 + (-\log_2 (1/2)) = \log_2 2 = 1$

$$H_{\text{barometr}} = H_{\searrow} * P(\searrow) + H_{\nearrow} * P(\nearrow) + H_{\rightarrow} * P(\rightarrow) = 1 * \frac{4}{7} = \frac{4}{7}$$

2.obloha (zat., pol., jasno)

2.1
$P_1=P(\text{déšť} \mid \text{zat.}) = 1/3$
$P_2=P(\text{sníh} \mid \text{zat.}) = 1/3$
$P_3=P(\text{slunce} \mid \text{zat.}) = 1/3$
$H_{\text{zat.}} = \log_2 3 = 1{,}58$

2.2
$P_1=P(\text{déšť} \mid \text{pol.}) = 0$
$P_2=P(\text{sníh} \mid \text{pol.}) = 1/2$
$P_3=P(\text{slunce} \mid \text{pol.}) = 1/2$
$H_{\text{pol.}} = \log_2 2 = 1$